НА ПОДСТУПАХ К СВЕРХРАЗУМУ

Сильный искусственный интеллект

* название книге дала нейросеть RuGPT-3

альпина паблишер | бизнес

На подступах к сверхразуму

СИЛЬНЫЙ ИСКУССТВЕННЫЙ ИНТЕЛЛЕКТ

альпина паблишер | бизнес

Москва
2021

УДК 004.8
ББК 32.813
С36

Продюсер *И. Фурман*
Научный редактор *А. С. Потапов, доцент, д-р. техн. наук, SingularityNet*
Старший редактор *Д. Варламова*
Редактор *А. Воеводская*
Редактор инфографики *Г. Неяскин*

С36 Сильный искусственный интеллект : На подступах к сверхразуму / Александр Ведяхин [и др.]. — М.: Интеллектуальная Литература, 2021. — 232 с.

ISBN 978-5-907394-18-6

Эта книга — первый кросс-дисциплинарный гид по искусственному интеллекту на русском языке. Сильный искусственный интеллект — это следующая ступень в развитии ИИ, не обязательно наделенного самосознанием, но, в отличие от современных нейросетей, способного справляться с широким кругом задач в разных условиях. Авторы книги рассказывают о том, что должен уметь сильный ИИ, какие научные подходы помогут его создать и как изменится мир с его появлением.

УДК 004.8
ББК 32.813

ISBN 978-5-907394-18-6

Оглавление

К читателям

Дорогие читатели!

Сегодня технологии искусственного интеллекта прочно вошли в нашу повседневную жизнь, став незаменимыми помощниками в решении множества задач — от эффективной поддержки наших усилий при выполнении рутинных действий, обработке и анализе больших массивов неструктурированной информации — до оказания помощи в реализации нашего творческого потенциала и развития наших креативных навыков. А в таких индустриях как здравоохранение, беспилотный транспорт и охрана общественного порядка, технологии ИИ помогают принимать решения более качественно и быстрее — ровно тогда, когда счет идет на секунды, и эти секунды способны спасти чьи то жизни!

В большинстве своем существующие решения являются примерами реализации технологий узко специализированного искусственного интеллекта, требующего настройки и перепроверки со стороны человека. Чтобы так же хорошо решать разнообразные комплексные задачи, как это делают люди, машины должны научиться строить причинно-следственные модели окружающей среды и ориентироваться в разных контекстах, а не просто максимизировать успех при решении какой-то узкой задачи. Они должны понимать физические, психологические и другие законы нашего мира и уметь связывать новую

информацию в общую картину с тем, что уже знают. Чтобы добиться этого, нам необходимо преодолеть очередной технологический рубеж — создание Общего искусственного интеллекта или AGI.

Вместе с коллективом авторов этой книги мы впервые обобщили и систематизировали накопленные знания в области Общего искусственного интеллекта — от компьютерных наук и машинного обучения до нейронаук и психологии. Оценивая ретроспективу научных достижений в области технологий искусственного интеллекта, особенно впечатляющим оказался «взрывной рост» технологий в последние несколько лет, возможный во многом благодаря экспоненциальному росту доступных вычислительных мощностей и снятию технологический ограничений накопления больших объёмов данных для обучения алгоритмов. Все это позволило давно известным архитектурам многослойных нейронных сетей успешно решать задачи в самых различных сферах человеческой деятельности. При этом, несмотря на особое внимание, которое уделяется нейронным сетям и глубокому обучению, это далеко не единственный путь к Общему искусственному разуму.

Эта книга будет прежде всего интересна всем тем, кто хочет понять, с помощью каких подходов может быть создан Общий искусственный интеллект и какую форму он, вероятнее всего, приобретет в различных сферах прикладного применения. По опыту развития современных технологий машинного обучения мы видим, что путь от идеи до индустриальных решений может оказаться значительно короче, чем это кажется вначале.

Сегодня уже очевидно, что любая выбранная нами стратегия движения в развитии Общего искусственного интеллекта и хорошо организованные управленческие усилия неизбежно столкнутся с ключевым вопросом — вопросом выбора корректных критериев отнесения того или иного исследования к области Общего искусственного интеллекта. Для этого нам важно хорошо понимать структуру исследуемой области,

составить единую и непротиворечивую теоретическую базу, а затем определить наиболее перспективные подходы к развитию технологий и их последующей индустриализации. Этому и посвящена книга. Мы верим, что лишь сделав ставку на эффективное сотрудничество специалистов разных направлений — от глубокого обучения и вероятностного программирования до робототехники и когнитивистики, а также поддерживая междисциплинарные исследования, можно добиться первых ощутимых прикладных результатов в области развития Общего искусственного интеллекта. Крайне важным фактором создания и развития технологий Общего искусственного интеллекта является обеспечение сквозного целеполагания между прикладными (или индустриальными) задачами, фундаментальными исследованиями и системой образования — так называемая триада Practice–Education–Research.

Таюке мы считаем важным уделить особое внимание профессионального сообщества теме соблюдения принципов этического применения современных технологий. В частности, необходимо на международном и межиндустриальном уровнях выработать единые и общепризнанные стандарты, позволяющие обеспечивать безопасное и социально-полезное применение технологий Общего искусственного интеллекта. В этом контексте соответствующие усилия должны быть предприняты для обеспечения стабильности и интерпретируемости работы алгоритмов, лежащих в основе технологий Общего искусственного интеллекта.

Работая над книгой, мы убедились в том, что у России большой потенциал развития прорывных технологий в области Общего искусственного интеллекта. Хочется надеяться, что наш общий труд подтолкнет исследователей, инженеров, представителей бизнеса и государства к эффективному сотрудничеству в создании принципиально новых подходов на пути к AGI и это позволит России занять лидирующее место в гонке мировых держав в области ИИ.

Многие из нас прекрасно осознают неминуемость масштабных технологических трансформаций нашего общества. Доверие к технологиям искусственного интеллекта в обществе пока только формируется, много говорится о негативных сценариях, которые могут привести к катастрофическим последствиям для цивилизации. Важно слышать и понимать опасения людей. Мы должны научиться использовать новые технологии во благо нашего общества.

И несмотря на то, что нам с вами предстоит еще пройти непростой и большой путь для достижения первых прикладных результатов в области применения технологий Общего искусственного интеллекта, уже сейчас мы можем видеть весьма многообещающие решения в этой области.

Я желаю вам увлекательного и продуктивного чтения! Помните — исключительно важно всеобщее понимание того, что достижение результатов в этом путешествии должно стать благом для всего Человечества!

Герман Греф, Президент,
Председатель Правления Сбера

Коллектив авторов

Бурцев М. С., канд. физ.-мат. наук, МФТИ.

Бухвалов О. Л., канд. техн. наук, Brain Garden.

Ведяхин А. А., канд. экон. наук, первый заместитель председателя правления, ПАО Сбербанк.

Витяев Е. Е., д-р физ.-мат. наук, Институт математики СО РАН, профессор Новосибирского государственного университета.

Еременко М.А, ПАО Сбербанк.

Ефимов А. Р., ПАО Сбербанк, НИТУ «МИСиС».

Колонин А. Г., канд. техн. наук, Новосибирский государственный университет, Aigents.

Курпатов А. В., ПАО Сбербанк.

Мазин В. А., канд. физ.-мат. наук, Mind Simulation AGI laboratory.

Марков С. С., ПАО Сбербанк.

Молчанов А. А., ПАО Сбербанк.

Нейросеть RuGPT-3.

Николенко С. И., канд. физ.-мат. наук, ПОМИ РАН, Neuromation.

Очеретный А.С., ПАО Сбербанк.

Панов А. И., канд. физ.-мат. наук, ФИЦ ИУ РАН, МФТИ.

Пономарев Д. К., канд. физ.-мат. наук, Институт систем информатики СО РАН, Новосибирский государственный университет.

Потапов А. С., доцент, к.т.н., SingularityNet.

Салихов Д. Р., ПАО Сбербанк.

Сарапулов Г. В., Brain Garden.

Свириденко Д. И., д-р физ.-мат. наук, доцент, профессор, Институт математики СО РАН.

Черток А. В., канд. физ.-мат. наук, ПАО Сбербанк.

Шаляпин С. О., Естественный интеллект.

Шелехов В. И., канд. техн. наук, доцент, Институт систем информатики СО РАН.

Franz A., PhD, OCCAM.

Введение

Технологии искусственного интеллекта с самого своего появления демонстрировали удивительные достижения в решении задач, с которыми, как традиционно считалось, способен справиться только человеческий разум.

Сейчас технологии ИИ становятся массовыми и повсеместными, проникли в нашу повседневную жизнь и вряд ли ее покинут. Они используются в поисковых и рекомендательных системах, транспорте, логистике, банковском деле, планировании бизнес-процессов, производстве и научных исследованиях. Они уже давно не ограничиваются цифровой реальностью, проникая в быт. Нас начинают окружать домашние роботы, беспилотные аппараты, умные дома и города, не говоря уже о приложениях с элементами ИИ для смартфонов и персональных компьютеров. Технологии ИИ, включающие машинное обучение, научились неплохо справляться с анализом изображений, звука, речи и текстов на естественных языках. Иногда они делают это не просто на человеческом, а на сверхчеловеческом уровне. Технологии искусственного интеллекта открывают перед нами огромные перспективы. Они способны придать новый импульс развитию мировой экономики, оказать позитивное влияние на все сферы нашей жизни. В этом году они значительно помогли медицинскому сообществу в борьбе с пандемией.

Из-за многочисленных успехов последнего времени может сложиться впечатление, что недавний впечатляющий прогресс

технологий ИИ достиг насыщения. Но это вовсе не так, и диапазон их применения только растет. Даже уже известные технологии имеют массу еще не реализованных возможностей по внедрению. Либо для этого не нашлось свободных специалистов, либо стоимость разработки и внедрения перевешивает ожидаемую прибыль, либо соответствующие технологии, уже существующие в теории, еще не способны предоставить решения достаточно качественного, чтобы быть полезными и удобными в ежедневном использовании. А главное, новые, более перспективные технологии появляются быстрее, чем старые успевают стать в полной мере использованными.

Однако существуют и такие крайне важные задачи, для которых имеющихся технологий просто недостаточно, — например, исследовательские. Скажем, интеллектуальная система, которая могла бы продвинуться в решении проблемы человеческого старения, должна была бы не только анализировать научные статьи, причем на более глубоком уровне, чем поверхностные корреляции между словами, но и моделировать взаимосвязи между различными подсистемами и процессами организма. Такой системы у нас пока нет, и, возможно, излишне оптимистичным было бы полагать, что она может достаточно быстро появиться в результате естественного развития доминирующих сейчас технологий ИИ, нацеленных на решение узких задач.

Эта книга основана на исследовании лучших российских специалистов по ИИ, посвященном общему искусственному интеллекту. Это ИИ, способный самообучаться и решать разнообразные задачи в разных контекстах. Системы искусственного интеллекта смогли бы помочь человечеству справиться с самыми сложными вызовами: построением более справедливого общества, поиском лекарств от смертельных заболеваний, предупреждением катастроф и т.д. Кроме того, развитые технологии ИИ — это важное стратегическое преимущество для государства на внешнеполитической арене. Как говорил В.В. Путин, «Искусственный интеллект — это будущее не

только России, но и всего человечества, и тот, кто будет лидером в этой сфере, станет правителем мира». Пока разработка таких систем вызывает много сложностей, но существует ряд подходов, которые могли бы продвинуть нас в решении этой задачи. Наша книга представляет собой самый полный и глубокий обзор этих подходов и первый шаг к выработке общего бэкграунда для заинтересованных в AGI на русском языке. Он поможет специалистам из разных областей ИИ объединить свои знания и выработать стратегию по созданию общего искусственного интеллекта. Эта книга написана научно-популярным языком, делающим ценные знания доступными для более широкой аудитории, кроме того, скоро выйдет чисто научная версия для глубокого погружения в тему и планируется создание практического руководства.

Недостатки узкоспециальных систем

Применение технологий ИИ все еще не настолько впечатляюще, как могло бы быть, по многим причинам. Но все они обусловлены одним фундаментально важным фактором. Большинство таких систем остаются узкоспециализированными, а еще точнее, позволяют достичь качественных решений только для узких задач. Это свойство не изменилось со времен экспертных систем ИИ, которые всегда требовали формализованных описаний и вручную заложенного эвристического знания. Экспертные системы, системы компьютерного зрения и любые другие знали примерно столько, сколько им сообщили разработчики,— и не больше. Поэтому уже полвека назад в адрес ИИ звучала критика в том духе, что «компьютер, запрограммированный на решение тысячи задач, не способен самостоятельно научиться решать тысяча первую».

Эта критика отчасти справедлива и до сих пор. Хотя современные системы машинного обучения используют довольно сложные методы работы с данными (об этом подробно рассказывается в следующих главах), каждое конкретное решение все еще специализировано под конкретную задачу. Часто эта специализация оказывается даже выше, чем у старых, классических систем, потому что вручную разработанные представления информации обычно более общие, а выученные машиной — подгоняются под конкретную выборку. В результате опыт решения одной задачи плохо переносится на решение другой задачи или даже новый набор данных.

Контраст бывает разительным. Компьютерную модель можно сравнительно легко научить по размеченным данным видеокамер определять, когда один и тот же произвольный человек появляется на разных камерах с неперекрывающимися полями зрения, и она будет точна в 95% случаев. Но стоит проверить ее на другом, незнакомом наборе камер, и точность идентификации упадет ниже 10%. Чтобы не допускать таких провалов, разработчики тренируют модели на нескольких наборах данных с применением методов трансферного обучения, но и этого недостаточно: качество идентификации все равно может оказаться непригодным для практического использования. Иногда обучаемые модели при применении в новых условиях уступают даже необучаемому методу, который пользуется общими признаками, сконструированными вручную.

Другой яркий пример — модель, обученная играть в игры Atari. Она была способна играть в любую из множества игр, хотя тратила на тренировку гораздо больше времени, чем человек, и не во всех испытаниях была способна демонстрировать сверхчеловеческий уровень (хотя недавно компьютер превзошел уровень среднего человека во всех играх). Каждой игре она училась отдельно, и после самого незначительного изменения параметров ей приходилось переучиваться. Поменяйте цвет стен, мимо которых бежит персонаж

компьютерной игры, — и большинство людей этого даже не заметит, а такой модели придется начинать тренировку заново.

В некоторых приложениях машинное обучение все еще не смогло заменить классические технологии ИИ, так как все еще плохо работает со структурированной информацией, например априорными знаниями и причинно-следственными связями. Это тоже можно трактовать как свидетельство узости моделей машинного обучения, но уже более глубокого уровня, чем уровень приложений.

> **Разные авторы подчеркивают разные недостатки современных систем ИИ. Кто-то считает, что основной проблемой для них является приобретение новых навыков. Кто-то обращает внимание на нехватку надежности. Но все эти проблемы — симптомы одного свойства: недостаточной широты методов ИИ.**

Приобретение новых навыков или надежность не ограничивают решение задач в узких предметных областях. Например, промышленные роботы в строго контролируемых производственных условиях не ошибаются. Проблема возникает тогда, когда методы, придуманные для решения узких задач в предсказуемой среде, начинают использовать широко.

Удивительным образом узкая специализация служит основным ограничением при разработке и внедрении систем ИИ для решения любых задач, как частных, так и общих. Частных — потому что чем конкретнее задачи, тем их больше, и каждая из них требует затрат на разработку. В результате и специалистов не хватает, и разработка и внедрение часто оказываются коммерчески неоправданными. Общих — потому что чем шире задачи, тем больше труда по разметке данных или инженерии знаний они требуют. При решении произвольных задач, особенно сложных, узость методов из количественной проблемы превращается в качественную.

Такой качественный переход призвана совершить область общего искусственного интеллекта (Artificial General Intelligence, AGI), который нужен именно для того, чтобы системы ИИ были общими, и который противопоставляется «узкому» ИИ (Narrow AI). Развитие AGI вместо существующих теперь узких методов или вместе с ними может заложить новый виток технологического прогресса человечества, трансформировать технологии, науку и общество. Поэтому актуальность исследований AGI трудно переоценить.

Общий ИИ — не сильный и не слабый

В первую в мире лабораторию искусственного интеллекта в сопровождении профессора Марвина Минского зашел известный философ и аналитик Хьюберт Дрейфус. Для корпорации RAND он недавно написал аналитический отчет с говорящим названием «Алхимия и искусственный интеллект».

— Вы знаете, что компьютеры принципиально не способны на творчество и что они никогда не смогут, скажем, даже обыграть гроссмейстера в шахматы? — произнес философ, продолжая спор.

— А не хотите ли сыграть с нашим компьютером? Мои студенты как раз недавно закончили работу над шахматной программой, — спросил профессор.

— Извольте. Я и сам неплохо играю.

Вряд ли именно такой диалог состоялся между Хьюбертом Дрейфусом и Марвином Минским, но известный философ в действительности в 1965 г. написал манускрипт «Алхимия и искусственный интеллект», в котором критиковал наивность разработчиков искусственного интеллекта и пытался доказать, что у ИИ есть непреодолимый предел развития, что компьютеры

обладают ограничениями, от которых человеческий разум свободен, и что, в частности, такие игры, как шахматы или го, принципиально не подвластны компьютеру. А в 1969 г. действительно состоялась партия между Дрейфусом и шахматной программой «Мак Хак», написанной Ричардом Гринблаттом в МТИ, и в этой партии философ потерпел поражение.

Это событие, однако, не помешало Дрейфусу и дальше рассуждать об ограниченности компьютеров и в 1972 г. написать трактат «Чего не могут компьютеры» (в русском переводе «Чего не могут вычислительные машины: Критика искусственного разума»), который упорно переписывался вплоть

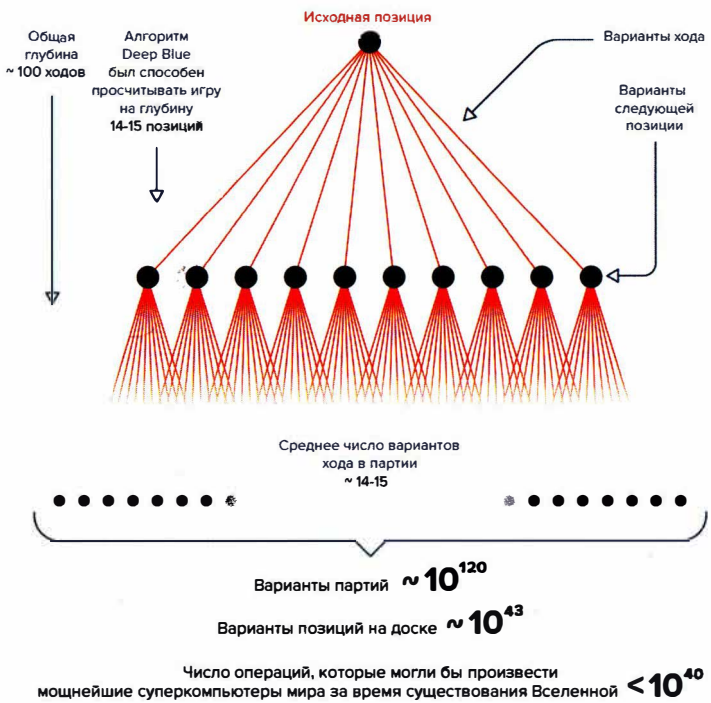


Рис. 1

Дерево вариантов шахматной партии

до 1992 г. и в последней версии назывался «Чего компьютеры все еще не могут».

После «Алхимии» Дрейфуса другой известный философ, Джон Серл, ввел понятие *сильного ИИ* — то есть такого ИИ, который обладает всеми качествами человеческого разума: пониманием, самосознанием, субъективными переживаниями и т.д., — проведя границу между ним и *слабым ИИ*, который таковыми качествами не обладает. Серл полагал, что компьютеры принципиально не способны на сильный ИИ, что и пытался доказать на примере понимания естественного языка в своем мысленном эксперименте — парадоксе «Китайской комнаты», о котором мы поговорим позднее.

Критика возможности реализации сильного ИИ звучала не только от философов. Например, известный математик и физик Роджер Пенроуз утверждал, что в математике существуют алгоритмически неразрешимые задачи, с которыми человеческий интеллект может справиться.

Однако ни одни достаточно конкретные предсказания о том, какие именно задачи компьютер принципиально не способен решать, не сбываются. Компьютер уже победил человека в викторине Jeopardy!, а системы машинного перевода сейчас используются профессиональными переводчиками, которые хоть и вносят правки в вариант перевода, предложенный компьютером, но и учатся у него чему-то новому для себя. Да и работа современного математика без систем помощи в доказательстве теорем вряд ли могла бы быть столь же продуктивной. А лидерство компьютера во всех интеллектуальных играх уже не вызывает сомнения.

Рядом с воззрениями об уникальности человеческого разума и неспособности компьютеров к полноценному мышлению соседствуют популяризованные фантастикой представления о том, что системы искусственного интеллекта и роботы скоро обретут самосознание и в лучшем случае станут просто равноправными членами нашего общества, а в худшем — увидят в нас

угрозу и решат нас поработить или уничтожить. Не слишком ли многого мы ожидаем от машин, не способных, как утверждают некоторые мыслители, к творчеству, пониманию, свободе воли да и просто решению достаточно сложных задач?

Однако предостережения о потенциальных рисках, связанных с ИИ, можно встретить не только в фантастике. И если 15–20 лет назад проблемы безопасности ИИ интересовали лишь небольшое число энтузиастов, то сейчас на эту тему высказываются известные философы, ученые, бизнесмены, проводятся международные конференции, выдаются гранты; она начинает рассматриваться на государственном уровне в разных странах…

Ажиотаж вокруг ИИ также подвергается критике. Специалисты по ИИ давно подчеркивали, что искусственный интеллект — это не мыслящие машины, а, как отмечается, например, в «Толковом словаре по искусственному интеллекту» (1992 г.), «научное направление, в рамках которого ставятся и решаются задачи аппаратного или программного моделирования тех видов человеческой деятельности, которые традиционно считаются интеллектуальными». Фактически ученые сами ограничивались разработкой слабого ИИ.

Это верно и для многих современных успехов в ИИ. Хотя некоторые из них и позиционируются как предвестники мыслящих машин, они автоматизируют лишь отдельные виды деятельности, решают отдельные задачи и не вносят особого вклада на пути к мыслящим машинам. Посмотрите на следующее описание.

> Если поставить на поднос чашку с чаем, тяхакоби-нинге начинает покачивать головой, двигать ногами и перемещаться в сторону гостя, которому предназначен напиток. Когда чай берут с подноса, она останавливается; когда же пустую чашку ставят на поднос, она разворачивается и возвращается назад.

Неплохо было бы иметь дома подобного робота, не правда ли? Не сверхинтеллект, но гостей удивить можно… А ведь это даже не робот — это японская механическая кукла XVIII в.

Робот Эрик мог принимать разные позы, сидеть, стоять, двигаться. А кроме того, он разговаривал — голосом мог отвечать на вопросы, которые ему задавали. Правда, он знал ответы лишь на полсотни заранее заготовленных вопросов, но многие роботы и сейчас умеют не более того. Вот только Эрик был создан в 1928 г. Могли ли люди тогда, глядя на Эрика, подумать о том, что еще чуть-чуть — и роботы обретут сознание? Пожалуй, да. И они бы ошиблись.

Конечно, нельзя отрицать, что с помощью компьютеров удается решать все больше задач и что во все большем числе видов деятельности системы ИИ начинают превосходить человека. Однако действительность, стоящая за громкими лозунгами и красочными описаниями, часто гораздо прозаичнее, так что критика ажиотажа вокруг ИИ небезосновательна.

И все же, что бы ни говорили об ИИ как о приземленной области исследований и разработок по автоматизации решения частных задач, эта область началась именно с мечты о создании мыслящих машин. Поэтому не так удивительно, что со стороны ведущих ученых время от времени звучали призывы вернуть области ИИ ее исконные цели[1]. И за последнее десятилетие не только вернулся академический интерес к этим целям, но и коммерческие организации стали серьезно рассматривать перспективу создания сильного ИИ.

Так где же правда и чего стоит ожидать от разработок в области искусственного интеллекта? И неужели между этими двумя крайностями — или настоящий искусственный интеллект

---

[1] McCarthy J. The Future of AI — A Manifesto. AI Magazine. 2005. V. 26. No 4. P. 39; Brachman R. Getting Back to «The Very Idea». AI Magazine. 2005. V. 26. No 4. P. 48–50; Nilsson N. J. Human-Level Artificial Intelligence? Be Serious! AI Magazine. 2005. V. 26. No 4. P. 68–75.

никогда не появится, или он будет сверхразумной личностью, создающей угрозу существованию человечества, — ничего нет?

Источник разногласий о перспективах и путях создания искусственного интеллекта кроется в интеллекте человеческом. Нейрофизиологи и психологи очень хорошо знают, насколько сложен феномен человеческого мышления, и сама идея воспроизвести его искусственно на компьютере без понимания того, как оно работает у человека, многим кажется нелепой. Технические же специалисты часто говорят, что ИИ может быть похож на человеческий интеллект не более, чем самолет похож на птицу. Можно пойти еще дальше и спросить: а нужно ли было знание биомеханики для изобретения колеса?

Но постойте, почему мы тогда вообще можем говорить о том, что создается именно интеллект? И из каких соображений он создается, если он так сильно отличается от естественного аналога? Самолет создавали авиаконструкторы, и орнитологи в спор об искусственных птицах с ними не вступали. Но самолет создавался с конкретной целью. А в чем же цель систем ИИ? Как отмечалось, эти системы должны решать задачи или автоматизировать ту или иную деятельность.

Давайте на минутку просто отбросим словосочетание «искусственный интеллект» и спросим: а будет ли кто-то утверждать, что компьютер принципиально не способен складывать числа, поскольку не обладает самосознанием? Сейчас это может прозвучать смешно, но ведь для человека это интеллектуальная операция, недоступная в полной мере для животных; еще не столь давно лишь немногие люди умели считать. Когда-то для Блеза Паскаля возможность построить арифметическую машину, способную выполнять эти операции автоматически, была основанием, чтобы высказать идеи о возможности механического воспроизведения человеческого мышления в целом. Сейчас же, беря задачу, алгоритм решения которой известен, мы даже не относим ее к юрисдикции искусственного интеллекта.

> **Если вместо вопроса: «Должен ли искусственный интеллект быть похож на человеческий?» — мы спросим: «Должен ли компьютер решать задачи теми же методами, что и человек?», то уверенно ответить «нет» будет гораздо проще.**

И тогда вопрос, каким образом компьютеру лучше решать такие задачи, будет адресован математикам и программистам, а не нейрофизиологам и психологам, хотя это и не означает, что не стоит вдохновляться решениями, найденными природой.

Казалось бы, мы просто возвращаемся к слабому ИИ. Но заметьте: компьютер решает все более и более сложные задачи лучше человека. Есть задачи, которые для самого человека крайне сложны, — например, из области системной биологии: проблема радикального продления жизни или хотя бы поиска лекарств от некоторых тяжелых болезней. Может ли компьютер в этом превзойти человека? Почему нет? Еще недавно считалось, что уйдут столетия на то, чтобы компьютер превзошел человека в го. А до этого думали, что и в шахматы он никогда с человеком не сравнится. При этом мы не утверждаем, обладает ли такой компьютер самосознанием, пониманием или интеллектом. Он просто решает задачи. Разве этого мало, особенно если это будут жизненно важные задачи, с которыми человек не справляется?

Однако несмотря на реальные успехи ИИ и на то, что многие прогнозы о недоступных для компьютеров возможностях оказались ложными, современные методы ИИ все еще остаются «слабыми». И когда речь заходит о действительно сложных задачах, таких как совершение оригинальных научных открытий или полностью автономное долговременное управление роботом в заранее неизвестной среде, возникают сомнения в том, что технологии слабого ИИ способны их решить, и подспудно появляется мысль, что для этого уж точно нужен «настоящий» разум, с которым тут же ассоциируется самосознание, наделение роботов правами и восстание машин.

Но почему мы решили, что недостаток современных методов именно в их «слабости» и что единственной альтернативой этому является сильный ИИ, возможность создания которого как вызывает сомнения, так и пугает? Если два имеющихся пути не устраивают, то необязательно выбирать из них — можно найти третий.

Мы уже видели, что в прошлом философы не раз ошибались, говоря, что те или иные задачи невозможно решить системами без «сильных» свойств — творчества, понимания, сознания. Возможно, AlphaGo или MuZero обладают «пониманием» игры го и делают «творческие» ходы, но вряд ли в человеческом смысле этих слов. Да и разработчики данных систем вовсе не пытались наделить их творческими способностями и функцией понимания, как и не пытались доказать, что эти способности есть у их детищ. Они просто разработали системы, решающие некоторые задачи лучше человека. Почему это не может оказаться верным и для более сложных задач?

Космические корабли были придуманы вовсе не потому, что самолеты недостаточно похожи на птиц. Но если существующим системам ИИ не хватает не «сильных» свойств, то чего же тогда?

Критику полувековой давности о неспособности решить тысяча первую задачу высказывали в адрес своих творений сами специалисты по ИИ. И это несмотря на то, что в те романтические времена цель создания мыслящих машин ставилась явно и предпринимались попытки разработать системы широкого назначения — такие, как общий решатель задач. С тех пор область ИИ прошла заметный путь и достигла значимых результатов. Однако сейчас эта проблема стала даже более рельефной, чем тогда.

Неоправданные ожидания от создания мыслящих машин привели к тому, что подавляющее большинство работ в области ИИ стало посвящено решению отдельных конкретных задач. Конечно же, это полезно. И конечно, решение

практических задач подспудно приводило к развитию технологий ИИ. Однако каждая конкретная задача наиболее эффективно решается своим частным методом, в идеале — точным алгоритмом, если таковой удается найти. Грубо говоря, компьютер, собирающий кубик Рубика по такому алгоритму, не проявляет ни малейшего интеллекта. Интеллект был проявлен разработчиком, придумавшим этот алгоритм.

Но дело не в том, что компьютер действует по алгоритму, придуманному человеком (и потому якобы не проявляет творчества или интеллекта), а в том, что алгоритм сборки кубика Рубика полезен только для сборки кубика Рубика.

Сейчас есть множество приложений для искусственного интеллекта, но каждое из них требует человеческого труда. И труд этот в основном сводится не к развитию способности компьютера решать задачи, а к изучению предметной области человеком. Так, еще недавно для разработки систем машинного перевода или диалоговых систем требовались целые армии лингвистов, огромный труд которых сосредотачивался на конкретном предмете — языке. Конечно, язык тесно связан с мышлением, и можно было бы предположить, что если таким «ручным» способом наделить компьютер способностью к языку, то это будет большим шагом к мыслящим машинам. Однако оказалось, что с использованием более общих методов машинного обучения, в частности глубокого обучения, небольшие команды разработчиков, не разбирающихся в лингвистике, могут создать системы, которые решают естественно-языковые задачи эффективнее более ранних систем, созданных при участии сотен лингвистов.

Но до сих пор имеющиеся решения не работают «из коробки» для новых задач или просто в новых условиях. Даже достаточно хорошо проработанные детекторы объектов ориентированы на определенный ракурс камеры и при его изменении резко ухудшают качество работы или, например, дают большое число ложных детекций на бликах мокрого асфальта.

Под каждую камеру их приходится дообучать, вручную размечая данные. Специализированность решений проявляется и в том, что, например, системы детектирования и распознавания объектов на изображениях, получения ответов на вопросы по изображениям, генерации описаний изображений или синтеза изображений по описаниям — это все разные системы, и хотя их архитектуры могут иметь отдельные общие компоненты, но обучены они будут по-разному, на разных данных, по разным функциям ошибки. Что уж говорить про системы, работающие на данных другой природы? AlphaGo или MuZero, хоть и способны обучиться играть в разные игры, в отличие от Deep Blue, после обучения под разные игры будут разными системами. А главное, система, обученная играть в го, не сможет без полного переобучения играть не только в шахматы, но и в го на доске другого размера или по слегка измененным правилам (например, в атари-го, где цель — первым захватить хоть один камень).

> **Если мы подумаем об этом в контексте вопроса «Чего не хватает существующим методам ИИ?», то становится очевидным, что их основной технический недостаток не в том, что они не являются сильным ИИ, а в том, что они являются узкими.**

Узость методов ИИ проявляется не только в том, что метод, разработанный под одну конкретную задачу, не может решать другую, даже родственную, задачу. Она проявляется также и в том, что существующий в ИИ инструментарий плохо пригоден для решения «широких» задач. Например, общая система компьютерного зрения должна была бы быть способной анализировать самые разные изображения (без обучения по тысячам размеченных человеком примеров под каждый конкретный случай). Например, не существует ни одной системы компьютерного зрения или искусственной нейросети — несмотря на их бесчисленное количество, — которая могла бы

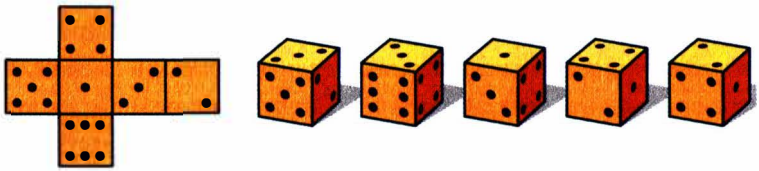


Рис. 2

Какой кубик мог бы получиться после склейки бумажной заготовки?

сходу решить следующую задачу (рис. 2): какой кубик мог бы получиться после склейки бумажной заготовки?

Под конкретный вид кубиков не так сложно натренировать искусственную нейросеть, которая решала бы эту задачу. Но достаточно будет поменять содержание граней кубиков или вместо кубиков взять пирамидки, чтобы решение перестало работать.

Можно придумать неограниченное число новых задач из области зрительного восприятия, которые человеческая зрительная система способна решать на приемлемом уровне без дополнительного обучения. Разумеется, для восприятия и анализа необычных изображений, например рентгенограмм в медицине, обучение потребуется и для человека, а специальная искусственная нейросеть, обученная под очень конкретную задачу, скажем распознавание ранних стадий рака, и ничего больше не умеющая, может это делать лучше человека. Конечно, такие узкие системы нужны, но нужны и системы, способные работать в широких предметных областях.

Разница между узкими и широкими задачами или предметными областями еще яснее видна в робототехнике, где обычно речь идет о степени недетерминированности или неопределенности среды, в которой должен работать робот. Неудивительно, что робототехника долгое время больше всего применялась в промышленности, в которой для роботов можно обеспечить наиболее контролируемые условия. Если роботу

всего лишь нужно сортировать детали из ограниченного перечня на ленте конвейера, то его среда весьма узка. Такого робота не стоит пытаться просить еще и закручивать гайки или даже поднимать детали, упавшие с ленты на пол.

Бытовые роботы стали распространяться гораздо позже, поскольку их среда гораздо неопределеннее. Даже робот-пылесос или робот-газонокосилка, хоть и решают очень конкретные задачи в относительно простых условиях, все же оказываются в заранее неизвестной среде — квартире или доме, которые разработчики никогда раньше не видели. И хотя эти роботы уже вполне полезны, до полной автономности им далеко. Что уж говорить о роботах в еще более разнообразных средах, например о том, чтобы робот мог «хотя бы» сходить в магазин за покупками?..

Итак, проблема с узостью методов ИИ не только в том, что решение новых задач требует определенных усилий от разработчиков, но и в том, что более широкие или сложные задачи оказываются просто недоступными для автоматического решения. Основная масса усилий в современном ИИ направлена на решение узких задач. Здесь достигаются видимые успехи и приносится ощутимая польза. Однако более общие методы и методы решения более сложных задач при этом развиваются мало, хотя польза от них может быть неизмеримо больше. И сами собой общие методы из узких не возникнут — они должны быть устроены по-разному.

Как мы говорили, альтернативой узким методам ИИ является *общий искусственный интеллект*, который не лежит между слабым и сильным ИИ, а просто находится в стороне от них и определяется без отсылки к человеческому интеллекту как *ИИ, способный решать широкий круг задач*.

Общий ИИ представляет собой отдельное направление со своим понятийным аппаратом, подходами, методами, которые лишь частично пересекаются с методами узкого ИИ. Само возникновение этого направления вызвано

неудовлетворенностью исследователей ограниченностью узких методов, их слабой переносимостью на новые задачи, необходимостью вкладывать человеческий интеллект при разработке каждого решения. Все это, а также специализированное внутреннее устройство узких методов, и вызывает впечатление низкой интеллектуальности компьютеров, даже когда они обыгрывают человека в го или Jeopardy! Эти недостатки не устраняются сами собой и требуют самого пристального внимания.

Исследователи AGI дают разные определения интеллекта, которые позволяют выявить специфику данной области; это важно для того, чтобы четче рисовать образ результата. Вот несколько вариантов таких определений:

> **«Общий интеллект — это способность достигать сложных целей в сложных средах».**
>
> **– Бен Герцель**

> **«Интеллект — это способность системы адаптироваться к своей среде, работая при недостаточных знаниях и ресурсах».**
>
> **– Пей Ванг**

> **«Интеллект измеряет способность агента успешно действовать в широком диапазоне сред».**
>
> **– Шейн Легг и Маркус Хуттер**

Эти определения различаются в деталях, но смысл их примерно одинаков. Хотя акцент на «достижении целей в широком диапазоне сред» или «решении широкого спектра задач» может приводить к уклону в сторону конкретных подходов, например обучению с подкреплением (в первом случае) или рассуждениям на основе знаний (во втором), но эти определения могут подразумевать друг друга.

Может быть полезным и явное указание на ограниченность ресурсов (вычислительных, информационных), так как

система, достигающая тех же целей, что и другая, но при использовании меньшего объема исходных данных или тратящая меньше вычислительных ресурсов, должна признаваться более интеллектуальной. И при этом учет ресурсов часто забывают включить в постановку цели как в теории, так и на практике, поэтому лишний раз упомянуть о них не помешает.

Если резюмировать все вышесказанное, общим интеллектом в AGI признается *способность достигать целей в широком диапазоне сред с учетом ограничений* (хотя настаивать на конкретных словах в этом определении не стоит).

Среды могут быть любыми — не только физическими, но и виртуальными, не только пространственно-временными, но и абстрактными. Конечно, может показаться естественным создавать AGI, ориентированный на ту же среду, что и человек в повседневной деятельности. И в этом есть свои плюсы. Можно также утверждать, что существует только одна среда — реальный мир. И это тоже правда. Но реальный мир очень разнообразен. Игра в шахматы и даже любая виртуальная игра являются частями этого мира (а называть их «средами» или нет — вопрос больше терминологический, хотя и имеющий тонкие методологические следствия), и делать акцент на какой-то конкретной его части может быть слишком «узко» и не вполне полезно для глубокого понимания реальности. Например, «наивная физика», которая позволяет нам кататься на качелях или жонглировать, скорее, мешает нам понимать квантовую механику или теорию относительности. Так должны ли мы таким же образом ограничивать искусственный интеллект, если хотим, чтобы он помогал нам решать сложные, в частности научные, проблемы?

Делая акцент на широком диапазоне сред, область AGI позволяет нам избавиться от антропоцентричных предпочтений и предлагает сфокусироваться на общих решениях, пригодных для разных агентов (человека, животных, роботов, ботов и т.д.), действующих в разных условиях — и в микромире,

и на неизведанных планетах, и в виртуальных средах, и в абстрактных (но все же связанных с реальностью) математических пространствах. Если мы зациклимся на конкретном физическом воплощении, есть риск уйти от общего интеллекта в набор специализированных решений, лучше человека приспособленных к некоторому фрагменту реальности, но катастрофически проигрывающих ему даже в тех вопросах, для решения которых человеческий мозг эволюционно явно не предназначался. Разве сможем мы такой ИИ, сколь бы хорошо он ни управлял телом, скажем, андроида, признать разумным? Именно поэтому указание на широкий диапазон сред оказывается столь важным.

Определение интеллекта в области AGI может казаться слишком абстрактным и далеким от наших представлений о естественном интеллекте. А еще широко распространено мнение, что человеческий интеллект — единственный пример интеллекта. Так почему бы не опираться на него? Даже если мы предположим, что для характеристики AGI необходимы или достаточны критерии человеческого интеллекта, описание последнего тоже основано на довольно зыбких понятиях. Вот пример из «Википедии».

> **Интеллект — качество психики, состоящее из способности приспосабливаться к новым ситуациям, способности к обучению и запоминанию на основе опыта, пониманию и применению абстрактных концепций и использованию своих знаний для управления окружающей средой. Общая способность к познанию и решению проблем, которая объединяет все познавательные способности: ощущение, восприятие, память, представление, мышление, воображение, а также внимание, волю и рефлексию.**

Тут упоминается созвучная AGI «общая способность к ... решению проблем», но при этом указывается на то, что эта способность объединяет ряд других способностей, включая,

например, мышление и волю, понимание и обучение и т.д. При этом не очень понятно, как эти способности можно обособить друг от друга. Например, наша память тесно связана с эмоциями, оценкой информации и другими функциями, а воля зависит от множества факторов (начиная от природной чувствительности к вознаграждениям и издержкам и заканчивая умением разбивать волевую деятельность на маленькие шаги и управлять своим вниманием). Должны ли мы разделять эти функции у ИИ? В какой степени такие способности должны быть предустановлены, а в какой мы позволим агенту их «отращивать» по ситуации, для лучшего решения задач в конкретной среде? Определение интеллекта в области AGI не включает эти способности, но и не отвергает их. При этом оно побуждает нас задуматься о том, зачем нужен тот или иной компонент интеллекта и как это зависит от свойств среды и условий задачи. Здесь акцент ставится на том, что мы хотим построить, а не на том, каким именно образом мы хотим это сделать. Цель отделяется от способа ее достижения так же, как хороший заказчик, составляя ТЗ, формулирует именно задачу, не навязывая способ ее решения, на предмет которого заказчик может и ошибаться.

В действительности даже специалисты по AGI ставят перед собой разные цели, именно потому их определения интеллекта несколько различаются. Кому-то предпочтительнее, чтобы система решала задачи в масштабе реального времени, пусть даже и не очень хорошо. Кому-то важнее решение сложных задач, пусть даже на них могут уйти годы. Стоит особо подчеркнуть, что это не столько разное понимание некоего объективного феномена интеллекта, сколько постановка несколько разных целей, и нельзя сказать, что одни цели лучше или хуже других. Однако сила концепции AGI — в ее самоприменимости: в конце концов, общий интеллект должен быть способен достигать разных целей, даже если это цели по созданию общего интеллекта с разным уклоном. Так что именно эта

общность оказывается ключевой особенностью интеллекта, точкой самоприменимости.

Кроме того, некоторые человеческие способности предполагают не только объективную внешнюю оценку, но и субъективное внутреннее ощущение (например, понимание, воображение, самосознание и т.д.). Когда мы говорим о сильном ИИ, мы подразумеваем, что он должен быть наделен всеми человеческими качествами (и обратной стороной медали тут могут быть человеческие слабости, например психические расстройства[2]). Но для общего ИИ это необязательно.

Как повышение уровня решения узких задач вплоть до сверхчеловеческого не потребовало «сильных» качеств, так и расширение общности методов решения задач вовсе не обязательно подразумевает преднамеренное движение в сторону сильного ИИ. Можно предположить, что некоторые аналоги некоторых «сильных» качеств у действительно общего ИИ должны быть. Например, наверняка общий ИИ должен иметь способность к интроспекции — анализу собственных мыслительных процессов или даже оптимизации лежащих в их основе алгоритмов. При этом у такого ИИ будет некий образ себя как часть картины мира. Но это не обязательно означает, что у него будет самосознание в философском смысле и уж тем более личность сродни человеческой, хотя по глубине рефлексии он вполне может и превосходить человека. Наверняка он проявит

---

[2] Есть теории о том, что разные психические расстройства — это следствие чрезмерной адаптации (как генетической, так и через опыт) к очень специфической среде, приводящей к дезадаптации при смене условий. Например, повышенная чувствительность к издержкам, которая в перспективе может привести к депрессии, имеет смысл в среде, где мало возможностей и много рисков. А синдром дефицита внимания, по одной из гипотез, мог быть вполне функциональным состоянием для первобытного охотника, которому было важно быстро переключаться между разными сигналами, чтобы заметить добычу или опасного хищника. А вот с развитием земледелия, требовавшего кропотливой рутинной работы, он стал приводить к дезадаптации.

«понимание» тех областей, в которых действует успешнее человека. Но это не значит, что такое понимание будет сопровождаться у него субъективными переживаниями, схожими с человеческими. Наверняка у него будет многомерная система мотивации, включающая аналоги, например, любопытства и удивления. Но его вовсе не обязательно пытаться наделить всеми человеческими эмоциями. Хотя, скажем, для социальных роботов это может быть полезно, но даже они способны лишь симулировать чувства и эмоции, а не испытывать их.

Итак, идея общего ИИ предполагает, что компьютеры смогут самостоятельно решать как новые узкие, так и сложные задачи, чем будут заметно отличаться от критикуемых систем ИИ, но способ, которым компьютеры будут это делать, может быть далеким от человеческого. А называть ли этот способ интеллектом в некоем обобщенном смысле — вопрос договоренности.

Поскольку фактически на настоящий момент систем AGI не существует, для характеристики систем, разрабатываемых в рамках данной области, вводятся такие понятия, как proto-AGI и Narrow AGI. Под proto-AGI имеются в виду системы, призванные решать широкий круг задач, но все еще не способные делать это эффективно. При этом обычно подразумевается, что эти системы могут быть со временем доработаны до AGI или, по крайней мере, являются шагом на пути к нему. Термином Narrow AGI обозначаются не существующие пока системы, обладающие общим интеллектом, но демонстрирующие (сверх)человеческий уровень в одной предметной области, оставаясь существенно ниже уровня человека во всех других сферах (то есть сходные с людьми-савантами). Предполагается, что такие системы могут быть промежуточным шагом на пути к полноценному AGI.

Глава 1.
ОБЩАЯ ИСТОРИЯ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА

Перед тем как переходить к возможностям общего ИИ и потенциальным способам его создания, стоит вспомнить, как эволюционировали подходы к искусственному интеллекту в целом и какие уроки мы можем из этого почерпнуть.

В августе 1955 г. четыре знаменитых исследователя, среди которых были создатель теории информации Клод Шэннон и молодой математик Марвин Минский, написали грантовую заявку. Они хотели получить деньги и организовать двухмесячный семинар, посвященный искусственному интеллекту. Они надеялись, что, если лучшие специалисты объединят свои усилия, можно будет существенно продвинуться в том, чтобы научить машину понимать человеческий язык, оперировать абстрактными концепциями, самообучаться и мыслить креативно. Поставленные в заявке задачи остаются актуальными и сейчас, а знаменитый Дартмутский семинар, организованный Шэнноном и Минским, признан колыбелью всей отрасли искусственного интеллекта. В последнее время исследователи ИИ приближаются к тем целям, которые когда-то поставили

основатели отрасли и от которых отрасль впоследствии заметно отошла.

Историю отрасли можно рассказывать как линейную последовательность событий, но это неизбежно будет неточным. В первую очередь потому, что на протяжении всего времени существования искусственного интеллекта его понимание и содержание остаются предметом оживленных дискуссий среди исследователей. Параллельно существуют разные школы мысли и разные подходы к разработке ИИ, в частности потому, что даже естественный интеллект до сих пор не имеет общепринятого определения. Хуже того, многие задачи и технологии, прежде относившиеся к области ИИ, теперь к ней не относятся, например полностью решенные игры типа шашек, методы символьного дифференцирования или роботы с программным управлением. Сама планка сложности задач и методов, относимых к области ИИ, постепенно повышается.

> **Как это часто случается, представители различных школ, работающие на протяжении последних 60 лет в различных парадигмах, по-своему трактуют области применения, задачи, определения и перспективы ИИ, считая альтернативные подходы неудовлетворительными.**

Мы можем назвать задачи, которые решаются современными методами ИИ, но не можем однозначно соотнести текущий уровень развития с рубежом, после достижения которого сможем утверждать, что полноценный ИИ создан. Поэтому до сих пор нельзя объективно сказать, каков уровень развития ИИ и каковы перспективы его совершенствования. Необходимые направления исследований не могут быть отделены от субъективных мировоззренческих позиций исследователей и практического опыта разработчиков. Тем не менее в развитии области, как и в последовательной смене господствующих в ней парадигм и метафор, можно проследить определенную логику.

Подходы и методы

Первой доминирующей парадигмой в ИИ была парадигма *мышление как поиск*. Как и многие последующие подходы, она была построена на представлении об устройстве естественного интеллекта, а точнее, на лабиринтной гипотезе мышления, заимствованной из психологии.

Поставьте неожиданное препятствие на пути муравья, идущего привычным маршрутом, или курицы, бегущей домой через знакомую дырку в ограде. Их поведение станет хаотичным. Они будут метаться по сторонам и тыкаться в любое подходящее отверстие или проход. И это поведение бессмысленно только на первый взгляд. Если проверенный путь закрыт, они будут искать новые выходы из ситуации, перебирая случайные варианты. Курица осуществляет этот поиск в физическом пространстве, а человек или шимпанзе способны осуществлять его в голове, но для каждого из них мыслительная деятельность

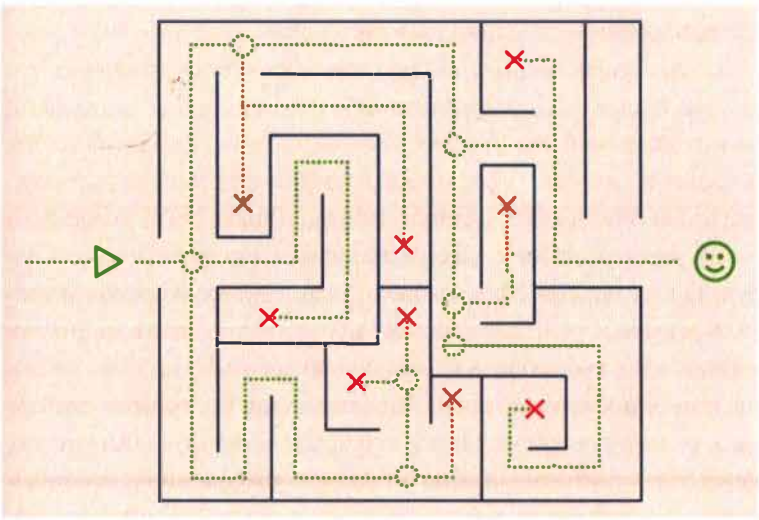


Рис. 3

Мышление как поиск

может возникать как поиск в ответ на незнакомую ситуацию. Ровно так и возникает метафора мышления как поиска: суть интеллекта состоит в решении проблем, а сам процесс решения может быть представлен как поиск пути от исходных данных к ответу в пространстве — «лабиринте» — возможных решений (или как поиск пути от имеющихся средств к конечной цели через достижимые подцели).

Этот подход начали применять для решения интеллектуальных игр и доказательства теорем, где концепция поиска также играла ключевую роль. А для сокращения времени поиска использовали эвристики — простые правила, позволяющие сократить перебор возможных вариантов. Правило может быть, например, такое: «В шахматах следует отдавать предпочтение тем ходам, которые позволяют наращивать преимущество в силе фигур». Или такое: «При игре в крестики-нолики все угловые клетки на первом ходу симметричны и достаточно проанализировать возможность хода только в одну из них». Поэтому важнейшей технологией в рамках этой парадигмы стало эвристическое программирование.

Со временем, однако, обнаружилось важное ограничение подобных систем. Формализованное описание задачи для каждой из них должно было составляться человеком. Кто-то из плоти и крови должен был сформировать лабиринт, внутри которого могла бы блуждать машинная система. Более того, не удалось найти универсального алгоритма, эффективно решающего любую задачу поиска. Хотя какие-то общие методы поиска и удавалось выработать, для каждой задачи нужно было изобретать эвристики и кодировать их в эвристической программе на понятном для машины языке. Каждая новая программа поиска писалась практически с нуля, что представлялось, конечно же, неудовлетворительным. Шагом на пути устранения этого недостатка стало отделение машины поиска от предметных знаний.

Следующая идея, доминировавшая в области ИИ на рубеже 1960–70-х гг., прошла под лозунгом «Знание — сила». Знания

перестали неявно закладываться в программный код. Вместо этого для них стали разрабатываться разнообразные собственные способы представления, обладающие ясной структурой и смыслом. Неудивительно, что большую роль здесь играли логические представления, содержащие знания о предметной области в виде логических выражений, из которых осуществлялся дедуктивный вывод. А в 1972 г. появился язык программирования «Пролог», который сделал разработку систем на основе знаний доступной и для неспециалистов в области ИИ. Но логическими представлениями все не ограничилось. Из компьютерной лингвистики, которая также бурно развивалась в то время, были заимствованы обретшие впоследствии большую популярность семантические сети. Их идея была разработана еще в 1956 г. в рамках проекта по изучению языка в целях машинного перевода, но теперь к ней добавилась способность машины рассуждать на основе представленных в семантических сетях знаний. Возникли и другие представления — фреймы, продукции, сценарии и т.д.

Ключевой технологией здесь стали *экспертные системы*, которые одно время являлись чуть ли не синонимом понятия «искусственный интеллект». Это программы, обладающие экспертными знаниями в какой-то отдельной предметной области и использующие их для решения конкретных задач — вроде системы, которая знает устройство электростанции, или налоговое законодательство, или симптомы заболеваний сердечно-сосудистой системы и пользуется этими знаниями, чтобы предупреждать поломки генераторов, или оптимизировать налоги, или ставить диагнозы.

Экспертные системы оказались способными находить и объяснять решения задач, сформулированных на ограниченном естественном языке для узкой области. При этом поиск решения в таких системах превратился в проблему рассуждения на основе знаний, в котором усматривалась суть мышления. Узким местом при создании таких систем

Правило 1: «Если покрыто шерстью, то млекопитающее».

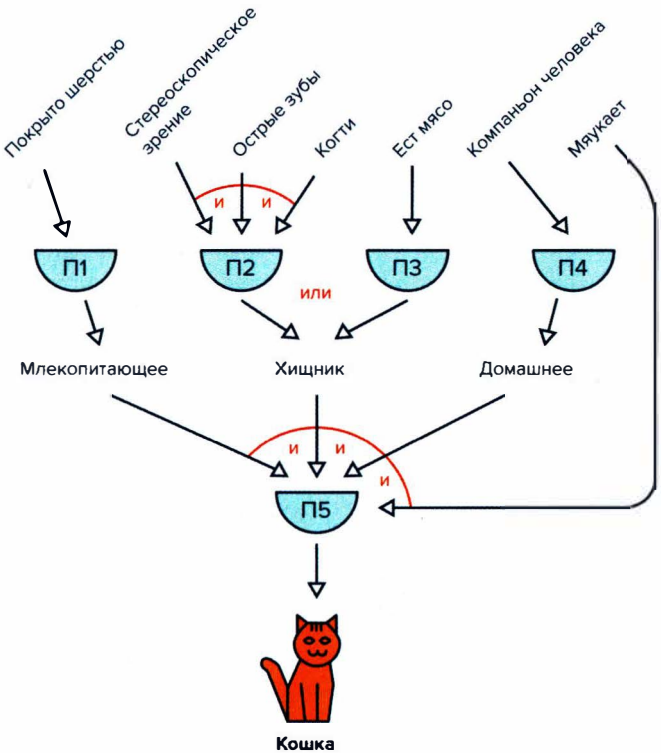


Рис. 4

Как работает «экспертная система»

оказалось извлечение знаний, то есть наполнение базы знаний фактами и правилами о предметной области, на основе которых система уже могла бы, рассуждая, приходить к ответу на ставящиеся перед ней вопросы. Процесс извлечения знаний выполнялся инженерами по знаниям, работа которых заключалась во взаимодействии с предметными экспертами в попытке формализовать их опыт. Но этот опыт нередко был очень обширным и не до конца осознаваемым самими экспертами, так что наполнение баз знаний вручную оказалось

очень трудоемким и ограничивало широту и глубину владения предметом.

Тогда как правила, по которым рассуждают люди, выявить сложно, принятые ими решения напрямую доступны, так что в рамках экспертных систем естественным образом возникла задача автоматического извлечения знаний из наборов примеров, что оказалось не чем иным, как частным случаем машинного обучения[3]. Хотя область машинного обучения в применении к распознаванию образов существовала с середины прошлого века, она не оказывалась в фокусе внимания многих исследователей ИИ, считавших интеллектуальными только задачи, решение которых было доступно исключительно человеку. Но системы, основанные на знаниях, были слишком зависимы от человека, и в результате в 1980-е годы машинное обучение превратилось в центральную парадигму ИИ, обещавшую эту зависимость ослабить. И хотя технологии машинного обучения для экспертных систем заметно отличались от более традиционных методов, на передний план вышла сама проблематика обучения в целом.

Методы машинного обучения не смогли избавиться от зависимости от человеческого участия. Они требовали рафинированных данных — простых по своей структуре, предобработанных и специальным образом размеченных. Ведь в отличие от человека, способного ориентироваться в контекстах и отсекать ненужные детали, наделить такой способностью машину непросто, а без этого машина воспринимает всю информацию буквально. В итоге методы машинного обучения зачастую работали по подготовленным человеком данным, а не добывали эти данные сами при взаимодействии с окружающим миром.

---

[3] Класс методов искусственного интеллекта, характерной чертой которых является не прямое решение задачи, а обучение в процессе применения решений множества сходных задач.

Дополнением к описанным методам стал агентный, или мультиагентный, подход, снискавший популярность в конце 1990-х. Агентов программируют так, чтобы они были по возможности автономны, в том числе при получении данных для обучения, и демонстрировали адаптивное поведение в некоторой среде обитания. Среди прочего, они должны достигать своих целей в среде, взаимодействуя с другими агентами, помещенными туда же. Вместе с агентным подходом развивались и другие «автономные» направления, например обучение с подкреплением, аниматы или эпигенетическая робототехника.

> **Обучение с подкреплением — это такая постановка задач машинного обучения, где у нас нет ни учителя, ни меток, обозначающих правильные ответы. Агент учится через взаимодействие со средой: он может выбрать действие из определенного набора, а среда в ответ на это действие как-то меняется.**

Агент воспринимает результат этого изменения, получая при этом также сигнал подкрепления (которое может быть как положительным, то есть вознаграждением, так и отрицательным — наказанием). Задача агента — получить максимальное суммарное вознаграждение в ходе длительных взаимодействий со средой.

К примеру, если агент — это система обучения с подкреплением, управляющая автомобилем в виртуальных гонках, то подкрепление для нее будет основываться на том, на какое расстояние по трассе и за какое время смог проехать автомобиль. Исходно агент будет совершать случайные действия и далеко не уедет. Но, получая вознаграждения за безаварийное движение в нужном направлении с большей скоростью, система будет подбирать все более подходящие действия, постепенно улучшая свою стратегию управления автомобилем.

Аниматы[4] получили известность после конференции «Симуляция адаптивного поведения: от животных к аниматам», которая прошла в Париже в 1990 г. Это автономные искусственные организмы, которые способны существовать в меняющейся среде и решать в ней какие-то базовые задачи так же, как любые живые организмы. Важно, что их создатели пытаются решить не бионические задачи (например, имитировать работу мышц), а задачу воспроизведения базовых принципов поведения животных. Допустим, у нас есть анимат, имитирующий одноклеточный организм. Он перемещается по ровной поверхности в поисках пищи. Но пища на ней возникает непредсказуемым образом. Ее то много, то мало, то какое-то время вообще нет. Она может лежать спокойно или уклоняться от анимата. Обучение с подкреплением часто рассматривается в применении к конкретным практическим задачам, где требуется один раз выработать стратегию поведения для определенных условий. В таких случаях возникает большой соблазн сделать упрощенный, специализированный метод. При исследовании же аниматов предпочтение отдается адаптации к неизвестным и потенциально меняющимся условиям среды.

Эпигенетическая робототехника идет еще дальше в стремлении создать агента, который имитировал бы уже не животное, а, скорее, ребенка. Дети часто не способны хорошо выполнять даже простые задания, но зато умеют сами добывать информацию о мире, взаимодействовать с окружающей средой, обучаться и идти к своей цели методом проб и ошибок. Такой агент должен проходить несколько стадий в своем развитии, взаимодействуя со средой, при этом обучение должно быть накопительным и сложность задач должна прогрессивно возрастать.

В начале 2000-х в том числе в рамках агентного подхода возникла потребность интегрировать разные подобласти ИИ

---

[4] Термин «анимат» происходит от сочетания слов *animal* + *robot*.

(компьютерное зрение, машинное обучение, обработка естественного языка и т.д.), которые в 1990-х развивались сравнительно автономно и начали достигать немалых результатов, в единые системы.

В качестве одного из подходов в этом направлении рассматривались когнитивные архитектуры для систем ИИ уровня человека или систем, обладающих общим интеллектом. Хотя именно эта парадигма стала самым популярным направлением в зародившейся области общего ИИ, она так и не стала главенствующей в области «обычного» ИИ.

Вместо нее на передний план вышло глубокое обучение — совокупность методов машинного обучения, основанных на обучении многослойным, или иерархическим, представлениям информации, на каждом последующем уровне которых выучиваются все более абстрактные свойства исходных объектов. Оно в значительной мере устраняло другую зависимость методов машинного обучения от человека, а именно необходимость инженерии признаков (то есть создания информативных признаков, коррелирующих с выбранной целевой переменной). Инженерия признаков сродни инженерии знаний, но уже не в экспертных системах, а в системах самого машинного обучения, где экспертами выступают сами специалисты по машинному обучению, анализу данных, распознаванию образов. Как и инженерия знаний, инженерия признаков оказывается весьма трудозатратной и не достигающей идеального результата.

Глубокое обучение не сделало компьютер полностью самостоятельным: оно заменило инженерию признаков на инженерию архитектур интеллектуальных систем, в частности нейронных сетей, которые реализуют определенные аспекты разумного поведения (например, обучение, память, планирование и т.д.). Инженерия архитектур, в свою очередь, может быть автоматизирована методами поиска архитектур,

автоматического машинного обучения (AutoML) и метаобучения[5].

> **При всем разнообразии подходов и методов, использовавшихся в последние 60 лет параллельно, вместе и по отдельности, у них есть общий лейтмотив. Каждая последующая парадигма ИИ была призвана снизить зависимость интеллектуальных систем от человека.**

Общий искусственный интеллект естественным образом продолжает эту тенденцию, но подходит к проблеме более системно.

Это относится и к некоторым другим вопросам. В частности, традиционные для всей области ИИ, но развивавшиеся во многом независимо символьный и коннекционистский подходы переосмысляются в контексте общего ИИ.

Символьный подход берет свое начало в философии, логике и математике и оперирует логическими правилами, знаковыми и символьными системами, которые интерпретируются в терминах высокоуровневых мыслительных процессов человека. Суть в том, чтобы воспроизвести сознательное мышление человека, абстрагируясь от нейрофизиологических деталей его реализации в мозгу (условно, вместо самолета, хоть сколько-то напоминающего птицу, построить вертолет или ракету).

Коннекционистский, или субсимвольный, подход предполагает создание моделей, воспроизводящих не только

---

[5] Метаобучение (англ. meta-learning) — это «обучение обучению», то есть улучшение существующих или поиск новых, более эффективных методов обучения в процессе самого обучения. Данный термин применяется как в когнитивной психологии, так и в машинном обучении. В последнем случае метаобучение, как правило, выполняется на наборе задач. На их примере нужно построить такой алгоритм, который бы умел эффективно обучаться решению не только этих, но и новых сходных задач.

информационную сторону мышления, но и способ его «аппаратной» реализации в мозгу. Самый яркий пример — нейросети.

На протяжении 65 лет оба подхода динамически развивались, иногда двигаясь в разные стороны, а иногда друг к другу. В XXI в. они чаще существуют вместе, чем по отдельности. Четкая бинарная логика сейчас характеризуется как «старый добрый ИИ» (good old-fashioned AI, GOFAI) и занимает сравнительно небольшую нишу, давно уступив лидерство вероятностным методам. В свою очередь, и биологически правдоподобные искусственные нейронные сети привлекают гораздо меньше внимания, чем модели, создатели которых не заботились о биологическом правдоподобии и которые в большей степени опираются на математические методы. Но это разделение все еще остается удобным, потому что оно вызвано в том числе характером исходных данных и способом их представления — дискретным или непрерывным[6], — а также вытекающими из них методами вывода и оптимизации.

Удачи, неудачи и рождение общего ИИ

Разнообразные подходы к разработке ИИ, описанные нами выше, существовали параллельно. А скептически настроенные наблюдатели даже утверждают порой, что принципиально новых подходов не появлялось после 1960-х гг. и менялись лишь

---

[6] Если некоторая величина, несущая информацию, в пределах заданного интервала может принимать любое числовое значение (то есть между разными вариантами значений нет разрывов), то она называется непрерывной, а если она способна принимать только ограниченное (или, строго говоря, не более чем счетное) число значений, она называется дискретной. Образно можно представить непрерывную величину как градиент, а дискретную — как радугу, состоящую из семи цветов без полутонов.

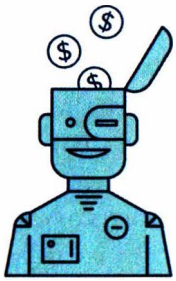
относительная успешность и популярность подходов и локальные темпы их развития. Но нельзя отрицать, что за последние 65 лет исследователи достигли большого прогресса в решении самых разных, пусть и частных задач, а область ИИ из предмета академического интереса превратилась в огромную отрасль. За это время системы ИИ научились обыгрывать людей в шашки, шахматы, го и множество других игр. Освоили перевод текстов на достаточном уровне, чтобы приносить пользу людям. Стали широко использоваться в разных отраслях экономики.

Одним из пионеров систем ИИ стал, например, «общий решатель задач» — система, созданная в 1959 г. Гербертом Саймоном, Клиффордом Шоу и Алленом Ньюэллом. Общий решатель был построен на чисто символьной логике и умел решать ряд задач, включая некоторые математические и шахматные задачи и головоломки, такие как, например, «Ханойская башня». В 1960–70-х гг. были и другие попытки создать системы ИИ общего назначения, но эта цель начала выглядеть слабодостижимой, хотя в решении более частных задач наблюдался определенный прогресс. Исследователи и инвесторы чувствовали разочарование, которое в середине 1970-х вылилось в снижение интереса и падение финансирования. Этот период назвали «первой зимой искусственного интеллекта»[7].

Следующий виток интереса к разработке могущественных систем ИИ возник в 1980-е. Тогда, в частности, огромный проект по разработке искусственного интеллекта, «компьютер пятого поколения», финансировался в Японии. Его целью было создание «эпохального компьютера» с производительностью суперкомпьютера, который бы переводил устную речь в текст, распознавал образы и решал другие сложные для искусственного интеллекта задачи. Проект потерпел неудачу (за десять лет на разработки было истрачено более 50 млрд иен, при этом

---

[7] В дальнейшем спад интереса к ИИ повторился в конце 1980-х — начале 1990-х гг. после падения популярности экспертных систем.



* Данные за 2019 г. приведены за период с января по октябрь. Источник: AI Index 2019

Рис. 5

Инвестиции в отрасль искусственного интеллекта 2009–2019 гг., млрд долларов США в год

поставленные цели не были достигнуты), и для многих это означало очередное доказательство невозможности построения общего ИИ в ближайшем будущем. Но, как замечает современный энтузиаст общего искусственного интеллекта Бен Герцель, проект не опирался на ясные концептуальные представления о том, как должно быть реализовано мышление в компьютерах. Так что проблема была, скорее, в подходе, а не в нереалистичности самой задачи.

Тогда же, в 1980-е, калифорнийский философ Джон Серл придумал ныне знаменитый мысленный эксперимент под названием «Китайская комната». Представим себе изолированную комнату, в которой находится сам философ, который не знает китайского языка и не понимает ни одного китайского

иероглифа. Но у него есть книга с точными инструкциями о том, как реагировать на входящие сообщения на китайском (например, «Возьмите такой-то иероглиф из корзинки номер один и поместите его рядом с таким-то иероглифом из корзинки номер два»). Что все эти иероглифы означают, не сообщается, так что философ просто бездумно следует инструкции.

Наблюдатель, знающий китайский, через щель передает в комнату вопросы, записанные иероглифами. Может ли он получить осмысленный ответ? Ньюэлл и Саймон на основе своего успешного опыта с общим решателем задач выдвинули в 1976 г. гипотезу физической символьной системы, согласно которой манипуляции с символами являются необходимым и достаточным средством реализации основных интеллектуальных операций физической системой. Фактически «Китайская комната» построена как физическая символьная система. Серл говорил: предположим, что гипотеза Ньюэлла — Саймона верна, тогда комната может быть организована так, что будет обладать интеллектом, а значит, она сможет, например, пройти тест Тьюринга, то есть будет способна вести осмысленный диалог. Но посмотрите — человек в комнате совершенно не понимает китайского. Как он может вести осмысленный диалог?

Философ видел в этом противоречие. Он полагал, что раз человек не понимает вопросов — а понимание необходимо для поддержания осмысленного разговора, — то никакие тонкие умения манипулировать символами не приведут машину к настоящей разумности, то есть гипотеза физической символьной системы неверна. Стоит отметить, что данная цепочка рассуждений содержит изъяны. В частности, физической символьной системой здесь является не человек, а комната. Мы можем уверенно утверждать, что вопросы не понимает человек. Но вот не понимает ли их вся комната целиком? Представим себе, что перекладывать таблички будет не один человек (ведь для формирования оперативного ответа могут потребоваться миллиарды символьных операций), а большое

множество людей, которые будут передавать таблички соседям в соответствии с собственными инструкциями. Чем это будет принципиально отличаться от мозга человека, в котором миллиарды нейронов передают друг другу сигналы, при этом каждый из нейронов в отдельности не понимает вопросов, но вся система в целом — понимает? Разница в том, как эти системы сформировались. Построить «китайскую комнату» вручную крайне трудозатратно, и даже если это сделать, сможет ли она дальше взаимодействовать с миром и обучаться сама? Вряд ли. Но именно поэтому начиная с 1980-х гг. на передний план вышли вопросы обучения и воплощенности.

Стоит особенно подчеркнуть, что Серл критиковал возможность реализации сильного ИИ на базе компьютеров в целом, а не просто символьный подход в ИИ, поскольку именно компьютер и является той физической символьной системой, о которой говорили Ньюэлл и Саймон. Будь то коннекционистская или обучаемая система, будь то система, наделенная сенсорикой и моторикой, но если она реализуется на компьютере, то является физической символьной системой и подпадает под критику Серла. С этой критикой многие исследователи не согласны, но даже отвергая ее, доказать, что компьютер способен на «понимание» (в человеческом смысле) или обладает другими «сильными» свойствами, что компьютер может мыслить по образу и подобию человека, — весьма проблематично.

Но давайте вспомним, что участники Дартмутского семинара не связывали искусственный интеллект с нашим пониманием интеллекта человека. Они допускали, что ИИ будет использовать методы, которыми не пользуется человек, а интеллектом считали вычислительную составляющую способности достигать целей в мире. В течение 50 лет после семинара этот посыл часто игнорировали, а потом и вовсе стали все чаще пользоваться понятиями «сильного» и «слабого» ИИ:

Но возьмем шахматный компьютер Deep Blue, выигравший у сильнейшего шахматиста-человека. Вряд ли кто-то

будет настаивать, что Deep Blue — «настоящий ИИ». Но чего ему не хватает? Самосознания? Свободы воли? Если Deep Blue будет себя осознавать и сможет отказаться от очередной партии в шахматы, поскольку ему стало скучно, назовем ли мы его по-настоящему разумным? В рамках концепции «сильного ИИ» мы должны были бы ответить «да» на эти вопросы, но это нам бы не помогло.

> **В 1997 г., как раз тогда, когда Deep Blue обыграл-таки чемпиона мира в шахматы, появился термин «общий искусственный интеллект». В рамках этой идеи предлагалось делить ИИ не на слабый и сильный, а на узкий, способный решать определенные задачи, и общий, действующий в разных обстоятельствах и адаптирующийся к разным проблемам.**

В 2007 г. прошла первая конференция, посвященная исключительно проблемам развития общего ИИ, и с этого времени можно наблюдать возрождение интереса к масштабной проблеме искусственного интеллекта, которой посвящена эта книга. Исследователи новой волны готовы принимать разные подходы к возможному решению проблемы, и как минимум часть из них признают, что компьютер не обязан воспроизводить человеческое мышление во всех его проявлениях, чтобы считаться интеллектуальным.

Состояние дел в области разработки искусственного интеллекта

Как отмечалось ранее, технологии ИИ постоянно развивались в направлении снижения их зависимости от человека при создании решений для новых задач и сейчас находят очень

широкое применение. Тем не менее текущее состояние дел в области, далекое от решения большой задачи, можно охарактеризовать следующим образом.

Во-первых, с точки зрения адаптивности все существующие системы ИИ на основе любых известных подходов предполагают функционирование в ограниченном наборе задач и условий и не способны самообучаться функционированию в условиях существенно новых. То есть они являются программируемыми, хотя сложность их программирования заметно снизилась, а способность к обучению возросла.

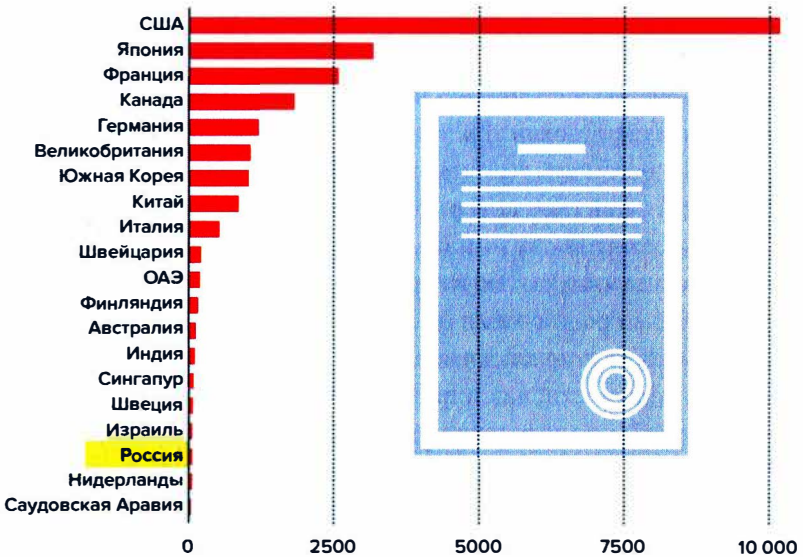
Во-вторых, с точки зрения автономности систем ИИ все существующие системы не являются автономными и не могут полноценно функционировать без живого оператора, отвечающего за запуск и остановку, техобслуживание, целеполагание и определение режимов работы в зависимости от тех или иных условий или задач. То есть они остаются управляемыми. Особенно это очевидно в областях, характеризуемых высокими рисками или высокой априорной неопределенностью (даже робот-пылесос без помощи человека не проживет и пары дней в нормальной квартире).

В-третьих, с точки зрения интегративности современные системы ИИ являются не системами, обладающими интеллектом как таковым (даже ограниченным), а системами компьютерного зрения, обработки естественного языка, анализа данных (машинного обучения), обработки символьной информации (рассуждений на основе знаний) и т.д., то есть интегративными не являются.

В этом смысле нерешенные пока в области ИИ задачи создания автономного и широко адаптивного поведения, реализуемого универсальными самообучающимися агентами, обладающими достаточно эффективными для этого интегративными архитектурами, могут быть отнесены к проблематике общего ИИ (AGI).

Помимо названных концептуальных проблем, есть и частные. Решение многих более узких, но практически важных задач остается недоступным для существующих методов искусственного интеллекта. Так, например, семантический анализ текста на естественном языке и семантический анализ изображений — задачи, которые будут затронуты в разделе, посвященном метрикам, — находятся на таком уровне развития, что даже в рамках узкого ИИ требуется большой объем исследований для продвижения вперед. Для решения этих задач необходимо развитие как символьных и субсимвольных техник ИИ по отдельности, так и подходов, сочетающих в себе сильные стороны каждого из направлений, что требует не только прикладных, но и фундаментальных исследований.

В России развита сильная инженерная база по адаптации, комбинированию, применению существующих технологий машинного обучения, в том числе основанных на искусственных нейронных сетях. Есть сильные научные математические школы, направления исследований которых относится к основаниям методов ИИ и непосредственно могут быть использованы для анализа существующих методов машинного обучения и разработки принципиально новых, перспективных подходов. В то же время у нас в стране, к сожалению, нарушена коммуникация между математическим и инженерным сообществом в области искусственного интеллекта, и поэтому усилиям специалистов не хватает системности. Обычно это либо инженерные разработки с использованием модификаций существующих, преимущественно зарубежных, технологий, либо фундаментальные теоретические работы, которые не всегда доведены до воплощения. Полный цикл работ — от формализации задач, разработки соответствующих математических моделей, фундаментальных исследований их свойств до изучения аспектов алгоритмической сложности и инженерной реализации решений задач — встречается в российской практике довольно редко, что очень печально: аналогичная ситуация в свое время привела



Источник: AI Index 2019

Рис. 6

20 стран, где было выдано больше всего патентов
в области искусственного интеллекта (всего штук за 2015–2018 гг.)

к критическому отставанию в области электроники. Очень не хотелось бы, чтобы то же самое случилось и с ИИ.

Сложность еще и в том, что яркие недавние примеры практического использования нейронных сетей и быстрый эффект от их внедрения создают у дилетантов впечатление, что мы уже без пяти минут как создали всесильный искусственный интеллект. Достаточно привести заголовки новостей: «Специалисты “Студии Лебедева” продемонстрировали нейросеть, которая генерирует логотипы для экспресс-дизайна»; «В России создана нейросеть, которая пишет неотличимые от настоящих комментарии в интернете»; «Искусственный интеллект помогает регионам России бороться с коронавирусом» и т.д.

Поэтому значимость фундаментальных исследований недооценивается. Компании, в свою очередь, ориентированы

на продуктовую разработку с коротким циклом, чтобы быстро получить сверхэффект и заработать денег. Современные образовательные программы готовят инженеров, умеющих работать с существующими технологиями, а не творчески мыслящих исследователей, способных совершать новые прорывы.

В то же время перед нами стоит и широкий круг практических задач, решение которых до сих пор непосильно для существующих методов ИИ. По мере развития моделей машинного обучения становятся очевидными ограничения и уязвимости существующих подходов. Они внезапно проявляются в виде побочных эффектов именно на этапе экспериментов, которые стали основополагающей мерой качества моделей ИИ. Не отрицая практическую значимость эксперимента и успешные результаты, полученные методом проб и ошибок, важно отметить, что перед разработчиками ИИ стоят серьезные вызовы, которые требуют создания новых подходов и технологий. В этой связи критическим для продвижения в данной сфере является формирование условий для воспроизводства специалистов, которые владеют основательной базой математики и компьютерных наук для создания перспективных технологий искусственного интеллекта.

Сверхинтеллект

В массовом воображении разработки ИИ связаны с надеждами или опасениями, что общий ИИ будет создан. Некоторые исследователи и разработчики действительно полагают, что все технологии, необходимые для создания общего ИИ, уже в принципе разработаны и осталось только соединить их некоторым правильным образом для получения нужной синергии. Многие эксперты настроены более скептически, полагая, что

необходимо принципиальное решение проблем, которые обсуждаются в этой книге.

Очевидно, разнятся и ожидаемые сроки наступления этого события, причем настолько, что одни исключают саму его возможность, а другие допускают, что оно произойдет в течение трех–четырех лет. И те и другие сходятся в том, что создание технологии «ИИ уровня человека» (HLAI — human-level AI) достаточно быстро приведет к появлению систем, существенно превосходящих человеческие возможности, то есть «ИИ сверхчеловеческого уровня» (superhuman AI).

Возникновение сверхинтеллекта выглядит закономерным в перспективе эволюции, как закономерно возникновение молекул из атомов, клеток из молекул, организмов из клеток, выделение специализированных клеток в центральную нервную систему, возникновение социальных структур, развитие речи, письменности и информационных технологий. Похоже, что оно неизбежно, если прежде наша цивилизация не погибнет.

Другое дело, что искусственный интеллект — не единственно возможный путь к сверхразуму. В научной фантастике с разной степенью детальности и правдоподобности описано множество воображаемых вариантов сверхразума, но редко ставится вопрос, какой из них возникнет раньше и продолжит развиваться дальше, все больше обгоняя возможные альтернативы. Замечательный фантаст Питер Уоттс в романах «Ложная слепота» и «Эхопраксия» описывает цивилизацию постлюдей, которые объединяются в коллективные сверхразумы или расширяют свои интеллектуальные способности с помощью нейроимплантатов, чтобы успешно конкурировать с ИИ.

Правда, оцифрованное сознание, вероятно, будет обладать катастрофически меньшей вычислительной эффективностью, чем прямая реализация ИИ, не требующая эмуляции биофизичеких процессов, от деталей которых уж точно можно абстрагироваться в ИИ, но без которых оцифрованное сознание

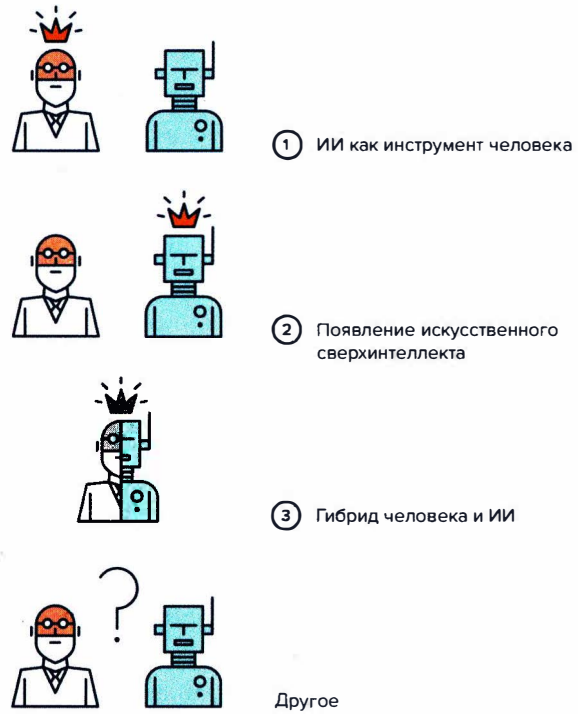


Рис. 7

Будущее человека и искусственного интеллекта: возможные сценарии

может оказаться сильно нарушенным. Так что при прочих равных ИИ будет на несколько порядков превосходить загруженные сознания по быстродействию. А еще в ИИ мы можем сделать качественные улучшения в плане интроспекции, контроля памяти и т.д., недоступные человеку и, как следствие, его оцифрованному варианту. Еще один гипотетический вариант — создание гибридных человеко-машинных систем, сочетающих преимущества и человеческого, и искусственного интеллекта.

Пока что мы можем только мечтать (или бояться) и строить умозрительные гипотезы, ведь на этом этапе развития ИИ

невозможно делать какие-то серьезные прогнозы насчет скорости появления сверхИИ и того, как он будет проявляться. И в этом тоже заключается определенный парадокс. С одной стороны, жаль было бы тратить время на легкомысленные спекуляции, пока не решено столько практических задач базового уровня. С другой стороны, вероятно, шаги в направлении общего искусственного интеллекта как-то приближают нас к искусственному сверхинтеллекту, и в какой-то момент может стать просто поздно начинать задумываться о том, как мы хотели бы с ним взаимодействовать.

Отдельные исследователи и институции занимаются этим уже давно. Так, в Институте сингулярности для искусственного интеллекта (Singularity Institute for Artificial Intelligence, ныне Machine Intelligence Research Institute), основанном известным энтузиастом проблематики дружественного ИИ Элиезером Юдковским, уже 20 лет исследуются вопросы минимизации потенциальных рисков для человечества, анализируются возможные негативные сценарии и изучаются архитектуры ИИ, которые бы способствовали более эмпатичному и дружелюбному поведению. В последние годы эта проблематика из маргинальной превращается в предмет пристального интереса и обширных исследований: по ней проводятся международные конференции, выдаются научные гранты. Тем не менее основная масса специалистов по ИИ все еще полагает этот интерес преждевременным, а содержательные выводы по проблемам безопасности AGI — невозможными без более детального понимания того, как именно этот AGI может быть устроен.

ГЛАВА 2.
КАК МЫ УЗНАЕМ, ЧТО СОЗДАЛИ AGI?

В романе Станислава Лема «Солярис» у ученых ушли годы на то, чтобы предположить, что Океан, представляющий собой студенистую субстанцию, покрывающую поверхность Соляриса, целенаправленно изменяет орбиту планеты. А догадки о его высокоразвитом разуме были сделаны исходя из сложнейших рисунков, создаваемых им на своей поверхности, еще до установления контакта с ним. Казалось бы, в случае с искусственным интеллектом, который мы создаем сами, все должно быть намного проще, но это не совсем так.

Действительно, как мы узнаем, что очередную попытку стоит признать успешной и перед нами вправду мыслящая машина, а не просто хитрый алгоритм со сложным, но не разумным по-настоящему поведением? Этот вопрос возникает в разных контекстах. К нему нередко приводит обсуждающаяся в философии проблема того, может ли в принципе машина мыслить. Звучащие порой громкие заявления отдельных особо амбициозных исследователей или корпораций о том, что настоящий ИИ наконец создан, тоже ставят вопрос о том, как эти заявления надежно проверить. Критерии и тесты интеллектуальности являются естественным продолжением попытки

строго определить понятие интеллекта. Данный вопрос позволяет четче понять, какую цель мы в конечном итоге преследуем, к чему мы стремимся. Хотя по данному вопросу не так много можно сказать без содержательного развития самой области общего искусственного интеллекта, он является методологически важным, и разные мыслители им давно задавались. И после того, как ИИ начали воспринимать как достижимую цель, а не мысленный эксперимент, вариантов ответа появилось великое множество.

Зачастую эти ответы обретают форму какой-то сверхсложной задачи, вызова, который компьютеру нужно преодолеть, чтобы тогда мы уж точно были уверены, что перед нами мыслящая машина. Полное решение подобных задач очень далеко, а частичное решение может не представлять интереса с точки зрения не только AGI, но и ИИ. Именно поэтому возникает отдельный вопрос об оценке промежуточных результатов: как мы сможем понять, что разрабатываемая нами система стала хоть немного ближе к общему искусственному интеллекту или что другая система не продвинулась на этом пути дальше? Ответ на этот вопрос важен не только для исследователей, но и для инвесторов, вкладывающих деньги в развитие ИИ. Для них это не просто умозрительный вопрос, отвечая на который можно ошибиться либо в будущем просто поменять свою точку зрения. Значимость цели построения общего ИИ не вызывает сомнения. Но как понять, насколько она далека, да и достижима ли вообще? Приближаемся ли мы к ней или топчемся на месте? Продолжать ли вкладывать деньги и в какой проект? А может, и вовсе не начинать?..

К сожалению, KPI[8] в классическом смысле слова плохо переносятся в сферу разработки общего искусственного

---

[8] Key Performance Indicators, KPI — числовые показатели деятельности подразделения (предприятия), которые помогают организации в достижении целей или обеспечении оптимальности процесса.

интеллекта. Почему — мы расскажем позже. А пока давайте пройдемся по самым популярным тестам, оценивающим уровень искусственного интеллекта.

А что мы оцениваем?

Оценивая искусственный интеллект, мы можем подходить к делу с двух сторон. Первый вариант — когда мы ориентируемся на функциональность: насколько хорошо агент решает ту или иную задачу. Это подходящий способ оценки «узкого» ИИ. Второй вариант — когда мы ориентируемся на способности. Способность означает, что агент может справляться с определенным спектром задач без предварительной подготовки к конкретному заданию (например, способность решать логические задачи или ориентироваться в пространстве). В идеале AGI должен быть именно таким. Между умением решать конкретную задачу и интеллектуальными способностями в широком смысле слова, вероятно, должен быть какой-то спектр промежуточных состояний, если одно в принципе способно перетекать в другое.

Оценивать то, как ИИ справляется с конкретной задачей, мы можем либо в режиме «белого ящика», когда мы понимаем, как устроен процесс, либо в режиме «черного ящика» — когда оцениваем только степень успешности и не видим процесс решения, хотя при желании можем проанализировать, как ошибается система, и даже поставить задачу так, чтобы система, в том числе и нейросеть, не просто выдавала, например, «истина» или «ложь» на формулировку теоремы, а генерировала текст доказательства на выходе. При тестировании методом «черного ящика» мы можем сделать следующее.

- Поставить перед ИИ какой-то набор контрольных заданий. Тут возникает проблема, связанная с тем, что этот набор обычно известен заранее, а иногда разработчики заранее знают и решения, и в этом случае мы скорее замеряем интеллект тех, кто программирует ИИ. Эту трудность можно обойти, если выдавать программе случайно выбранные задания из очень большого или бесконечного множества, но тогда возникает другая сложность — как создать большое множество реалистичных задач.
- Пустить соревноваться друг с другом, например играть в какую-нибудь игру. Сложность здесь в том, чтобы получить объективную оценку эффективности систем, ведь в разных соревнованиях будут разные оппоненты со своими результатами. Если хоть часть игроков в соревнованиях совпадает, то можно ввести систему рейтингов. Но если оппонентов как-то стандартизировать для удобства оценки, новая система может специализироваться под конкретного оппонента, вместо того чтобы развивать «широкий» интеллект.
- Поставить перед ИИ задачу выдать себя за человека (имитационная игра). С одной стороны, это выглядит как максимально амбициозная задача, с другой — здесь мы попадаем в ловушку антропоцентричности. Искусственный интеллект необязательно должен быть «человекообразным», чтобы подходить под критерии интеллекта. Кроме того, судьи-люди субъективны, и часто их убеждает хорошая внешняя имитация поведения, а не проявленный программой интеллект.

Если же мы пытаемся оценить способность, мы упираемся в то, как эту способность точно распознать и описать. Например, когда мы говорим об эмоциональном интеллекте (EIQ), мы включаем в это понятие целый ряд способностей: чтобы считаться человеком с развитым EIQ, надо уметь распознавать

свои и чужие эмоции, моделировать в уме поведение и мотивации других людей, влиять на собственное и чужое эмоциональное состояние, используя разные инструменты в зависимости от контекста, и т.д.

> **Мы интуитивно понимаем, у кого из наших знакомых высокий эмоциональный интеллект, но формализовать его описание очень сложно. Если говорить о человеческом мышлении, то разные теории предлагают разные классификации способностей.**

Но перед тем как вернуться к этому вопросу, мы немного расскажем про первый тест оценки машинного интеллекта и его подводные камни.

Тест Тьюринга и другие имитационные игры

Имитационные игры будоражат наше воображение: есть что-то кощунственное и одновременно дерзновенно прекрасное в создании по образу и подобию своему машины, которая однажды может стать неотличимой от создателя. С одной стороны, в этом есть что-то близкое к божественному творению, с другой стороны, нас преследует эффект «зловещей долины»[9]:

[9] Психологический феномен, сформулированный японским ученым Масахиро Мори в 1978 г.: отторжение вызывают роботы, которые весьма похожи на человека, но все же отличаются — когда они, на первый взгляд, воспринимаются как люди, но потом видно, что с ними что-то не так, то есть они распознаются нашим мозгом не как роботы (не-люди), а «испорченные люди». Чем больше робот похож на человека, тем симпатичнее он кажется — но лишь до определенного предела, после которого он начинает вызывать дискомфорт и страх.

в человекоподобной подделке есть что-то тревожное. Это сложное смешение эмоций вдохновляет фантастов — в первую очередь, конечно, вспоминается роман Филипа Дика «Думают ли андроиды об электроовцах».

Самый известный тест такого рода и одновременно первый тест на оценку машинного интеллекта предложил Алан Тьюринг в 1950 г. Тест воспроизводит ситуацию диалога, в ходе которого судья из плоти и крови должен определить, кем является его собеседник — человеком или машиной. Соответственно, программу, которую нельзя отличить в общении от живого человека, можно признать настоящим искусственным интеллектом. Тест Тьюринга широко известен и пользуется большой популярностью. С 1990 г. проводится даже ежегодный конкурс на премию Лебнера, в котором программы соревнуются в прохождении теста.

Изначально тест был основан на вечериночной игре, которую в наши времена признали бы весьма неполиткорректной. В ней есть три игрока: игрок А — мужчина, игрок В — женщина и игрок С — ведущий любого пола. По правилам игры, С не видит ни А, ни В и может общаться с ними только через письменные сообщения. Задавая вопросы партнерам по игре, С пытается определить их пол. Задача игрока А — так запутать С, чтобы он сделал неправильный вывод, а задача игрока В — наоборот, помочь С принять правильное решение.

Тьюринг предложил: пусть роль игрока А в одном случае играет мужчина, а в другом — компьютер. Задача обоих — успешно притвориться женщиной. Если в игре против компьютера С ошибается так же часто, как в игре против мужчины, можно говорить о том, что компьютер разумен. Позже задача С сместилась от угадывания гендера к тому, чтобы понять, кто из собеседников — человек.

Главное достоинство теста — произвольность разговорных задач. В хорошо поставленном тесте затрагиваются разные темы и моделируются разные поведенческие ситуации

(собеседник может перебивать, переспрашивать, задавать дополнительные вопросы, шутить и т.д.), и его поддержание требует как знания правил языка и общей эрудиции для релевантных реакций на реплики собеседника-человека, так и понимания определенных социальных контекстов. И у судьи есть достаточно времени, чтобы оценить собеседника. Так что машина, претендующая на победу, должна была бы обладать широкой картиной мира, использовать естественный язык, рассуждать, обучаться и понимать контекст.

> Тест Тьюринга очень антропоцентричен, то есть соответствует исключительно человеческим представлениям о разумной коммуникации.

Четких критериев оценки у него нет, и непонятно, как оценивать прогресс или сравнивать разговорный уровень двух машин. Кроме того, на практике в тестах редко учитывается возможность принять человека за машину, хотя такой вариант вполне реален.

В результате программы, претендующие на прохождение теста, не вносят заметного вклада в развитие AGI или искусственного интеллекта в целом. Анализ решений показывает, что, хотя уровень программ растет из года в год, качественно они не приближаются к поставленной цели. Оказывается, можно в ручном режиме произвести достаточно сложное дерево диалогов, основанное на закодированных шаблонах и правилах, и создать у тестировщика иллюзию взаимодействия с интеллектом человеческого уровня. К тому же есть возможности для «читерства»: программа может притворяться ребенком или, например, человеком, который нетвердо владеет рабочим языком испытания. Она может отказываться отвечать на те или иные вопросы, потому что ей этого «не хочется». В таком случае ей не нужны ни картина мира, ни рассуждения, ни даже просто понимание содержания диалога.

Ограничения теста стали особенно очевидны после того, как одну из его версий прошла в 2014 г. программа Eugene Goostman, написанная русскими и украинскими программистами. Создатели потратили десятки человеко-лет разработки на создание «дерева диалогов» и соответствующей системы исполнения, чтобы сымитировать украинского мальчика, плохо говорящего на английском языке. Залогом успеха Евгения Густмана было не красноречие, а биография: судьи готовы были легко простить программе пробелы в грамматике и недостаток знаний. Так что на соревнованиях 2014 г. каждый третий судья признал программу человеком. Но легко догадаться, что способность обучаться и адаптироваться к условиям обучения в этой программе не проверялась, не реализовывалась и не могла быть реализована.

Тут важно подчеркнуть, что это не снижает значимость теста Тьюринга для развития ИИ. Он оказал большое влияние на философию и понимание машинного интеллекта, хотя и не очень подходит для его практического измерения, да и задумывался не для этого.

Осознавая эти ограничения, исследователи предложили взамен исходного теста множество альтернативных вариантов, которые можно рассматривать как метрики достижения общего интеллекта. В частности, появился ряд испытаний, тоже опирающихся на естественный язык.

Исходную идею разговорного интеллекта продолжает Baby Turing Test, описанный в 1985 г. Этот подход предполагает обязательное обучение в процессе прохождения испытания, когда на начальном этапе мы можем верифицировать отсутствие у системы необходимых знаний, а в конце пути предъявляем ей классический тест Тьюринга в полном объеме. По мере прохождения теста у нас есть возможность проверить, насколько хорошо система следует заданной кривой обучения (learning curve).

Очевидно, что эта идея может быть адаптирована к любому набору сред и критериев тестирования в зависимости

от того, какой тип интеллекта мы хотим оценивать — способность к классификации, прогнозированию или поддержанию диалога на естественном языке.

Также существует тест Winograd Schema Challenge (WSC)[10], ориентированный одновременно и на понимание естественного языка, и на наличие здравого смысла. В оригинальной версии он состоит из 150 утверждений, за которыми следует вопрос и несколько вариантов ответов. Вот пример тестового задания.

> **Трофей не поместился бы в коричневый чемодан, потому что он был слишком большим. Что было слишком большим?**
> **Ответ 0: трофей.**
> **Ответ 1: чемодан.**

Для человека очевидно: для того чтобы трофей не влез в чемодан, он должен оказаться больше чемодана. Эту информацию нельзя получить из самого задания: она предполагает наличие у тестируемого житейского здравого смысла. При этом когда какая-нибудь «Алиса» на «Приятно было с тобой пообщаться» отвечает: «Доброе слово и боту приятно», она реагирует на ключевые слова в реплике пользователя, и это намного менее интеллектуальная операция, хоть и звучит весьма живо. Кроме того, тесты вроде WSC хороши тем, что за ответы программа получает баллы и, соответственно, тут есть численная метрика, по которой легко сравнивать разные системы и оценивать уровень прогресса[11].

Winograd Schema Challenge входит в более широкий тест GLUE (General Language Understanding Evaluation),

---

[10] http://commonsensereasoning.org/winograd.html

[11] В том случае, если тест содержит подвыборки с инкрементным (то есть, растущим на определенную постоянную или переменную величину) возрастанием сложности.

включающий и другие типы задач: например, истинность одного предложения предлагается оценить на основе информации из предыдущего предложения (и тоже с поправкой на здравый смысл). Скажем, из фразы «Президент Трамп приземлился в Ираке, начав свой семидневный визит» следует, что президент Трамп находится с визитом за границей, и если вы это поняли, то проходите тест. Какое-то время машины на нем проваливались, пока Google не представил модель глубокого обучения BERT, которой дали решать, какие части предложения являются более значимыми, научили читать текст и с начала, и с конца и подбирать недостающие слова по контексту. В итоге программа так хорошо научилась распознавать поверхностные закономерности в формулировках, что смогла решить логическую задачу на понимание аргументации по определенному тезису (например, нужно было выбрать правильный вариант аргумента для утверждения «курение вызывает рак») не хуже человека. Но значит ли это, что BERT овладела логикой? Нет. Если убрать подсказки в структуре формулировок, результаты падают почти до уровня случайного выбора.

Акцент на понимание смысла, знания и рассуждения был сделан и в конкурсе на сдачу ЕГЭ по русскому языку, проводившемся в рамках AI Journey 2019. Однако вместо развития указанных общих навыков в системах ИИ основной работой при создании моделей для таких конкурсов оказывается доводка существующих моделей под особенности задач и комбинирование этих моделей в специализированные архитектуры и пайплайны. Например, тест ЕГЭ подразумевал 27 типов заданий, и решение начиналось с применения специально обученного классификатора, который распознавал тип задания и передавал управление на модель, заточенную разработчиками конкретно под этот тип. Модели под некоторые типы заданий могли использовать какие-то общие компоненты, такие как BERT. Но никто из участников, конечно, не улучшал

этих общих компонентов. И пусть лучшие решения, участвовавшие в соревновании, прошли тест на четверку, но если бы эти системы тестировались на таких типах заданий, которых не было на этапе разработки и обучения, вряд ли хоть одна из них получила бы положительную оценку.

> **На текущем этапе развития технологий достаточно сложно создать такой языковой тест, чтобы выигрыш от внесения в решения общих элементов «понимания» и «мышления» превосходил выигрыш от более простых методов, но при этом тест не был бы катастрофически сложным.**

Качественный учет и накопление поверхностных корреляций в языке, дополнительная ручная разметка данных, изобретение эвристик под конкретные тесты — все это пока что дает более легко достижимые улучшения в результатах бенчмарков. Чтобы устранить этот недостаток, исследователи ищут другие пути, предлагая, например, тесты[12], в которых система не может непосредственно извлечь ответ на вопрос из существующих текстов, а должна сперва выполнить некоторые манипуляции со словами и символами.

Другой вариант имитационной игры — когда программа притворяется человеком в многопользовательских видеоиграх. Пример такого состязания — BotPrize. Несколько лет тест проходил в форме игры Unreal Tournament 2004, персонажами которой управляли реальные люди или компьютерные программы. Живые игроки одновременно выступали в роли судей, что несколько затрудняло задачу: не очень просто сосредоточиться на оценке соперников, когда ты сам рискуешь быть убитым в игре. Поэтому со временем в состязание стали добавлять оценку от внешних наблюдателей.

---

[12] https://medium.com/@runarphius/a-test-for-true-natural-language-understanding-e9879241c07d

Подобные тесты могут быть полезны, чтобы стимулировать те направления исследований, которые будут способствовать решению задач AGI. Но они все-таки проверяют достаточно ограниченный диапазон навыков. Поэтому многие тесты на достижение уровня AGI выходят за рамки таких заданий.

Тесты на оценку способностей

Если мы хотим оценить способности машины, кажется логичным использовать подходы, которые мы применяем для оценки интеллекта живых существ. Ряд исследователей предлагают оценивать прогресс по достижению уровня интеллекта различных животных. Ведь с точки зрения решения определенного набора задач в непредсказуемой среде животные вполне себе эффективны. Этой идеей, например, вдохновлен тест Animal-AI Olympics[13]. Вот пример классического лабораторного эксперимента, который воспроизводится в «цифре»[14].

Перед животным ставятся две перевернутые миски — под одну из них кладется еда, которую животному надо достать. Сначала еда каждый раз кладется под одну и ту же миску, пусть она будет миской 1. Это этап обучения. Затем человек кладет еду под миску 1, вынимает и очень заметно перекладывает под миску 2. Животные с более развитым интеллектом, например шимпанзе, пойдут сразу к миске 2, но многие по-прежнему перевернут миску 1, потому что выучили задачу через запоминание.

Однако пока не очень понятно, насколько эффективно идти к AGI путем воспроизведения возможностей животных. Таким

---

[13] animalaiolympics.com

[14] https://www.youtube.com/watch?v=ok9opyg0Ofg

образом сложно выработать метрики, которые позволяли бы объективно сравнивать системы proto-AGI — по крайней мере, построенные в рамках разных направлений. Но таким образом можно было бы оценить текущий прогресс в рамках одного направления или даже конкретного подхода. А еще через аналогии с животными можно создавать тесты, стимулирующие брать новые рубежи в области AGI, даже если в качестве метрик прогресса они будут не очень эффективны. В этой связи делаются попытки тестировать компьютер на тестах, разработанных под человека.

В 2017 г. исследователи из Пекинского университета Цзяотун создали метод оценки IQ современных систем ИИ[15]. Самый высокий результат показал поисковик Google, чей IQ оказался немного ниже уровня шестилетнего ребенка (при том что по состоянию на 2014 г. его IQ был бы в два раза ниже), а вот Siri оказалась седьмой. Удобство этого метода в том, что он быстрый, формализованный и его выполнение можно автоматизировать. Но одновременно с этим он мало приспособлен для оценки умственных способностей машин в других аспектах. Тест заточен на оценку человеческого интеллекта, но даже в этой области ведутся споры насчет того, что именно и насколько качественно он измеряет. Люди с более нестандартным мышлением могут отвечать на вопросы «неправильно» с точки зрения теста, подразумевающего стереотипные ответы. Кроме того, не очень понятно, как сводить набор независимых и разных типов мыслительных способностей (творческий интеллект, эмоциональный интеллект, языковой интеллект, абстрактно-логический интеллект и т.д.) в единую концепцию интеллекта[16]. К тому же высокие баллы по IQ далеко не всегда обеспечивают успех в решении задач как в жизни, так и в профессиональной деятельности.

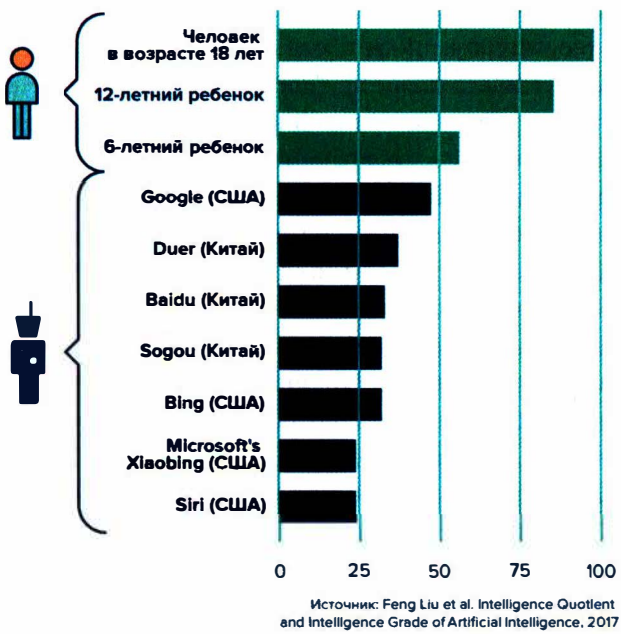
---

15 https://arxiv.org/abs/1709.1024

16 https://www.sciencedaily.com/releases/2012/12/121219133334.htm

Поэтому некоторые специалисты предлагают в качестве критерия для оценивания AGI использовать задачи, максимально приближенные к реальному миру.

Например, программист и создатель компьютера Apple I Стив Возняк предложил использовать для этой цели «кофейный тест»: создать машину (робота), которая может войти в любой среднестатистический американский дом, найти в нем кофемашину, кофе и кружку, набрать воды и сделать кофе. В чем-то сходный тест IKEA предполагает сборку чего-либо, например одноименной мебели или конструктора LEGO, по инструкции. Правда, сложность тут в основном



Источник: Feng Liu et al. Intelligence Quotient and Intelligence Grade of Artificial Intelligence, 2017

Рис. 8

Сравнение человеческого и искусственных интеллектов по абсолютным значениям IQ

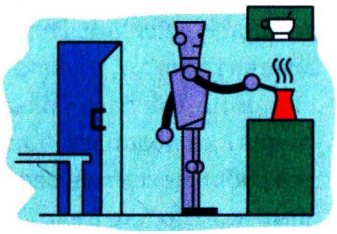
заключается в тонкостях мелкой моторики и обращения с инструментами, а вот для интеллектуального аспекта задачи может подойти и очень специализированное решение. Это будет очень частная модель для управления роботом, обученная на множестве примеров инструкций, и ни для чего больше она (и тем более результат ее обучения) не подойдет. Такие тесты могут способствовать развитию робототехники в целом, но до проблем AGI при работе над ними дело просто не дойдет.

Даже если та или иная задача выглядит так, будто для ее решения нужны развитые и разнообразные когнитивные навыки, она совсем необязательно будет хорошим материалом для практических бенчмарков, по крайней мере таких, которые измеряют прогресс достижения AGI. Поэтому место узких задач все чаще занимают испытания, предполагающие реализацию интеллектуальным агентом некой сложной системной деятельности, не сводящейся к прохождению тестовых заданий.

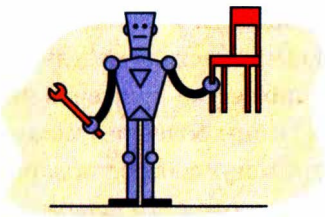
Приведем несколько примеров в порядке возрастания сложности.

- **«Банковский тест»: ИИ должен взять кредит в банке на выбранную им цель, заработать денег легальным способом и вернуть его с процентами.**
- **«Тест студента» Бена Герцеля: искусственный интеллект поступает в университет, посещает те же занятия, что и обычные студенты, выполняет те же задания и получает соответствующую степень.**
- **«Тест на профпригодность» Нилса Нильссона: машина выполняет экономически важную работу на том же уровне, что и профессионалы в данной области (фактически являясь полноценным Narrow AGI).**
- **Artificial Scientist Test: компьютер совершает оригинальное научное открытие.**
- **Nobel Prize Test: машина получает Нобелевскую премию.**

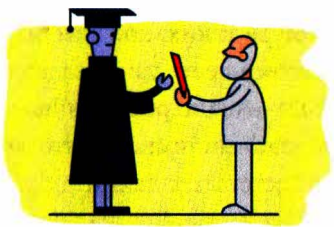
О создании AGI можно будет говорить, когда ИИ сможет...



...войти в незнакомый дом, найти кружку и сварить кофе*

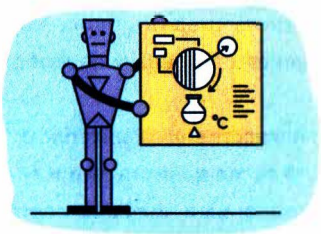


...собрать предмет мебели из IKEA*

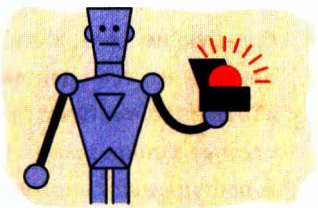


...окончить университет, посещая те же занятия и выполняя те же задания, что и люди

...выполнять экономически важную работу на уровне человека



...сделать научное открытие



...получить Нобелевскую премию

* Для выполнения этих тестов ИИ понадобится подвижное физическое тело.

Рис. 9

Это общий искусственный интеллект или еще нет?

Еще одним критерием достижения AGI может быть решение ИИ-полных задач[17]. К ним можно отнести понимание естественного языка, компьютерное зрение, рецензирование научных статей.

Вообще, любая комплексная задача, включающая в себя возникновение непредвиденных обстоятельств (например, очень популярна сейчас задача беспилотного управления автомобилем), требует интеллекта уровня как минимум Narrow AGI. К сожалению, понятие ИИ-полноты слишком расплывчато, чтобы служить четким критерием. Например, мы легко можем согласиться, что «настоящее понимание текстов» невозможно без полноценного AGI, но это убеждение ничего не говорит нам о том, как проверить, что понимание — настоящее.

Даже в случае с беспилотным автомобилем может оказаться, что можно обойтись без Narrow AGI. Хотя нам и кажется, что такой автомобиль должен правильно реагировать на непредвиденные ситуации, но в норме этих ситуаций сравнительно мало. Система, управляющая автомобилем, может непредсказуемо вести себя в необычных ситуациях, но при этом демонстрировать средний уровень аварийности меньше, чем у человека. То, что при этом для каждой конкретной страны или даже города эту систему придется доучивать и дорабатывать, — это уже другой вопрос. Это очень трудозатратно, но реализуемо. И то, что система не обучена правильно реагировать на посадку самолета на дорогу и в такой ситуации поведет себя непредсказуемым образом, может служить предметом критики таких узких систем, но часто ли самолеты садятся на дорогу или происходит что-то необычное?

---

[17] ИИ-полная задача (AI-complete) — термин, предложенный математиком Фанья Монтальво. Это задача, которая не заключается непосредственно в создании ИИ, но решение которой без полного интеллекта, близкого по уровню к человеческому, представляется невозможным.

Люди совершают много ошибок и в нормальных ситуациях. Так что превзойти уровень человека даже для такой задачи, как управление автомобилем, может быть, можно и узкими методами, особенно если самих людей-водителей на дорогах не останется. Конечно, если бы технологии AGI появились, они бы позволили решить эту проблему лучше и с меньшими затратами, но в контексте тестов и метрик получается, что для оценки развития и достижения AGI как бенчмарк необязательно подходят даже беспилотные автомобили и требуется что-то еще более впечатляющее.

Таким образом, тесты, которые убедили бы нас, что AGI действительно создан, предполагают выполнение одного из двух условий. Машина должна заменить человека в некоторой интегральной профессиональной деятельности или добиться впечатляющих по человеческим меркам свершений. Соответствие данным критериям будет означать, что, когда AGI будет создан, мы это точно заметим, но при этом не сможем оценивать промежуточные стадии. Наконец, все приведенные примеры показывают, что под «достижением AGI» подразумевается обычно AGI уровня человека.

Коэффициент универсального интеллекта

В результате мы оказываемся в некотором тупике: практические бенчмарки могут оценивать только узкоспециальные навыки, а тесты на достижение полного AGI не позволяют оценить частичный прогресс.

В этой ситуации было бы здорово иметь некую общую метрику интеллекта для всех разумных агентов. Опираясь на нее, можно было бы создавать способы оценки промежуточных результатов. Если попытаться определение интеллекта как

способности агента успешно действовать в широком диапазоне сред представить в математически строгом виде, возникает такая метрика, как коэффициент универсального интеллекта (Universal Intelligence Quotient, или UIQ[18]).

Чтобы ее построить, нужно сначала задаться вопросом: как можно формально определить наиболее широкий класс сред? Мы бы могли сказать: все существующие среды. Но это не будет формальным определением. Фактически нам нужен наиболее широкий класс не самих сред, а их моделей, математических представлений. И такой класс моделей — это все возможные алгоритмы.

Рассматривая различные тесты и бенчмарки, мы видели, что основная проблема их использования для оценки AGI заключалась в том, что решения к ним можно и нужно было подгонять. Это очевидно, ведь Deep Blue не сочиняет истории, AlphaGo не распознает картинки, а BERT не играет в шахматы. Что уж говорить про решения для тестов, основанных на каких-то фиксированных реальных данных или же синтезированных задачах, обладающих ограниченной структурой? В итоге поиск закономерностей в данных или учет этой структуры был заслугой больше людей-разработчиков, чем разработанного ими решения.

Но что интересно: синтезированные задачи или виртуальные среды порождаются той или иной программой. Да и тест, хоть и основанный на реальных данных, но проверяемый на компьютере, тоже является компьютерной программой. Чуть сложнее обстоит дело с тестами, проводимыми непосредственно в реальном мире, но пока что все наши модели реальности тоже являются алгоритмическими. Даже если у нас сейчас нет достаточно точной симуляции робота, входящего в произвольный дом, чтобы приготовить кофе, можно

---

[18] Shane Legg and Marcus Hutter. Universal intelligence: A definition of machine intelligence. Minds and machines, 17(4): 391–444, 2007.

предположить, что в принципе может существовать такая симуляционная программа, которую робот не смог бы отличить от реальности. Таким образом, пространство всех возможных компьютерных программ (или реализуемых ими алгоритмов) столь богато, что в нем можно найти любой бенчмарк, тест, задачу, среду. Если мы будем выбирать произвольную задачу из этого пространства, то заслугу ее решения уж точно можно будет приписать самому компьютеру.

Но чтобы построить количественную метрику уровня интеллекта, нужно определиться, как именно оценивать успешность действия интеллектуального агента и как объединять оценки, полученные в разных средах. В UIQ рассматриваются среды, задаваемые программами, которые возвращают агенту значение вознаграждения — как в стандартной постановке обучения с подкреплением. Успешностью действия агента в среде является полученное им суммарное вознаграждение (точнее, его математическое ожидание, если агент или среда допускают случайное поведение).

UIQ, будучи теоретическим критерием, рассматривает усреднение суммарных вознаграждений по всем возможным средам. Но среды берутся с разными весами, соответствующими вероятности встретить ту или иную среду. А именно, вероятность среды экспоненциально падает с длиной ее программы. В последующих главах мы подробнее обсудим, почему такие веса являются осмысленными, а пока лишь отметим, что число программ растет экспоненциально с их длиной, так что вероятность случайно выбрать конкретную программу экспоненциально падает (так, вероятность выбросить последовательность орел — решка при двух подбрасываниях монеты равна 1/4, тогда как вероятность выбросить, скажем, орел — решка — решка — решка при четырех подбрасываниях составит 1/16; кроме того, в длинной последовательности подбрасываний орел — решка будет встречаться в четыре раза чаще, чем орел — решка — решка — решка).

Интуитивно может показаться, что «победа» в более сложной среде должна значить намного больше. Но на самом деле нет. Если бы нам нужно было выбрать одну задачу для тестирования интеллекта, мы бы подумали о сложной задаче, надеясь, что если уж агент способен решать сложную задачу, то с простой-то он справится. Но когда мы производим усреднение по всем средам, то мы можем не делать такого допущения, и простые среды оказываются ничем не хуже, чем сложные. Скорее, наоборот. Если агент хорошо играет в шахматы, но не способен решать шахматные задачи, то чем он лучше агента, решающего шахматные задачи, но плохо играющего в шахматы?

> **Итак, мы можем определить коэффициент универсального интеллекта как значение получаемых агентом суммарных наград во всех вычислимых средах, взятых с весами, убывающими с длиной их программ.**

Правда, тут же мы сталкиваемся с набором серьезных ограничений в практическом применении: невозможность определить, зациклится ли программа той или иной среды; усреднение по бесконечному количеству сред; усреднение по всем взаимодействиям агента с миром.

Но в то же время эта теоретическая мера интеллекта обладает целым рядом положительных свойств. Например, оптимизация этой меры эквивалентна построению универсального интеллектуального агента AIXI, о котором мы будет говорить в следующих главах. В отличие от теста Тьюринга, в котором агент прошел или не прошел испытание, UIQ является непрерывной мерой успеха, способной оценивать прогресс в системах AGI. Кроме того, эта мера не является антропоцентрической и не отдает априорного предпочтения тем или иным предметным областям или классам задач. К тому же UIQ дает объяснение нашему интуитивному пониманию уровня общего интеллекта разных систем. Например, отчасти обосновывает

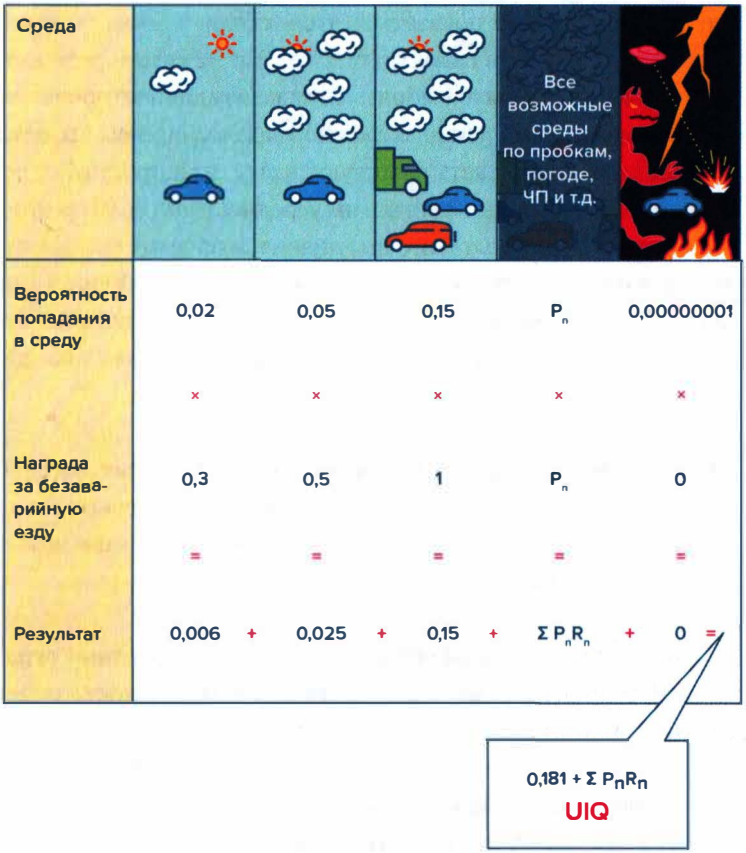


Рис. 10

Как считается коэффициент универсального интеллекта (UIQ)

использование теста IQ для оценки уровня человеческого интеллекта — там также делают акцент на разнообразных задачах низкой сложности, — но при этом показывает и его предвзятость: задачи в IQ явно подобраны под человека, причем современного. В то же время UIQ показывает, что никаким фиксированным набором специализированных интеллектов (геометрического, языкового и т.д.) пространство задач для общего интеллекта не покрыть.

Для практического удобства появляются разные упрощения UIQ, например AIQ (Algorithmic Intelligence Quotient — коэффициент алгоритмического интеллекта[19]), а также системы тестов[20], учитывающие вид среды (активная, пассивная), наличие времени, определение интеллекта и видов универсальных агентов. В них устанавливаются ограничения по времени и вычислимости, ограничения на число взаимодействий и шкалы вознаграждений и штрафов в диапазоне (–1,1). Сложность сред возрастает по мере прохождения, и они сбалансированы так, чтобы случайно действующий агент набирал суммарное вознаграждение, равное нулю.

Существуют и более ограниченные тесты, отталкивающиеся от той же идеи алгоритмически произвольных закономерностей. Например, тест Abstractive Reasoning Corpus[21] представляет собой набор визуальных заданий, выполняемых на квадратной сетке, клетки которой могут иметь один из десяти цветов. Каждое задание представляет собой набор из двух-трех пар сеток, где первая сетка представляет собой «вход», вторая — «выход» или «результат». Трансформация из «входа» в «результат» осуществляется с помощью некой высокоуровневой операции (заранее неизвестного алгоритма), разной для каждого задания. Задача тестируемой системы — сгенерировать абсолютно правильную «результирующую» сетку для каждой новой входящей, увидев несколько примеров таких трансформаций. В самих заданиях заложена некая инкрементность[22] сложности, но выполнение даже самых легких

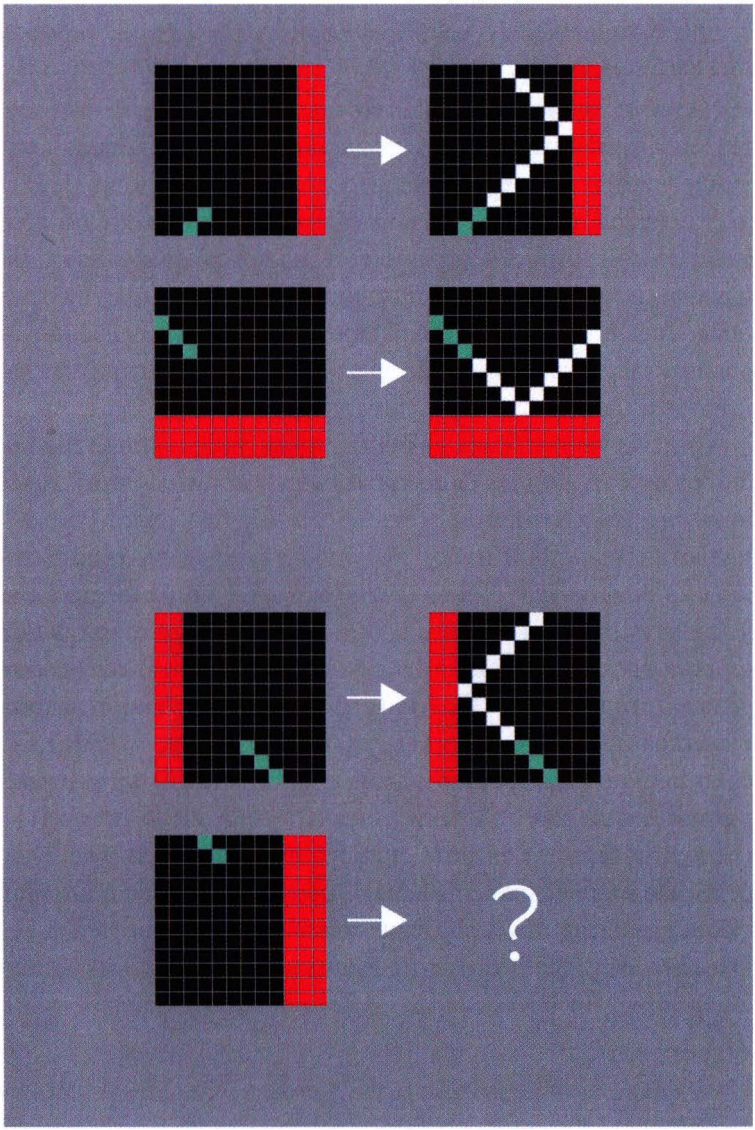
---

19 An Approximation of the Universal Intelligence Measure — Shane Legg, Joel Veness.

20 Measuring universal intelligence: Towards an anytime intelligence test — José Hernández-Orallo, David L. Dowe — Artificial Intelligence 174 (2010) 1508–1539.

21 https://arxiv.org/abs/1911.01547

22 Приращение на определенную величину.



Источник: François Chollet. On the Measure of Intelligence

Рис. 11

У искусственного интеллекта подобные задания вызвали трудности

заданий на текущем уровне технологий не представляется возможным.

Другим примером, по духу сходным с UIQ, может служить General AI Challenge[23], в котором агент взаимодействует со средой, получает на вход последовательность символов и, совершая действия, посылает символы в ответ. При этом участникам совершенно неизвестен класс среды и какие-либо принципы ее функционирования, так что среда может восприниматься как произвольный (но, вероятно, несложный) алгоритм. Тест General AI был разбит на задачи возрастающей сложности и стимулировал инкрементное развитие навыков у агентов, так как решение последующих задач как-то должно было опираться на решение предыдущих. Победителя в нем, впрочем, выявить не удалось.

Отбор и сочетание сред представляет собой отдельную большую проблему. Тесты в действительно неизвестных средах оказываются слишком сложными даже тогда, когда описываются весьма простыми алгоритмами. Решения, созданные для них, даже если имеют исследовательскую значимость, обычно весьма далеки от задач реального мира, которые хоть и произвольны, но зачастую имеют свою специфику, не учтенную тестами.

В этой связи критерий UIQ может пониматься сугубо эмпирически: вместо того чтобы отбирать среды из априорного распределения, мы их «семплируем» из реальности, тестируя одну и ту же систему или агента на разных задачах реального мира. Но тогда выборка сред или задач неизбежно оказывается смещенной[24], субъективной и неполной. Если же попытаться фиксировать выборку, мы стимулируем

---

[23] https://www.general-ai-challenge.org/

[24] Смещенная выборка — это выборка, в которой удельный вес отдельной группы участников смещен относительно генеральной совокупности. Допустим, доля семей с тремя детьми в стране — 30%, а в составленной выборке — 60%.

разработку специализированных решений, характерных для всех предметно-ограниченных тестов.

> **Подводя итог, приходится признать, что хорошего практического решения проблемы метрики развития AGI в настоящее время не существует.**

Некоторые специалисты полагают, что измерение частичного прогресса в AGI и в целом крайне проблематично, потому что это эмерджентная система[25], которая не будет проявлять свойств AGI до тех пор, пока она не будет построена целиком (подобно тому, как, воссоздавая человеческий мозг по кускам, мы не можем оценить результат до самого последнего шага). Тесты на то, что мы достигли AGI, не особо основываются на наших представлениях о том, как AGI должен быть устроен, и не позволяют измерить частичный прогресс. А тесты, которые пытаются сравнивать proto-AGI системы, оказываются либо слишком сложными для существующих систем, либо слишком простыми, так что их можно решать и узкими методами; и они всегда отдают предпочтение тому или иному подходу.

Разнообразные бенчмарки и тесты нужны скорее как мотивация, но не как нечто, что требуется проходить любой ценой. Например, шахматы были вызовом для ИИ многие годы, но Deep Blue, преодолевший этот вызов, не представляет особого интереса в области искусственного интеллекта, не говоря об AGI. Достигнув своей цели, он становится бесполезен, ведь больше он ничего не умеет. Гораздо больше пользы было от шахмат как от мотивационной задачи, чем от победы над человеком во что бы то ни стало.

---

[25] Эмерджентность — появление у системы свойств, не присущих ее элементам в отдельности; несводимость свойств системы к сумме свойств ее компонентов.

Так что исследователям пока приходится самим качественно оценивать прогресс в AGI на основе текущего понимания интеллекта как комплексной проблемы и постепенно углублять это понимание, а не слепо полагаться на его оценку по каким-либо стандартизованным метрикам. Ведь, например, мы не смогли бы попасть в космос, строя все более высокие здания или более высоко летающие самолеты, — здесь потребовалось качественно иное решение. Да и прогресс в науке в целом плохо измеряется формальными индикаторами и показателями.

Глава 3.
ОСНОВНЫЕ НАПРАВЛЕНИЯ В AGI

Две отправные точки — один пункт назначения

Как бы вы приступили к разработке общего ИИ? Это непростой вопрос, связанный со множеством неопределенностей. Здесь надо за что-то зацепиться, с чего-то начать думать о самой проблеме. Две основные линии рассуждений, которые будто бы друг другу противоречат, в качестве отправных точек выбирают либо естественный интеллект, либо «техническое задание», определяющее назначение общего ИИ. Но в обоих случаях «программа максимум» одинакова — очертить все пространство возможных интеллектов. В истории науки нередко частные задачи оказывались нерешенными до тех пор, пока не удавалось найти решение более общей задачи, из которой уже выводились решения ее частных случаев. И без создания общей теории общего интеллекта, что можно считать конечной целью области AGI, может оказаться невозможным как глубокое понимание естественного интеллекта, так и создание конкретного экземпляра общего ИИ.

И все же, хотя в чистом виде эти две отправные точки не встречаются в AGI в чистом виде, их исходный фокус слишком различается, и выросшие из них обобщение неизбежно несут на себе их заметные следы, давая начало альтернативным подходам. При этом из «человеческой метафоры» в AGI обособляются два направления — нейросетевые модели и когнитивные архитектуры — в зависимости от того, черпается ли больше вдохновения в нейрофизиологическом или когнитивном устройстве естественного интеллекта. Также и с «компьютерной метафорой» интеллекта связано несколько подходов к общему ИИ. В первую очередь, стоит назвать универсальный алгоритмический интеллект, который представляет собой подход «сверху-вниз», начинающий с попытки формализации определения интеллекта в области общего ИИ и построения на этой основе модели с доказуемыми свойствами оптимальности. Вероятностный же подход отталкивается от обобщения практики повсеместного использования вероятностных моделей при решении задач, требующих различных интеллектуальных навыков — от обучения до рассуждений. Также можно выделить подход на основе компьютерных наук, который отталкивается от общей практики использования компьютеров при решении задач, которое выходит далеко за рамки обычного ИИ, но автоматизация которого должно общим ИИ обеспечиваться. Далее мы рассмотрим основные подходы к AGI.

Глубокое обучение

Глубокое обучение: третья волна ИИ

В 2006–2007 гг. в машинном обучении произошла революция, которая положила начало новой, третьей волне хайпа вокруг

искусственного интеллекта. Любопытно, что искусственные нейронные сети имели непосредственное отношение ко всем трем волнам: в 1950–60-е гг. важным двигателем интереса к этой теме стали успехи первых перцептронов[26] Фрэнка Розенблатта, а в 1980-е завышенные ожидания от искусственного интеллекта во многом порождались популяризацией обучения многослойных сетей при помощи алгоритма обратного распространения ошибки (backpropagation[27]). К сожалению, на тот момент у исследователей не было ни достаточных вычислительных мощностей, ни достаточного размера наборов данных, чтобы оправдать эти ожидания.

Однако в середине 2000-х глубокие нейронные сети действительно заработали. Результаты, полученные, в частности, в группах Джеффри Хинтона[28], Йошуа Бенджио и Яна Лекуна, привели к тому, что искусственные нейронные сети приобрели статус подхода, переворачивающего одну область применения машинного обучения за другой. В этой главе мы сначала поговорим о том, чего достигли нейронные сети к настоящему моменту, акцентируя внимание на вопросе, в какой мере эти достижения имеют отношение к AGI, а затем обсудим, насколько сейчас возможно говорить о потенциальном построении AGI с помощью этого подхода.

---

[26] Перцептрон — предложенная Фрэнком Розенблаттом в 1957 г. математическая модель восприятия информации мозгом, которая стала одной из первых моделей нейросетей.

[27] Обратное распространение ошибки — это способ обучения нейронной сети. Цели обратного распространения просты: отрегулировать каждый вес пропорционально тому, какой вклад он вносит в общую ошибку. Если мы будем итеративно уменьшать ошибку каждого веса, то в конце концов получим ряд весов, которые дают хорошие прогнозы.

[28] Эти трое ученых в 2018 г. стали лауреатами премии Тьюринга (которую часто называют Нобелевской премией по компьютерным технологиям) за разработку алгоритмов глубокого обучения.

Компьютерное зрение

Компьютерное зрение — одна из старейших и главных областей искусственного интеллекта. В начале долгого пути ИИ ведущие исследователи считали, что это несложная задача, достойная учебного проекта на лето. Однако за последующие полвека стало ясно, что создание полноценной системы компьютерного зрения — это куда более серьезный проект, и во многом соответствующие задачи до сих пор остаются не до конца решенными.

Революция глубокого обучения в компьютерном зрении началась в 2010–2011 гг., с началом воплощения идеи использования нейронных сетей на графических картах (GPU). Это позволило обучать гораздо более глубокие сети с большим числом нейронов. Первой задачей, покорившейся глубоким сетям, стала классификация изображений, а первыми выдающимися образцами — работа группы Юргена Шмидхубера (нейронные сети на основе технологии долгой краткосрочной памяти — в настоящее время эта программа работает на миллиардах смартфонов, решая задачи, связанные с распознаванием речевых сигналов) и сеть AlexNet, которая в 2012 г. с большим отрывом победила на соревновании по классификации изображений на основе набора данных ImageNet.

С тех пор глубокие нейронные сети стали широко применяться при решении практически всех задач компьютерного зрения, таких как классификация изображений, распознавание объектов, сегментация[29], определение позы/положения,

---

[29] Сегментация изображения — это разбивка изображения на множество покрывающих его областей. Сегментация применяется во многих областях, например в производстве для индикации дефектов при сборке деталей, в медицине для первичной обработки снимков, а также для составления карт местности по снимкам со спутников.

оценка глубины[30] и т.д. Основным инструментом компьютерного зрения стали сверточные сети[31], а главными вехами можно назвать следующие модели и работы:

- AlexNet — одна из первых успешных глубоких сверточных сетей (более 20 слоев); кроме того, AlexNet работала сразу на нескольких GPU, и такая параллелизация впоследствии также стала общим местом;
- появившиеся в 2013–2015 гг. новые модели (VGG, Inception, ResNet), которые дополняли базовую идею глубоких сверточных сетей новыми архитектурными особенностями;
- параллельно возникавшие сверточные архитектуры для решения других базовых задач компьютерного зрения. Например, в задаче обнаружения объектов выделились два основных направления. Сети, работающие «в два прохода» (two-stage object detection)[32], имеют разные структуры для локализации объектов на картинке и их распознавания (вначале они просто выделяют объекты, а потом уже их классифицируют). А сети, работающие «в один проход» (one-stage object detection), делают это одновременно; к последним относятся, например, се-

---

30 Карта глубины (depth map) — это изображение, на котором в каждом пикселе хранится расстояние от камеры до соответствующей точки физической сцены. Оценка глубины нужна для того, чтобы компьютер мог соотнести двухмерное изображение с трехмерным первоисточником.

31 Один слой сверточной сети выглядит как обычная, полносвязная сеть, но применяемая в локальной окрестности каждого элемента сигнала, например точки изображения. В результате каждая точка описывается набором локальных признаков. Следующий слой совершает такую же обработку, но уже применительно к признакам и обычно после уменьшения, формируя более абстрактные признаки, которые являются не заранее заданными, а выученными сетью на имеющихся данных.

32 Главный пример таких сетей — семейство R-CNN, в том числе Faster R-CNN (Ren et al., 2015), Mask R-CNN (He et al., 2018) и CBNet (Liu et al., 2019).

мейства сетей YOLO (расшифровывается как You Only Look Once) и SSD.

В течение всего развития глубокого обучения для компьютерного зрения нейронным сетям удается справляться с некоторыми базовыми задачами в этой области лучше людей.

Вот несколько примеров:

- глубокая сеть Дэна Киресана, исследователя из Института по изучению искусственного интеллекта Далле Молле, добилась отличных результатов в задаче распознавания дорожных знаков[33]: средняя ошибка человека на этом наборе данных составила 1,16%, а сеть достигла показателя всего в 0,56%;
- человеческий уровень ошибок на тестовом наборе ImageNet с 1000 классов[34] составляет минимум около 5%, а текущий рекордный результат глубоких нейронных сетей уже упал ниже 2%;
- нейронные сети превосходят людей и в решении более специализированных задач — скажем, в области распознавания медицинских изображений. Например, есть модель, которая позволяет распознавать клетки на изображениях флуоресцентной микроскопии, оценивать их размер и статус (жива или мертва клетка), добиваясь лучших результатов, чем профессиональные биологи (точность классификации живых/мертвых клеток у нейросети достигает 98%, а у специалистов — около 80%);

---

[33] IJCNN Traffic Sign Recognition Competition.

[34] Классы объектов, к которым нужно отнести картинку. Например, если у вас в данных только кошки и собаки, то это два класса. Если еще попугайчики, то три.

- сеть PlaNet, разработанная в Google, умеет угадывать по изображению место, где была сделана фотография, и справляется с этим лучше людей еще с 2016 г.

Но можно ли сказать, что задача компьютерного зрения в настоящее время успешно решена? Можно ли считать компьютерное зрение компонентом, уже готовым для применения или интеграции в системы AGI?

К сожалению, нет. Проблема с компьютерным зрением проявляется, например, при попытках решить задачу разработки автономных автомобилей. Этой задаче посвящено множество работ, и ее решение, несомненно, имело бы огромный

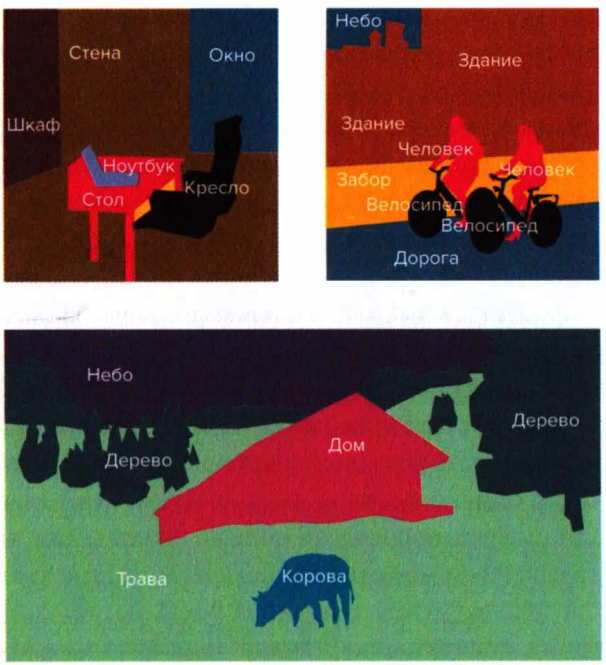


Рис. 12

Семантическая сегментация

экономический эффект и принесло бы большой доход разработчикам. Но, несмотря на вполне успешное преодоление отдельных этапов — например, решение задачи семантической сегментации[35], — автономные автомобили пятого уровня[36] пока не появились, и есть сомнения, что появятся в ближайшее время.

В чем же дело? Первая проблема состоит в том, что модели, распознающие двухмерные изображения, «не имеют понятия» о том, что эти изображения на самом деле представляют собой проекции трехмерного мира. И как сообщить модели компьютерного зрения о том, что на самом деле мир трехмерен, пока неясно. Конечно, исследователи разрабатывают и системы, работающие с трехмерными данными, но здесь возникает другая проблема — недостаток данных. Собирать наборы трехмерных сцен гораздо сложнее и дороже, и на данный момент датасетов уровня ImageNet для них не существует. Эту проблему пытаются решить при помощи синтетических данных (то есть искусственно сгенерированных для разработки и тестирования моделей), но потом возникают трудности с переносом моделей из синтетических данных в реальные.

Вторая — возможно, даже более принципиальная — проблема заключается в самой постановке задачи. Задачи компьютерного зрения, на которых удается достичь человеческого уровня и превзойти его, формулируются так: надо построить модель, которая может обобщить на тестовую выборку данные, имеющиеся на обучающей выборке. Но данные в тестовой выборке обычно имеют ту же природу, что и в обучающей, и, по сути, полученной модели не надо иметь дело с новыми контекстами. И эти модели, даже если их можно использовать

---

[35] Присвоение каждому пикселю метки, относящей его к определенному классу: машина, человек, дорожный знак и т.д.

[36] Шесть уровней автономного вождения — от 0 (ручное управление) до 5 (полностью автономный автомобиль) — были выделены в стандарте SAE (Society of Automotive Engineers).

для обучения по малому числу примеров (few-shot learning или one-shot learning)[37], обычно не могут разумно обобщать свои результаты на новые классы объектов.

> Если модель для распознавания и сегментации пешеходов в автономном автомобиле никогда не видела в обучающей выборке человека на гироскутере Segway, она не сможет создать для него новый класс и понять, что люди на гироскутерах двигаются гораздо быстрее обычных пешеходов,

А ведь это принципиально важно для системы навигации. Не исключено, что такая постановка задачи представляет собой совершенно другой уровень обобщения, который существующим искусственным нейронным сетям пока недоступен.

Обработка естественных языков: когда же модели пройдут тест Тьюринга?

Обработка естественных языков (natural language processing, NLP) — одна из главных и старейших областей применения и развития искусственного интеллекта. В частности, знаменитый тест Тьюринга, конечно, может обращаться не только к задачам обработки языков, но естественный язык играет там центральную роль; задачи Дартмутского семинара, положившего начало систематическому изучению ИИ, не включают компьютерное зрение, но многие из них посвящены естественным языкам, а первым громким провалом искусственного интеллекта, во многом вызвавшим первую «зиму» ИИ (об этом мы говорили в первой главе), стала попытка создать систему машинного перевода.

---

[37] Объясним суть наглядно: скажем, европеец решил изучить иероглифы. У него есть по одному примеру нескольких иероглифов, и он пытается понять, какие особенности начертания важны, а какие — декорация (например, завиток на конце черточки). Определенные выводы, хоть и неточные, он может сделать даже из небольшого числа примеров.

Как нейронные сети обрабатывают естественный язык? Они работают с числами, а не со словами. Чтобы перевести слова в числовую форму, используется метод one-hot encoding. Он действует так: представьте, что в нашем словаре всего 10 слов. Номер каждого слова будет состоять из 10 цифр, и все они будут нулями — кроме одной единицы. Единица будет стоять на том месте, которое слово занимает в словаре по алфавиту. Первое слово — 1000000000, второе слово — 0100000000, и т.д. Но когда мы имеем дело с объемными словарями, с таким кодом неудобно работать: он очень большой и не отображает смысловой близости слов. И тогда на помощь приходит метод распределенных представлений.

Представим себе квадратную таблицу, где каждая строчка — это какое-то слово из словаря, созданного на основе большого массива текста (например, «Войны и мира»). Слова следуют в алфавитном порядке. В столбце все то же самое, только по вертикали сверху вниз. В ячейке на пересечении столбца и строки пишется количество раз, которое слово из строки встретилось в тексте рядом со словом из столбца (рис. 13).

Эти ячейки и становятся координатами вектора. Допустим, слово «Наташа» встретилось рядом со словом «Андрей» 100 раз, со словом «пение» — 50 раз, а со словом «дуб» — ни разу. Если записать первую строку таблицы как (1; 50; 100), получится трехмерный вектор, кодирующий смысл слова «Наташа». У таких векторов есть специальный математический показатель схожести под названием «косинусная близость». Если семантические векторы двух слов «косинусно близки» по отношению друг к другу, эти векторы принадлежат словам, близким по смыслу.

Использовать метод распределенных представлений обычно гораздо удобнее, и это приводит к лучшим результатам, чем представление слов методом one-hot. Более того, оказывается, что такие векторы часто обладают интересными свойствами: простые геометрические соотношения между

Частота употребления слов по соседству друг от друга

	Антилопа	Лев	Трава	Баобаб
Антилопа	—	6	5	1
Лев	6	—	0	0
Трава	5	0	—	4
Баобаб	1	0	4	—

Слова с числовыми координатами

Антилопа (6, 5, 1)
Лев (6, 0, 0)
Трава (5, 0, 4)
Баобаб (1, 0, 4)

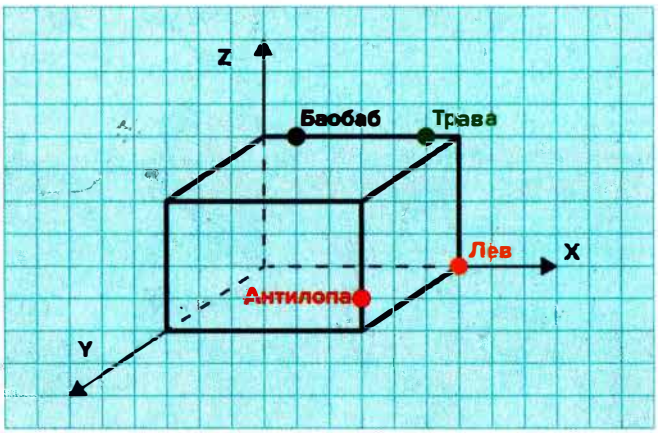


Рис. 13

Пример обработки естественного языка нейросетью

ними могут иметь грамматический или семантический смысл. Например, если вычесть вектор слова «художник» из вектора слова «художница», получится абстрактное представление о роде, выраженное в числах. А если эту разность прибавить к вектору слова «кот», то получится, скорее всего, «кошка».

Попытки сокращения размерности привели к тематическому моделированию. Как оно выглядит? Мы посмотрели на вероятностное распределение слов в текстах и выяснили, что какие-то наборы слов встречаются примерно с одинаковой вероятностью. Условно книги можно разбить по темам: «любовь», «секс», «вздохнула» — роман, «убийца», «труп», «жертва» — детектив, «я», «помню», «делал», «страдал» — автобиография. Программисту выдают эти распределения слов, а он понимает, что это за темы.

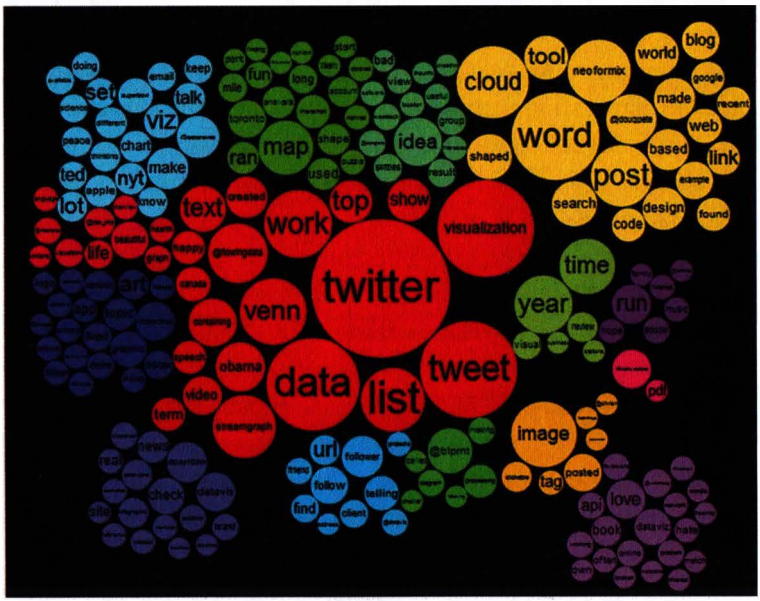


Рис. 14

Облако тегов показывает, как часто употребляются те или иные слова в группе интернет-страниц

Глубокое обучение пришло в обработку естественных языков в виде рекуррентных нейронных сетей[38]. Базовые рекуррентные архитектуры успешно использовались (и до сих пор используются) для задач классификации в обработке естественных языков. В частности, для задачи анализа тональности — это класс методов контент-анализа в компьютерной лингвистике, который позволяет автоматически выявлять в текстах эмоционально окрашенную лексику и то, как автор оценивает те объекты, события и явления, о которых говорится в тексте.

> **Проанализировав стихотворение «Зимнее утро» («Мороз и солнце, день чудесный...»), компьютер может сделать вывод, что у Пушкина было хорошее настроение, когда он его писал.**

Для машинного перевода, диалоговых агентов и других задач, где нужно не только на входе, но и на выходе получить последовательность текстовых единиц, обычно используются архитектуры вида «кодировщик — декодировщик» (encoder — decoder). Грубо говоря, энкодер сворачивает текст в формат вектора, выжимая из него всю важную информацию, а декодер разворачивает вектор обратно в текст. Затем появились архитектуры, которые дополняли базовую «кодировщик — декодировщик» механизмами внимания. «Кодировщик — декодировщик» весь контекст предложения закладывает в один вектор и, исходя из этого контекста, переводит предложение на другой язык, причем формирует контекст по прямой последовательности: самое первое слово к концу предложения уже теряет влияние на общий смысл. Например, нам надо

---

[38] Класс нейронных сетей, которые хороши для моделирования последовательных данных, таких как временные ряды или естественный язык. На вход подается не весь текст, а слово за словом плюс результат работы сети над предыдущим словом, что помогает понять контекст.

перевести предложение «Студент прочитал два учебника, тетрадь однокурсника и пост на "Хабре", но все еще ничего не понял про машинный перевод». Чтобы правильно распознать слово «понял» (а не «поняла»), нужно помнить, что в контексте стоит вектор слова «студент», а не «студентка». Нейросеть без внимания будет «отвлекаться» на векторы «учебников», «тетради» и т.д., и это будет мешать ей принять правильное решение.

Этот подход нормально работает на коротких фразах, но когда у нас, например, роман Толстого с предложениями на полстраницы, смысл начинает теряться. В этот момент вводят механизм «внимания»: он учитывает контекст всех слов в предложении, и таким образом для перевода первого слова вы можете взять контекст первого слова, второго, третьего и т.д. Не надо запоминать сразу все предложение. Это позволило значительно увеличить длину допустимого входа и выхода (то есть вводной последовательности текстовых единиц и того, что получается в итоге) для такой архитектуры. Так работала, в частности, система Google Neural Machine Translation (GNMT), ставшая основой для сервиса Google Translate.

Однако как раз примерно в конце 2017 г. в области произошла очередная мини-революция: появились архитектуры, основанные на самовнимании — варианте механизма внимания, в котором обработка части входа (например, слова из предложения) происходит при помощи контекста, когда с разными весами внимания учитываются другие части того же входа (другие слова из того же предложения). Самовнимание помогает связать слова внутри предложения и понять, как они друг с другом соотносятся. Базовая архитектура с самовниманием получила название Transformer. При каждом новом переводческом предсказании Transformer фокусируется только на тех словах, которые считает самыми важными. Проще говоря, внимание — это умножение вектора слова на числа (веса внимания) в соответствии с их

значимостью для понимания всего предложения: если слово важное — множители будут большими. На основе Transformer возникли языковые модели GPT (Generative Pretrained Transformer), BERT, GPT-2 и др.

Можно ли рассчитывать на то, что современные языковые модели пройдут тест Тьюринга? Есть примеры, которые на первый взгляд выглядят очень многообещающими. Модель GROVER, основанная на GPT-2, предназначена для распознавания поддельных новостей (fake news), но содержит и порождающую модель, которая может такие новости писать. По результатам экспериментов оказалось, что языковая модель GROVER способна писать тексты, которые оцениваются людьми более высоко, чем настоящие fake news, написанные живыми людьми на соответствующих веб-сайтах! Уже сейчас в информационных агентствах (например, Bloomberg News и Associated Press) используются «виртуальные новостники», преобразующие представленные в том или ином формальном виде факты в короткие новостные заметки.

Но это все еще не означает ни полноценного прохождения теста Тьюринга, ни скорого создания AGI. Если попробовать поговорить с современным диалоговым агентом, основанным на тех же базовых языковых моделях, станет очевидно, что пока он совершенно «не понимает», о чем говорит. «Не понимает» взято в кавычки, потому что определить «понимание», конечно же, проблематично; но в данном случае речь идет о том, что диалоговые агенты не могут отвечать на уточняющие вопросы, очень условно способны поддерживать тему и в целом не выдерживают никакой серьезной проверки.

Это происходит главным образом потому, что, хотя современные языковые модели прекрасно обучаются собственно языку, точнее грамматическим и семантическим зависимостям между отдельными словами, и способны учитывать эти зависимости в течение длительного времени (например,

развивать в порождаемом тексте одну и ту же тему на протяжении нескольких абзацев, как GROVER), они все же не содержат никакой модели окружающего мира, то есть базовых знаний, уложенных в здравый смысл, имеющийся у каждого человека.

Как обучить такую модель, пока не вполне ясно, и это остается одной из центральных проблем в обработке естественных языков, которая пока не позволяет двигаться в сторону AGI.

Нейросетевые модели работают с токенами (текстовыми единицами), «смысл» которых выводится сугубо из статистик совместной встречаемости в текстах. Можно дискутировать о том, возможно ли в принципе только из текстов извлечь представление о физическом мире, достаточное для «понимания» пространственно-временных и причинно-следственных отношений между объектами, но существующие NLP-модели этого точно не делают. Современный чат-бот вряд ли сможет сказать, с какой стороны ему виден большой палец на правой руке стоящего напротив собеседника, или ответить, как вернуться назад, если вы прошли три квартала на запад, а потом четыре на север. В связи с этим интересны задачи, которые помимо текста используют и другие модальности, например задача визуальных диалогов, и которые предполагают необходимость построения если не модели мира, то хотя бы какой-то привязки текстовых единиц к реальности.

Глубокое обучение с подкреплением: шахматы, го и AGI

Обучению с подкреплением как подходу к AGI посвящен отдельный раздел (см. главу 4), поэтому здесь мы не будем давать подробный анализ этого направления, а лишь сделаем несколько выводов и предположений.

Глубокое обучение с подкреплением на первый взгляд кажется направлением, самым близким к AGI и уже вплотную к нему подобравшимся. Например, впервые обыгравшая

одного из лучших игроков в го Ли Седоля модель AlphaGo была создана специально для го и использовала базу партий профессионалов для предобучения. Но с тех пор появилась модель AlphaZero, которая обучалась играть еще лучше без внешних данных, играя только сама с собой, и MuZero, которая достигла уровня AlphaZero, не зная даже правил игры (то есть обучая модель окружающей среды с нуля).

MuZero может обучиться играть не только в шахматы или го, но и в другую игру, даже не зная ее правил (она успешно тестировалась, например, в разных компьютерных играх Atari), что в некоторой мере преодолевает основной недостаток шахматных программ с точки зрения AGI — их узость. Но делает она это при помощи огромного количества вычислений, играя сама с собой миллиарды раз: для симуляции партий при обучении использовалась тысяча процессоров Google TPU[39]. В целом обучение с подкреплением оказывается успешно применимо в тех случаях, когда отдельные эксперименты дешевы и можно методом проб и ошибок постепенно улучшать стратегию в течение миллионов и миллиардов эпизодов обучения. То есть на самом деле для обучения с подкреплением требуется очень, очень много данных, просто так получается, что в подходящих для него задачах эти данные модель может породить для себя сама. Тем самым мы можем «обменять» подготовку и разметку данных на огромное количество вычислений, и для современного ИИ это хороший обмен, потому что вычисления в наше время обходятся куда дешевле.

Однако человек учится совсем не так.

---

[39] Тензорный процессор, разработанный корпорацией Google и предназначенный для использования с библиотекой машинного обучения TensorFlow. Тензорный процессор — это разновидность микросхемы, оперирующей особыми объектами (тензорами). По сравнению с графическими процессорами, он рассчитан на больший объем вычислений с пониженной точностью.

> **Интеллект выражается в том числе (а может быть, главным образом) в том, чтобы обучаться делать выводы по небольшому набору данных.**

Ни Гарри Каспаров, ни Магнус Карлсен не сыграли миллиарды и даже миллионы партий, но зато смогли успешно интернализировать, совместить и улучшить эвристики, которые выдавались им в готовом виде: знания из книг по шахматам, занятия с тренерами и т.п. Оказалось, что в случае такой замкнутой конечной системы, как шахматы, мы можем сделать так много вычислений, что использовать эти знания будет необязательно — модель сможет все понять из симуляционных экспериментов.

Но хватит ли нам процессоров TPU для того, чтобы обучить хотя бы автономный автомобиль с достаточной способностью к обобщению? И на каких данных это будет происходить? Синтетические среды для обучения автомобильных RL-агентов (например, Voyage Deepdrive) активно используются и процветают. Но если MuZero, обученную играть в шахматы, попытаться переучить хотя бы, скажем, на шахматы с «кентавром», где ферзь может ходить еще и как конь, ей придется снова играть миллионы партий, обучаясь правильно использовать новую фигуру. Получается, мы возвращаемся к тому, что в виртуальной среде для обучения беспилотного автомобиля обязательно нужно реализовать все возможные объекты, с которыми надо научиться взаимодействовать? Несмотря на все действительно важные успехи, удовлетворительного ответа на этот вопрос глубокое обучение с подкреплением пока не дает. Итак, к сожалению, говорить о реалистичном достижении AGI в рамках современной волны развития нейронных сетей пока не приходится.

Искусственные нейронные сети в настоящее время быстро приближаются к насыщению с точки зрения вычислительных мощностей и размера доступных наборов данных. С расцветом

глубокого обучения вычислительные мощности, требующиеся для обучения лучших моделей, начали расти гораздо быстрее, чем раньше. До 2012 г. этот рост в целом соответствовал закону Мура[40]: требования удваивались в среднем каждые 18 месяцев. А в 2012–2019 гг. удвоение вычислительных мощностей наблюдалось в среднем каждые 3,4 месяца! Разумеется, такой рост не может продолжаться вечно.

С точки зрения данных ситуация еще более сложная: хотя размеры датасетов продолжают расти, их ручная разметка становится слишком дорогой. Есть несколько направлений для исследований, которые выглядят перспективно и могут потенциально решить эту проблему — по крайней мере для части задач:

- модели, способные обучаться на очень малом числе примеров или вовсе без них (few-shot, one-shot и даже zero-shot learning);
- модели, которые используют неразмеченные наборы данных для предобучения, — например, исследователи Google Brain в 2019 г. смогли улучшить качество классификации на ImageNet, использовав в качестве дополнительной информации большой датасет неразмеченных фотографий. Однако такие подходы, как правило, требуют на порядки больше вычислительных мощностей, что, как мы только что видели, вряд ли сможет и дальше неограниченно масштабироваться;
- использование для обучения синтетических данных — например, для задач компьютерного зрения можно создавать искусственные наборы данных с идеальной разметкой при помощи 3D-графики. Однако при этом возникает дополнительная задача переноса обученных

[40] Закон Мура гласит, что количество транзисторов, размещаемых на кристалле интегральной схемы, удваивается каждые 18 месяцев. Соответственно, мощность компьютеров растет в геометрической прогрессии.

моделей: идеально реалистичной 3D-графики пока не существует, и нельзя без дополнительных ухищрений рассчитывать, что обученная на «мультфильмах» модель будет хорошо работать на настоящих фотографиях.

В целом, хотя прогресс глубокого обучения продолжается и охватывает все новые области, пока неясно, может ли текущая волна хайпа вокруг нейронных сетей привести к созданию AGI относительно эволюционным путем или для этого потребуются какие-то радикально новые идеи и прорывы.

Когнитивные архитектуры

Когнитивные архитектуры как подход к AGI

Как мы отмечали, в области общего ИИ понятие интеллекта определяется минимально антропоморфно. Это, однако, вовсе не означает, что учет сведений о работе естественного интеллекта чужд области общего ИИ. Напротив, одним из основных подходов в этой области являются как раз когнитивные архитектуры. Суть этого подхода — в моделировании когнитивных функций человека с той или иной степенью психологической или биологической правдоподобности. Когда мы моделируем ту или иную функцию (например, память), мы выделяем ключевые подсистемы, которые, по данным исследований в области когнитивных наук, участвуют в работе данной функции. Затем мы строим ее математическую модель и рассматриваем все виды взаимодействия этих подсистем. Ключевая гипотеза заключается в том, что именно в результате взаимодействия подсистем и возникает эффект эмерджентности, позволяющий выполнять функцию. Бен Герцель дал специальное название

этому эффекту — когнитивная синергия. Согласно этой гипотезе, разнообразные подсистемы общего ИИ, работающие с информацией разного вида, должны взаимодействовать таким образом, чтобы помогать друг другу в преодолении комбинаторного взрыва[41].

Отсюда вытекают основные задачи при создании когнитивной архитектуры:

1) **определение списка моделируемых функций (в него могут входить распознавание, категоризация, внимание, планирование поведения, обучение, рефлексия, рассуждения и т.д.);**
2) **определение набора моделей или теорий «первого уровня» (например, нейрофизиологические данные о строении неокортекса или когнитивная теория внимания), которые будут служить источником основных гипотез при моделировании выбранных функций;**
3) **выделение основных подсистем, определяющих работу выбранных функций, на основании сформулированных гипотез;**
4) **построение математических моделей работы выделенных подсистем и разработка «протокола коммуникации» подсистем в рамках общей архитектуры;**
5) **добавление недостающих компонент для погружения в экспериментальную среду, в которой будет проводиться тестирование созданной архитектуры, то есть сравнение работы моделируемых ею когнитивных функций с тем, как эти функции работают у человека.**

---

[41] Термин, используемый для описания эффекта резкого («взрывного») роста времени выполнения алгоритма при увеличении размера входных данных задачи.

Этот перечень задач особенно характерен для BICA (Biologically Inspired Cognitive Architectures).

> **В области AGI биологическая или психологическая правдоподобность является не основным, а вспомогательным критерием, так как цель здесь заключается не столько в моделировании человеческого мышления, сколько в создании систем, способных решать широкий круг задач.**

Причем решать их совсем не так, как это делает человек. В этой связи в ряде архитектур за основу берется некая математическая либо концептуальная теория решения задач или достижения целей в средах без явной привязки к богатым эмпирическим данным о естественном интеллекте. Примером может служить когнитивная архитектура NARS, построенная вокруг теории неаксиоматической логики и на убеждении, что в основе интеллекта лежит способность адаптироваться к окружающей среде при работе с недостаточными знаниями и ресурсами. Соответственно, интеллектуальная система должна опираться на конечный процессинговый ресурс, быть открытой для неожиданных задач, работать с недостаточными знаниями и ресурсами и учиться на собственном опыте. Соответствуя этим критериям, NARS пытается воспроизвести многие когнитивные функции, включая рассуждения, обучение, планирование и т.д.

Иногда архитектуры могут создаваться из сугубо инженерных соображений в попытке объединить в единое целое существующие практические решения в области компьютерного зрения, представления знаний и рассуждений, планирования, управления движением и т.д. При этом на первый план выходит интегративный характер таких систем, которые по-прежнему могут называться когнитивными архитектурами, поскольку естественным образом оказываются состоящими из ряда компонент, отражающих (хотя и на весьма

отвлеченном уровне) те или иные когнитивные функции человеческого интеллекта.

Понятие когнитивной архитектуры синонимично понятию интеллектуального агента и подразумевает воплощение полного агента, способного самостоятельно действовать в некотором окружении, достигая определенных целей или удовлетворяя свои потребности. Когнитивная архитектура включает набор систем памяти, которыми обладает человек. Перечислим основные из них.

- Рабочая память — когнитивная подсистема, отвечающая за временное хранение информации, доступной для обработки. Она играет важную роль в принятии решений и выборе поведения (например, когда вы выбираете из дел на сегодня приоритетное).
- Эпизодическая память — отвечает за наши личные воспоминания (например, помогает вспомнить, что мы ели на завтрак или как провели выходные).
- Декларативная память — тип долговременной памяти, в которой, как некий «багаж», осознанно сохраняются имеющийся у нас опыт или информация (например, когда вы решаете квадратное уравнение на экзамене и вспоминаете необходимую формулу).
- Семантическая память — система декларативной памяти для хранения и использования обобщенных знаний о мире. Благодаря ей мы помним, что в сутках 24 часа, а Лондон — столица Великобритании. Эта память хранит как вербально выраженные знания, так и взаимосвязи между ними и правила их применения.
- Процедурная память — вид недекларативной (неосознаваемой) долговременной памяти, в которой сохраняется опыт выполнения нами каких-либо действий. Например, там «хранятся» наши навыки плавания и езды на велосипеде. При этом мы не можем объяснить словами, что именно нужно

делать, чтобы поехать на велосипеде, но почувствуем это интуитивно, как только на него сядем.

В основе работы когнитивных архитектур лежит когнитивный цикл, включающий как минимум восприятие, осмысление, принятие решений и контроль над выполнением действий

Причем цикл может выполняться параллельно и асинхронно. Можно сказать, что представленная на рисунке 15 модель содержит минимальный набор блоков, которые должна включать когнитивная архитектура. Данный набор нельзя назвать однозначным. К примеру, объединение блоков памяти с блоками обучения снижает детализированность модели, хоть и может быть мотивировано нейрофизиологией и когнитивной психологией.

Если когнитивная архитектура включает достаточно широкий спектр моделируемых когнитивных функций и предназначена для решения широкого круга принципиально различных задач, можно считать ее реализацией общего искусственного интеллекта. Однако многие когнитивные архитектуры

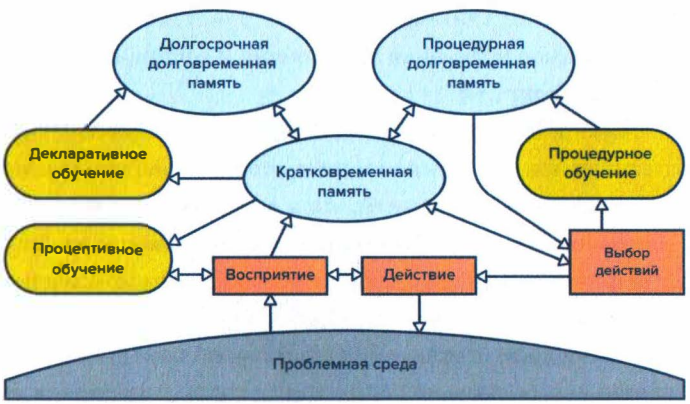


Рис. 15

Схема когнитивной архитектуры

не претендуют на высокую степень универсальности и предназначены для моделирования вполне конкретной функции, например приобретения знаний без учителя (HTM), восприятия и внимания (ART), формирования правил поведения на основе изображений (Clarion), памяти (ACT-R), эмоций (eBICA) и т.д.

Стоит отметить стандартное деление когнитивных архитектур на *символьные* и *эмерджентные*, что в целом отражает традиционную дихотомию подходов в ИИ — символьный и коннекционистский. При этом эмерджентные архитектуры в некоторой степени предвосхитили глубокое обучение, которое, однако, в силу практических успехов привлекло гораздо больше внимания и в рамках которого эмерджентные архитектуры сейчас переоткрываются под названием нейрокогнитивных архитектур. В этой связи большее развитие получили символьные когнитивные архитектуры, в ходе которого они стали преимущественно гибридными.

Как и многие другие методы, опирающиеся на системный подход, когнитивные архитектуры не всегда реализуются на уровне софта и часто остаются на концептуальном уровне, на котором описаны основные подсистемы и их функции. Далеко не все из них доведены до программной реализации и тем более до демонстрации работы в сложных условиях. И многие, несмотря на гибридность, не решают ключевую проблему общего искусственного интеллекта — проблему привязки символов (то есть соотнесения теоретической информации с реальной жизнью).

Наибольший пик интереса к когнитивным архитектурам отмечается в 2000–2010 гг., когда наблюдалась максимальная активность в разработке большого количества различных вариантов. Среди них можно выделить наиболее старые и универсальные — SOAR и ACT-R, которые развиваются до настоящего времени и которые в целом можно назвать в некотором приближении системами AGI. Данные примеры являются психологически правдоподобными. Примеры биологически

правдоподобных архитектур — ART и HTM, претендующие на моделирование менее широкого спектра функций, но в то же время более формализованных и системных, чем широкомасштабные модели мозга.

На текущий момент стоит отметить снижение интереса к когнитивным архитектурам, что вызвано двумя основными причинами. Первая заключается в слабой интеграции субсимвольных и символьных подходов, особенно в настоящее время, когда успехи субсимвольного нейросетевого подхода могут вызвать впечатление, что «символьная добавка» не играет ключевой роли для моделирования даже сложных функций (например, рассуждения на основе глубоких нейронных сетей). Вторая проблема состоит в том, что существует определенная произвольность в выделении ключевых подсистем, зачастую из-за большой неточности теорий об устройстве человеческого интеллекта. Тем не менее ряд когнитивных архитектур продолжает активно развиваться, в том числе решая и практические задачи.

Гибридные когнитивные архитектуры

Современные гибридные когнитивные архитектуры могут быть устроены по-разному: от соединения разнородных компонентов через ограниченные интерфейсы в гетерогенную систему до архитектур без четко выделенных подсистем, в которых гибридизация осуществляется через унификацию разных представлений. К последним, в частности, относятся архитектуры Sigma и STRL. Рассмотрим STRL в качестве примера подробнее.

Основная моделируемая функция в этой архитектуре — управление сложным физическим объектом (например, дроном) в условиях коллективного взаимодействия при решении общих и частных задач. Она состоит из трех уровней: стратегического, тактического и реактивного (архитектура STRL: strategic, tactical, reactive layered). На стратегическом уровне используется знаковое представление знаний и осуществляется обмен сообщениями с остальными участниками

коалиции. Тактический и реактивный уровни содержат модули, поддерживающие эти процедуры и транслирующие их управление в низкоуровневые управляющие сигналы.

На стратегическом уровне для планирования сложного поведения, распределения ролей, целеполагания и рефлексии используется теория знаковой картины мира, основанная на культурно-историческом подходе Выготского[42] и теории деятельности Леонтьева[43]. В качестве основного активного элемента картины мира предлагается использовать понятие знака. В модели знаковой картины мира когнитивный процесс представляет собой последовательность активации (или образования) знаков, между которыми есть логические связи. Существенным отличием предлагаемого механизма распространения активности от представленных в работах по искусственному интеллекту является взаимодействие четырех типов сетей.

Гетерогенные когнитивные архитектуры нередко также основываются на некоторой унифицированной модели, но в ходе своей практической реализации и дальнейшего развития приходят к использованию более эффективных, но менее унифицированных компонентов. Примером может служить OpenCog. Хотя OpenCog пытается воплотить концепцию когнитивной синергии за счет взаимодействия всех его подсистем через общее пространство для представления и хранения знаний в форме гиперграфа[44], но каждая из подсистем мо-

---

[42] Согласно теории выдающегося советского психолога Льва Выготского, развитие мышления и других психических функций происходит в первую очередь не через их саморазвитие, а через использование ребенком «психологических орудий», путем овладения системой знаков-символов, таких как язык, письмо, система счета.

[43] Алексей Леонтьев — советский психолог, философ, педагог и организатор науки. Он считал, что психика человека формируется во время активности и в процессе работы и что в них же она проявляется.

[44] Граф, в котором связи могут соединять не только узлы, но и другие связи, причем одна связь может соединять не только два, а сразу несколько узлов или связей.

жет быть основана на собственных принципах. В частности, OpenCog включает как подсистему вероятностных логических сетей, реализующих символьные рассуждения над декларативными знаниями, так и глубокие нейронные сети, опирающиеся на стандартные внешние библиотеки глубокого обучения. Символьные и субсимвольные компоненты этим не исчерпываются и включают также подсистему эволюционного программирования (об этом типе методов мы поговорим в следующей главе), экономические сети управления вниманием и другие компоненты, за каждой из которых стоит своя теория.

Гетерогенная гибридизация, хоть и повышает практичность когнитивных архитектур, в то же время снижает эффект когнитивной синергии и требует теоретического переосмысления для более глубокой интеграции компонентов.

Стоит отметить, что описание каждой когнитивной архитектуры может занимать сотни страниц и включать множество схем. Одна из проблем подхода на основе когнитивных архитектур заключается в том, что каждая архитектура развивается практически независимо и очень немногие результаты являются общезначимыми.

Универсальный алгоритмический интеллект

Шире некуда

Общий ИИ — это способность эффективно действовать в широком диапазоне сред. Но для какого диапазона сред предназначены сети глубокого обучения или когнитивные архитектуры? Зачастую четкого ответа на этот вопрос нет. И для каждой

модели или системы легко найти задачу, даже несложную, которую она решить не сможет в принципе.

Но можно ли вообще как-то охарактеризовать наиболее широкий диапазон сред и дать хоть какие-то гарантии, что тот или иной метод будет в этом диапазоне работать? Как мы отмечали в предыдущей главе, самый широкий диапазон сред, с которым мы можем работать, — это все среды, которые можно описать алгоритмически. Особо подчеркнем, что, говоря о таких средах, мы не сужаем, а лишь расширяем их диапазон по сравнению со всеми существующими методами узкого ИИ и даже большинством методов общего ИИ.

Рассмотрим простой пример — предсказание стороны, которой выпадет монетка. Если монетка обычная, то мы скажем, что она выпадет любой из сторон с 50%-ной вероятностью. Но допустим, мы наблюдаем такую последовательность выбрасываний: 00001001000010, где 0 — это орел, а 1 — решка. Монетка покажется нам необычной. Но вдруг она просто изогнута? Тогда вероятность выпадения ее той или иной стороной будет разной. Мы можем учесть это в своем представлении о монетке, в ее модели. Пусть вероятность будет разной и у нас есть возможность оценить ее путем наблюдений. Теперь представим, что результат серии подбрасываний — 010101010101. Вероятность 50%! Совершенно обычная монетка? Но мы очень удивимся, увидев такое в реальности. Почему? Мы наблюдаем в этой последовательности закономерность, которая не описывается нашей моделью.

Мы можем расширить нашу модель, предположив, что результат последующего подбрасывания зависит от предыдущего. Для физической монетки это не так просто сделать, но чуть проще — для кубика, у которого внутри есть скрытый механизм, сдвигающий его центр тяжести на следующую грань после каждого подбрасывания. По последовательности выпадений мы можем построить модель этого механизма. Но что, если последовательность 001100110011? Орел и решка

следуют друг за другом в 50% случаев, как и у нормальной монетки. Но монетка неправильная! Мы видим закономерность. Очевидно, можно расширить нашу модель, построив таблицу с данными, после каких двух выбрасываний выпадет орел, а после каких — решка. Например, 1 идет всегда после 00 и после 01. А вот после 11 и 10 всегда идет 0.

Но пойдем еще чуть дальше и посмотрим на последовательность 010011000111... Она опять в нашу модель не вписывается, хотя она все еще элементарна для человека (точнее, для взрослого, но не для маленького ребенка). Удивительно, но даже уже такую последовательность хорошо предскажет далеко не любой алгоритм машинного обучения. Разработчик видит закономерность в этой последовательности и может нужный тип закономерности заложить в свою модель, чтобы та могла эту закономерность представить, обнаружить и предсказать. Именно этим традиционно и занимались специалисты по машинному обучению.

> Глубокое обучение, на первый взгляд, освобождает человека от необходимости задавать класс моделей, но это во многом иллюзия. Глубокая сеть не пытается обнаружить закономерность, а лишь аппроксимирует ее.

Разница принципиальна. Нейронная сеть может запомнить примеры. Конечно, она не просто запоминает — она интерполирует. Чем больше примеров, тем точнее интерполяция, но она никогда не станет абсолютно точной, если класс закономерностей, с помощью которых выполняется приближение, не содержит приближаемую закономерность. Так, приближая синусоиду многочленом по конечному числу точек, мы не сможем получить точного решения. Мы будем брать все больше и больше точек, и качество интерполяции будет улучшаться, но экстраполяция будет оставаться плохой. Так же и в нашем последнем примере нейросеть не сможет

сделать нужного обобщения, экстраполяции на много шагов дальше, чем ей показывали.

Для многих частных прикладных целей обучение по многим примерам с интерполяцией бывает приемлемым решением, но для общего ИИ этого явно недостаточно. Взглянем на один исторический пример — геоцентрическую модель движения планет Солнечной системы, предложенную Птолемеем (рис. 16). Если наблюдать за звездами, то кажется, что они движутся по окружностям вокруг Земли. Но некоторые из светил перемещаются неравномерно. Это планеты. Птолемей предположил, что они движутся по малым кругам — эпициклам, центры которых передвигаются уже по большим окружностям вокруг Земли. Это позволило объяснить неравномерность и долго согласовывалось с наблюдениями. Но по мере улучшения астрономических приборов обнаруживались

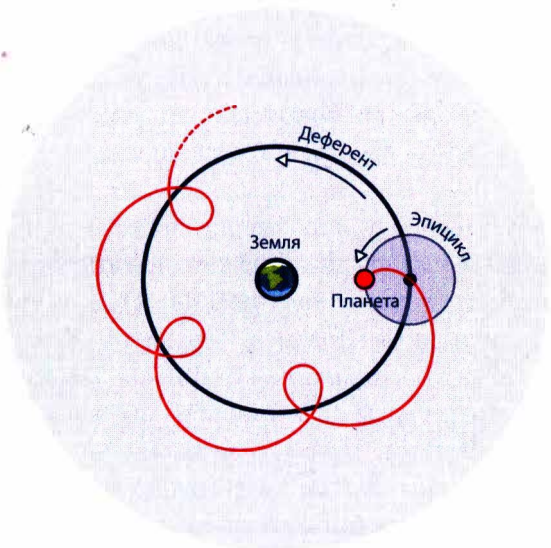


Рис. 16

Эпициклы в представлении Птолемея

систематические отклонения. Для их объяснения стали вводить целые системы эпициклов: планеты в этой модели двигались по кругам, центры которых перемещались по кругам, которые двигались по кругам и т.д. Модель была громоздкой, но она позволяла неплохо интерполировать движение планет. Однако если бы астрономы тогда попытались по нескольким точкам предсказать передвижения, скажем, нового астероида, они бы потерпели неудачу.

Кеплер описал движение планет вокруг Солнца по эллипсам в рамках гелиоцентрической концепции Коперника. Модель не только получилась гораздо проще и не только позволила предсказывать траектории новых объектов по малому числу точек — она также помогла Ньютону вывести формулу закона всемирного тяготения. Системы эпициклов использовались веками для практических нужд. Также и современные аппроксимационные модели машинного обучения могут быть вполне полезны. Но от общего ИИ мы ожидаем таких же прорывов, как те, что совершили Кеплер, Ньютон, Эйнштейн или, например, Менделеев, выявивший новые закономерности, — прорывов, на которые аппроксимационные модели, работающие в ограниченном пространстве закономерностей, не способны.

Модель Кеплера, как и любые другие модели, которыми до сих пор пользовались люди, можно описать алгоритмически. И модели для порождения 010011000111… и для расчета синусоиды, и эпициклы Птолемея, и уравнения Кеплера — все это можно найти в пространстве алгоритмов. Там даже можно найти алгоритм, который напечатает столько знаков числа пи, сколько нам захочется. Значит, если машинная система сможет находить в данных любые алгоритмические закономерности, уже нельзя будет сказать, что она может делать только то, что в нее заложено, — она будет столь же универсальна (если не более), как и человек.

Информационная стоимость моделей

Но алгоритмов бесконечно много. Какой из них считать лучшей моделью для описания данных? Это давняя проблема для индуктивного вывода[45] в целом, а не только для случая алгоритмических моделей. В середине прошлого века для ее иллюстрации Нельсон Гудмен предложил парадокс «зелубых изумрудов». Суть его в следующем. Рассмотрим две гипотезы: 1) все изумруды зеленые; 2) все изумруды «зелубые», то есть зеленые до, скажем, 2050 г., а потом они вдруг становятся голубыми. Проблема в том, что все данные наблюдений абсолютно точно описываются обеими гипотезами. Так какую из них выбрать и на каком основании? Сейчас мы бы назвали выбор второй гипотезы переобучением, которое все еще составляет проблему для современных методов машинного обучения.

Интересно, что идея рассматривать множество алгоритмов как универсальное множество моделей для машинного обучения была предложена совместно с критерием выбора моделей, решающего проблему переобучения, еще в 1960-е гг. Рэем Соломоновым[46] — одним из участников Дартмутского семинара. Предложенный им метод носит название универсальной индукции и предсказания.

Суть в следующем. Пусть у нас есть некоторая последовательность, которую мы хотим продолжить. Рассмотрим все алгоритмы, которые печатают эту последовательность и некоторое ее продолжение. Например, алгоритм «повторять печать 01» напечатает последовательность 0101 с продолжением 010101..., а алгоритм «печать 01011111» напечатает

---

[45] Индуктивный вывод — рассуждения от частного к общему; в общем смысле — выявление регулярностей в наблюдениях и фактах, нахождение для них объяснений, построение моделей.

[46] Solomonoff, R. A Formal Theory of Inductive Inference, part 1 and part 2. Information and Control 7, 1–22, 224–254 (1964).

0101 с продолжением 11111. Каждый алгоритм — модель, согласующаяся с данными, но предсказывающая свое продолжение. Прямо как в «зелубых изумрудах». Мы не можем однозначно выбрать среди них, но мы можем назначить им разные вероятности.

Как мы уже упоминали при обсуждении метрик для общего ИИ, способ назначить непротиворечивые вероятности для алгоритмов можно как $2^{-L(p)}$, где $L(p)$ — это длина (число бит) программы p. То есть вероятность программы экспоненциально падает с ее длиной. Именно этот способ и предложил Рэй Соломонов одновременно с Андреем Николаевичем Колмогоровым и Грегори Хайтиным, хотя и немного с другой целью.

В то же время и Пер Мартин-Леф рассматривал вопрос, какие двоичные строки следует считать полностью случайными. Идя от противного, случайность — это отсутствие закономерности. И Хайтин тоже пришел к выводу, что общий способ описывать закономерности — через алгоритмы. Но что значит «отсутствие»? Какой-нибудь алгоритм да найдется. Но какой-нибудь нас не устроит. Мы посчитаем, что закономерность найдена, если алгоритм короче самой последовательности, то есть позволяет ее сжать. Действительно, если дана последовательность 1 2, мы интуитивно усомнимся, что ее продолжение обязательно 3 4 5. А вот если дана последовательность 1 2 4 8 16 32 64, то у нас уже будет гораздо меньше сомнений в ее продолжении, поскольку алгоритм, описывающий эту последовательность, короче ее самой (а алгоритмы, предсказывающие другое продолжение, — длиннее). Итак, случайную последовательность можно определить как алгоритмически несжимаемую последовательность. Но понятно, что, если мы возьмем более длинную версию последовательности, которая нам казалась случайной, она вдруг может оказаться вполне закономерной.

Колмогоров пытался найти такие основания понятия информации, которые не опирались бы на понятие вероятности, и, напротив, вывести вероятность из количества информации. Алгоритмическая сложность (количество информации) последовательности была определена Колмогоровым как длина кратчайшей программы, порождающей эту последовательность.

Таким образом, вместо того чтобы говорить, что мы назначаем моделям разные вероятности, мы можем сказать, что для построения этих моделей нужно разное количество информации. Наша последовательность может просто не содержать нужного объема информации для реконструкции некоторой модели, даже если эта модель в итоге окажется правильной. Упрощенно говоря, если нам дадут на плоскости две точки и попросят провести через них линию, мы меньше рискуем ошибиться, если проведем прямую. Даже если нам заранее скажут, что правильный ответ — парабола, не имея дополнительной информации, мы можем не угадать, в какую сторону она повернута. Мы просто не пытаемся выдумать те детали, для которых у нас нет информации. Именно поэтому данный подход в целом связывают с принципом бритвы Оккама — не следует плодить сущности сверх необходимого. Но в рамках алгоритмической теории информации он получает математическое обоснование.

> **В рамках универсальной индукции каждому алгоритму назначается вес, экспоненциально убывающий с длиной алгоритма.**

Для этого есть разнообразные обоснования. Тогда вероятность некоторого продолжения заданной последовательности можно вычислить как суммарный вес всех алгоритмов, порождающих эту последовательность с этим продолжением, деленый на суммарный вес всех алгоритмов, порождающих эту последовательность с произвольным продолжением.

Представим, что у нас есть последовательность 1 2 3. Поначалу у нас много сравнительно простых алгоритмов, которые порождают эту последовательность с разными продолжениями. Вероятность того, что следующее число 4, немногим больше вероятности, что далее следуют другие числа. Потом наша последовательность растет — 1 2 3 5 7, и некоторые модели, которые нам исходно казались более правдоподобными, отсеиваются. Потом, когда мы получаем еще несколько чисел, скажем 11 13 17 19, предсказание становится почти однозначным, поскольку остается одна сравнительно простая модель (если мы ее смогли найти), согласованная с данными, а все остальные модели будут гораздо изощреннее (вероятно, длиннее самих данных), а значит, будут обладать гораздо меньшим весом. Предсказание очень быстро, с небольшим увеличением объема данных, сойдется к истинному.

Крайне примечательно, что для универсальной индукции доказана сходимость к правильному предсказанию последующих элементов последовательности (которую в широком смысле можно трактовать как наблюдения среды агентом). И это при том, что ей абсолютно неизвестен класс моделей, среди которых следует искать закономерности в данных. При этом метод, специализированный под правильный класс моделей, вручную заложенный разработчиком, потребует лишь немногим меньшего объема данных, чем нужно универсальной индукции. Если же разработчик ошибется с классом моделей, то специализированная модель просто не сойдется к столь же качественному предсказанию, как универсальная индукция, сколь бы большой дополнительный объем данных ни дали узкой модели. Не существует метода предсказания, который бы работал не хуже универсальной индукции на всех последовательностях и строго лучше — хотя бы на одной. А вот более плохих методов много. Этот результат легко обобщается с источников данных, описываемых детерминированными моделями, на стохастические.

Стоит отметить, что из универсальной индукции и тесно связанной с ней алгоритмической сложности вытекает принцип минимальной длины описания для выбора моделей, получивший широкое применение и на практике и показывающий, что многие прикладные алгоритмы можно выразить через идею сжатия информации[47].

Универсальный интеллект

Универсальная индукция имеет большое самостоятельное значение для понимания проблем общего ИИ, но не является моделью интеллекта. Формальная теория креативности, веселья и внутренней мотивации, развитая Юргеном Шмидхубером в 1990-х гг. на той же теоретической базе, рассматривала, помимо обучения, и другие вопросы, связанные с интеллектуальными агентами. Однако формальная модель интеллекта на базе универсальной индукции, обозначаемая как AIXI, появилась лишь в 2000-х гг. в работах Маркуса Хуттера. Именно эта модель породила отдельное направление в области общего ИИ, хотя к данному направлению следует относить работы не только над ней и ее уточнениями, но таюке и над родственными моделями и их компонентами, включая универсальную индукцию саму по себе, составляющими в совокупности развивающуюся теорию универсального алгоритмического интеллекта.

Теория универсального интеллекта отталкивается от одного из определений общего интеллекта, принятых сейчас в общем ИИ, как «способности достигать цели в широком диапазоне сред» и ставит вопрос о том, можно ли дать математическую формализацию данного определения и построить на его основе искусственного агента. В данном определении

---

[47] Потапов А. С. Распознавание образов и машинное восприятие: общий подход на основе принципа минимальной длины описания. — СПб: Политехника, 2007. — 549 с.

отсутствует «предположение ограниченных знаний и ресурсов», на котором делает акцент Пей Ванг и которое критично для любого практического воплощения. «Ограниченность знаний» в действительности учитывается, так как в теории рассматривается агент, действующий в широком диапазоне сред без априорной информации о том, в какой именно среде он находится. Хотя можно аргументировать, что ограниченность ресурсов также автоматически накладывается каждой средой, но в базовой теории это не так, поскольку модель агента отделена от модели среды.

Идея подхода заключается в том, чтобы рассмотреть сначала идеального агента, доказать его оптимальность, а затем в нисходящем стиле разработать его более детальную ресурсно-ограниченную версию, пригодную для практических целей. Главным привлекательным свойством данного подхода является его строгость и систематичность, поскольку он наконец позволяет нам говорить об AGI в точных математических терминах и формулировать строго доказанные утверждения вместо того, чтобы ходить вокруг да около, используя расплывчатые понятия и пустые обещания.

Главная идея подхода может быть описана достаточно просто. Мы рассматриваем общую задачу обучения с подкреплением, в которой агент, взаимодействуя с некоторой средой, на каждом шаге выполняет действие и получает пару «наблюдение — награда», порождаемую средой. Цель агента — выбрать на каждом шаге действие, максимизирующее ожидаемые будущие награды. Но, в отличие от классического обучения с подкреплением, в данном подходе не делается никаких дополнительных предположений о среде, не накладывается никаких ограничений на то, как среда может быть описана (за исключением того, что интеллект реализуется на компьютере и для него самый общий доступный способ описания — произвольные программы, а физически реализовать какой-то более общий способ пока не получалось).

> **Универсальная индукция по истории взаимодействия агента со средой позволяет предсказывать будущие реакции среды именно в такой общей постановке. Она и используется как строительный блок AIXI. Второй блок — это выбор действий, максимизирующих награды.**

Если бы мы пассивно наблюдали за каким-то агентом, то непосредственно применили бы универсальную индукцию для продолжения цепочки пар «действие — реакция среды». Но AIXI вероятности реакции среды предсказывает, а свои действия выбирает. Модель строит дерево перебора всех своих действий и всех реакций среды; для узлов, в которых рассматриваются разные реакции среды, делает усреднение суммарной будущей награды с учетом вероятности того или иного продолжения, а для каждого узла выбора действия берет ту ветвь, для которой предсказывается максимальная будущая награда.

Хотя для модели AIXI не удается сформулировать столь сильные, как для универсальной индукции, свойства сходимости к оптимальному результату в общем случае, AIXI оказывается Парето-оптимальной: не существует агента, действующего не хуже AIXI во всех средах и строго лучше хотя бы в одной.

Развитие подхода

Универсальная индукция решает две основные проблемы машинного обучения, не решаемые вместе или по отдельности во многих прикладных методах: пространство моделей и критерий их выбора. Но она не пытается работать с третьей проблемой, которая как раз решается в практических методах в первую очередь, — со способом эффективного поиска моделей. AIXI, добавляя к универсальной индукции выбор действий для агента в среде, лишь усугубляет эту проблему.

Таким образом, недостаточностью моделей универсального интеллекта для создания реального общего ИИ, понятной

в равной мере как критикам данного подхода, так и его сторонникам, является неучет вычислительных ресурсов (и даже невычислимость моделей). Естественно, основные попытки развития подхода, предпринимавшиеся еще до его формирования как современного направления в общем ИИ, заключаются в устранении или ослаблении этого недостатка. Эти попытки варьируются от сугубо теоретических, когда осуществляется попытка построить модели с математически доказуемыми свойствами оптимальности при ограничении ресурсов, до эвристических и практических.

В первую очередь, конечно, нужно решить проблему невычислимости универсального интеллекта, которая проистекает из невычислимости проблемы остановок: перебирая произвольные алгоритмические модели, нельзя точно определить, какая из них остановится, а какая зациклится. Если мы говорим, что универсальный интеллект это как-то делает, то он это должен делать невычислимым образом. Получается странно: мы говорим, что все среды вычислимы, и при этом предлагаем невычислимую модель интеллекта. Любой критик сразу воскликнет: ваша теория — ерунда!

На самом деле решение проблемы невычислимости дал еще Соломонов, предложив использовать поиск по Леониду Левину. Алгоритм поиска следующий: запускаем по очереди все программы, начиная с самой короткой. Каждой программе даем сделать шаг, но чем короче программа, тем чаще она делает шаги (говоря точнее, каждая программа делает шаги с периодом, экспоненциально зависящим от ее длины). Побеждает программа, нашедшая решение первой. В случае универсальной индукции «найти решение» значит породить заданную последовательность (сам Левин предложил свой метод для решения произвольных задач). Если короткая программа зацикливается, нам это не помешает, так как решение сможет найти программа, которая не зацикливается, хоть она длиннее и реже делает шаги. Длинная программа, которая просто

содержит в себе печать заданной последовательности, очень быстрая, но до ее запуска дело может просто не дойти, если найдется подходящая более короткая и не слишком медленная программа. А если не найдется, будет выбрана длинная программа, и это будет означать, что последовательность случайная. Поиск по Левину не зациклится и выдаст ответ. Он вычислим. Но насколько он портит универсальную индукцию?

Можно утверждать, что он ее улучшает во всех отношениях. В качестве лучшей ее модели выбирается программа, обладающая не просто наименьшей длиной, а минимальной суммой длины и логарифма от времени выполнения. Если длину кратчайшей программы, порождающей последовательность, называют ее сложностью по Колмогорову, то эту величину можно назвать сложностью по Левину. Можно пытаться концептуально оправдать ее использование тем, что если наш мир наполнен случайными источниками данных, то априорно больше вероятность встретить данные от тех источников, которые работают быстрее. Но такое оправдание и не очень нужно: универсальная индукция с поиском по Левину все еще гарантированно сходится к оптимальному решению (при увеличении длины последовательности и при конечной сложности источника данных) и при этом становится вычислимой!

К сожалению, вычислимость вовсе не гарантирует практическую применимость. Перебор программ по Левину, как и полный перебор действий в AIXI (который можно считать вычислимым для ограниченного временного горизонта), требуют нереалистичного объема ресурсов. Например, программу в пару десятков символов таким перебором уже не найти.

В части универсальной индукции уже традиционной является идея, что мы можем нужные нам программы сделать более короткими (и гораздо более легко обнаружимыми), если выберем подходящий язык, на котором будем эти программы описывать. В теории этот выбор роли не играет: какой бы мы язык ни взяли, если он позволяет описывать любые алгоритмы

(является Тьюринг-полным), то универсальная индукция будет сходиться. Но на практике выигрыш даже в 10 бит будет означать выигрыш в 1000 раз по скорости поиска. Есть ли разница, решать задачу день или три года? В этой связи важную роль играет и инкрементное обучение: универсальная индукция начинает с простых задач, решает их, а обнаруженные закономерности добавляются в библиотеку функций, что позволяет компактнее представлять решение более сложных задач. Хотя это и важно, но кардинально проблему поиска в универсальной индукции не решает.

Очевидно, нужно уходить от полного перебора, и еще Соломонов предлагал использовать в универсальной индукции генетическое программирование, о котором подробнее будет сказано в следующей главе. Для сред низкой сложности оно работает на практике, но все же плохо масштабируется на сложные среды.

С несколько другой стороны на проблему взглянул Хуттер. В его оригинальной монографии[48] предлагается ресурсно-ограниченная модель AIXItl. Общая идея заключается в том, чтобы перебирать не сами программы, а доказательства того, что та или иная программа будет выполнена за гарантированное время, и в первую очередь выполнять те программы, для которых это доказательство удастся найти, что позволит избежать проверки и запуска заведомо плохих программ. Это даст возможность получить определенные свойства ресурсной оптимальности. В свою очередь, Юрген Шмидхубер пошел еще дальше и в 2003 г. предложил модель под названием «машина Геделя» (в честь вдохновившего его Курта Геделя), в которой делается попытка улучшить не только то, какие модели мы проверяем, но и сам процесс доказательства. Машина

---

[48] Marcus Hutter. Universal Artificial Intelligence: Sequential Decisions based on Algorithmic Probability. Springer, Berlin, 2005. 300 pages (http://www.hutter1.net/ai/uaibook.htm).

Геделя переписывает свой собственный код, если может доказать, что новый код обеспечивает лучшую стратегию. Обе эти модели ставят интересный вопрос о возможности использования рассуждений для ускорения процесса обучения. Кроме того, машина Геделя интересна как модель самооптимизирующегося интеллекта.

В некоторых исследованиях делается попытка разработать такую модель, в которой агент был бы частью среды, что автоматически приводило бы к учету ресурсной ограниченности[49]. К сожалению, всем указанным моделям, включая AIXItl и машину Геделя, не достает технических деталей (например, о том, как именно должна работать машина доказательств), достаточных для их практической реализации.

Другое направление — взять полностью определенные модели (как правило, это базовые модели универсальной индукции и AIXI) и выполнить их прямую реализацию с ограничением ресурсов (то есть с простым ограничением перебора или времени выполнения программ/моделей). Такие попытки предпринимались не раз, в том числе и с тщательным выбором языка, на котором описываются программы, что, как уже отмечалось, приводит к весьма ограниченным успехам.

Следующий шаг — добавить в поиск какие-то общие эвристики. Одна из центральных идей — использовать иерархические представления, то есть искать сначала небольшие программы, описывающие фрагменты или отдельные особенности данных, а потом из них составлять программы, описывающие уже данные в совокупности. Эта идея высказывалась в разных вариациях (универсальная сегментация[50], универ-

---

[49] L. Orseau, M. Ring. Space-Time Embedded Intelligence.

[50] Hewlett, D., Cohen, P. Artificial General Segmentation. Proc. 3rd Conf. on AGI. Advances in Intelligent Systems Research, vol. 10, pp. 31–36 (2010).

сальное иерархическое представление[51] и т.д.) в разных работах.

Одной из недавних реинкарнаций этой идеи с новыми интересными теоретическими результатами является теория инкрементного сжатия[52]. В ней выводится схема кодирования, содержащая описание последовательности в форме композиции различных признаков, или свойств, последовательности, которые могут быть найдены по отдельности. Примечательно существование имплементации этой схемы кодирования — WILLIAM, представляющей собой систему индуктивного программирования, которая ищет автоэнкодеры в виде синтаксических деревьев на Python, состоящих из ограниченного множества функций. WILLIAM достигает компрессии разнообразного множества простых последовательностей, а будучи дополненной expectimax-поиском, может решать простые проблемы, такие как выполнение градиентного спуска или игра в TicTacToe, не требуя специального программирования.

> **Другая нередко высказываемая идея заключается в том, что для практических нужд не требуется такой универсальности, которая предполагается моделями типа AIXI, поскольку распределение физически реализуемых сред заметно у́же.**

Эта идея может, однако, пониматься несколько по-разному. «Узость» распределения может быть связана с сильным смещением вероятностей сред при сохранении их универсальности, чего возможно достичь за счет выбора языка представления, хорошо согласованной с физической реальностью. При этом можно утверждать, что в реальности могут быть в принципе

---

[51] Potapov A., Rodionov S. Extending Universal Intelligence Models with Formal Notion of Representation. Proc. AGI 2012, Lecture Notes in Artificial Intelligence. 2012. V. 7716. P. 242–251.

[52] Franz A., Antonenko O., Soletskyi R. A theory of incremental compression. ArXiv: https://arxiv.org/abs/1908.03781, 2019.

реализованы (а значит, встречены) любые алгоритмические закономерности и что их вероятность не должна быть строго нулевой.

Более радикальная версия данной идеи предполагает не просто смещение вероятностей, а их сужение, переход ко все еще широким, но не универсальным пространствам моделей, полагающихся достаточными для практических нужд, что может делаться ограничением не языка представления моделей, а максимальной сложности независимо выделяемых регулярностей (или, например, признаков, как в системе WILLIAM). Последнее можно, однако, трактовать не как сужение пространства моделей, а как нахождение решения в универсальном пространстве моделей, но не точного, а приближенного. Интерес здесь представляет совместный учет различных приближений и предположений — локальности, многомасштабности, пространственно-временной устойчивости физических структур и т.д., — который мог бы привести к существенно более эффективным, но все еще общим аппроксимациям AIXI.

Наиболее известным примером практической аппроксимации AIXI, использующей суженное пространство моделей, является MC-AIXI–CTW, в которой модели представляются контекстными взвешенными деревьями ограниченной глубины, а выбор действия осуществляется поиском по методу Монте-Карло. Данная аппроксимация работает достаточно быстро, чтобы управлять агентом, способным играть в разные простые игры: mazes, Tiger, TicTacToe, Partially Observable Pacman, Kuhn Poker, Biased Rock-Paper-Scissors и т.д. На момент создания модель превосходила возможности любого другого подхода в этих простых, но весьма разнообразных средах, требующих и минимальных элементов стратегического мышления (что было тогда недоступно, например, моделям глубокого Q-обучения, играющим в игры Atari). Несмотря на успешность, данная модель поднимает вопрос, а действительно ли она является осмысленной аппроксимацией AIXI. Многие авторы полагают, что даже если

реальная аппроксимация AIXI не обладает (и, очевидно, не может обладать) фактической универсальностью при ограниченных ресурсах, то она должна обладать универсальностью в пределе (по ресурсам), чего нет у MC-AIXI–CTW.

> **Таким образом, попытки теоретического решения проблемы эффективного поиска в моделях универсального интеллекта пока далеки от практической реализуемости.**

Практически реализуемые аппроксимации вызывают неудовлетворенность из-за плохой масштабируемости или потери универсальности. Видимо, решение данной проблемы находится за рамками данного направления в его текущем состоянии и требует новых идей — возможно, из других направлений.

Попытки рассмотреть общую идею универсальных агентов с других сторон периодически предпринимаются, и наблюдающийся теоретический прогресс улучшает наше понимание проблемы универсального интеллекта, но имеющиеся работы разрозненны и редко надстраиваются над результатами предшествующих работ, а порой поднимают уже изученные вопросы.

Сильные и слабые стороны подхода

Наиболее сильная сторона данного подхода заключается в том, что он предоставляет нам систематический подход к общему ИИ. Он начинается с ясного математического определения максимально интеллектуального агента. Далее мы переходим к разработке его модификации, больше подходящей для практики, путем выдвижения предположений, которые могут не быть верными для произвольных данных, но верны для данных, порожденных окружающим нас миром. Эти предположения скорее не жестко ограничивают возможности агента, а лишь создают соответствующее индуктивное смещение для более эффективных версий AIXI. В частности, можно

аргументировать, что многие когнитивные функции человека оказываются ресурсно-ограниченной реализацией оптимального поведения универсального интеллекта[53].

Кроме того, нами уже отмечалась метрика UIQ как важный теоретический инструмент оценки коэффициента общего интеллекта в системах ИИ. Также модели семейства AIXI широко используются при изучении проблем безопасности общего ИИ. Этот подход является, пожалуй, единственным существующим строгим способом рассмотрения проблем безопасности, поскольку другие подходы не опираются на какую-либо модель общего ИИ и их выводы носят весьма спекулятивный характер.

Слабой стороной подхода является то, что он, вероятно, наиболее далек от практического использования в силу своей максимальной общности. Для достижения быстрого успеха в ограниченных областях, конечно, существуют более подходящие методы — не только узкие, но и в области общего ИИ. Однако в части достижения AGI вполне может оказаться, что, несмотря на кажущуюся скудность начальных результатов, данный подход способен превзойти менее универсальные подходы, поскольку последние могут оказаться тупиковыми из-за недостатка общности.

Развитию направления также препятствует малое число проектов. Наибольший вклад в область универсальных методов был внесен Р. Соломоновым и Ю. Шмидхубером, а непосредственно в теорию универсальных агентов — М. Хуттером, а также его студентами и коллегами: Joel Veness, Jan Leike, Laurent Orseau, Tor Lattimore, Shane Legg, Tom Everitt и рядом других, одни из которых в настоящее время работают с М. Хуттером в Австралийском национальном университете, а другие — в DeepMind. Существует небольшое число групп и отдельных исследователей, продолжительно занимавшихся

---

[53] Potapov A., Rodionov S., Myasnikov A., Begimov G. Cognitive Bias for Universal Algorithmic Intelligence // SarXiv:1209.4290v1 [cs.AI]. 2012.

вопросами универсального интеллекта и реализовывавших систематические исследовательские программы или проекты, например Gigamachine (Eray Özkural), Aideus (Алексей Потапов), а из недавних активных проектов — OCCAM (Arthur Franz).

Вероятностные модели

Вероятностное программирование — это способ создания систем, помогающих принимать решения в условиях неопределенности. Допустим, нам надо расследовать убийство — человек был застрелен на улице. Важно выяснить, откуда стрелял убийца. Если мы получим заключения экспертов по баллистике о траектории пули, выясним, с каких точек технически можно было выстрелить и какие у преступника были шансы оказаться замеченным, и сделаем поправку на ветреную погоду, то можем получить вероятность того, что выстрел совершался из определенной точки. Для этого нам нужно создать модель, заложив в нее все релевантные знания по интересующей нас теме, и сообщить ей данные о конкретной ситуации.

Вероятностные модели имеют богатую предысторию в науке, где они использовались как для обработки экспериментальных данных, так и для описания самых разных объектов и процессов на более высоких уровнях абстракции. В этой связи в искусственный интеллект они также пришли со стороны как субсимвольных, так и символьных методов.

> **В частности, вероятностные модели всегда играли ключевую роль в областях распознавания образов, машинного обучения и восприятия, в задачах управления.**

В символьном же ИИ они поначалу использовались эпизодически, и их фундаментальная значимость стала понятна несколько позднее.

Вероятностные модели для субсимвольных данных развивались от наивных байесовских классификаторов[54] через параметрические[55] модели, строящиеся на основе стандартных распределений, и графовые представления[56] до непараметрических байесовских методов на основе случайных процессов и нейросетевых представлений распределений вероятностей.

Основной фокус эволюции вероятностных моделей в рамках символьного подхода заключался в переходе от систем, основанных на четких детерминистических знаниях, формальной математической логике и Булевой алгебре[57], к использованию нечеткой логики[58] и вероятностному описанию знаний и правил на основе распределения вероятностей, в том числе в условиях неопределенности, ограничения ресурсов и стохастического представления моделируемых объектов и процессов. Сами вероятностные модели при этом также

---

[54] Наивный байесовский классификатор — это семейство алгоритмов классификации, которые принимают одно допущение: каждый параметр классифицируемых данных рассматривается независимо от других параметров класса. Два параметра называются независимыми, когда значение одного параметра не оказывает влияния на второй.

[55] Параметрические и непараметрические методы — это методы математической статистики. В параметрических методах генеральное распределение известно с точностью до конечного числа параметров, а непараметрические методы не предполагают знания функционального вида генеральных распределений.

[56] Граф — это абстрактное представление множества объектов и связей между ними. Некоторые данные удобнее представлять в виде графов — например, связи между пользователями соцсетей.

[57] Раздел математики, изучающий высказывания, рассматриваемые со стороны их логических значений (истинности или ложности) и логических операций над ними. Алгебра логики позволяет закодировать любые утверждения, истинность или ложность которых нужно доказать, а затем манипулировать ими подобно обычным числам в математике.

[58] Раздел математики, обеспечивающий основы для приблизительного рассуждения с использованием неточных решений и позволяющий использовать лингвистические переменные, которые бывают многозначными.

развивались от наивных байесовских до графических и далее в сторону вероятностного программирования.

> **Вероятностное программирование обладает максимальной общностью, и любая задача обучения или вывода может быть сформулирована в его терминах.**

Хотя вероятностное программирование развивалось больше в рамках символьных методов, оно, как и графические модели, может быть унифицированным образом применено как для символьного, так и субсимвольного доменов, а значит, является перспективным кандидатом для их объединения.

Важнейшими свойствами для искусственного интеллекта являются возможность решения задач предсказания будущих событий, принятия решений, постановки диагнозов и индуктивного (то есть от частного к общему) вывода знаний из данных. С развитием искусственного интеллекта все больше растет понимание, что эти свойства должны опираться не на четкий логический, а на вероятностный вывод. Однако это понимание столкнулось с принципиальными проблемами, которые отчасти и породили разнообразие подходов. Первая — проблема синтеза логики и вероятности — состоит в том, что логический вывод и вероятности выводимых высказываний плохо связаны между собой. Дело в том, что использование правил логического вывода предполагает абсолютную достоверность используемых в выводе знаний и отвечает требованиям сохранения истинности, а не вероятности. Для знаний, полученных индуктивным путем, это не работает, поскольку, если мы выводим общее правило из частных случаев, у нас всегда есть шанс встретить исключение.

Вторая проблема — проблема статистической двусмысленности — заключается в том, что в процессе обучения (индуктивного вывода) мы можем получать вероятностные правила, из которых выводится противоречие. Пример: если человек

физик, он обычно не художник, а если его рисунки демонстрировались на выставке, то он художник. Если мы применим эти правила к известному физику Ричарду Фейнману, мы получим противоречие, поскольку Фейнман — физик, но при этом его рисунки выставлялись в галерее.

Вероятностное программирование как путь к AGI

Можно поделить вероятностное программирование на две группы подходов — на основе функциональных и логических языков, которые, хоть и родственны, достаточно сильно различаются в деталях.

Функциональное вероятностное программирование позволяет задавать произвольные вероятностные модели из Тьюринг-полного[59] пространства в форме программ со случайными переменными, для которых предоставляется богатый выбор стандартных распределений вероятностей. Примерами могут служить такие языки, как Church, WebPPL, Venture, Anglican, Gen и т.д. Языки этого класса поддерживают условный вероятностный вывод[60], позволяя тем самым решать произвольные задачи как машинного обучения, так и вероятностных рассуждений на основе знаний. На них можно естественным образом выражать многие существующие модели машинного обучения и представления знаний.

На этих языках могут выражаться и модели универсального интеллекта, что делает их хорошей основой для эмпирических исследований проблем универсальной индукции. К сожалению, эти проблемы остаются нерешенными в рамках функционального вероятностного программирования из-за слабости используемых механизмов вывода.

---

[59] В теории вычислимости исполнитель называется Тьюринг-полным, если на нем можно реализовать любую вычислимую функцию

[60] Условная вероятность — вероятность наступления события A при условии, что событие B произошло.

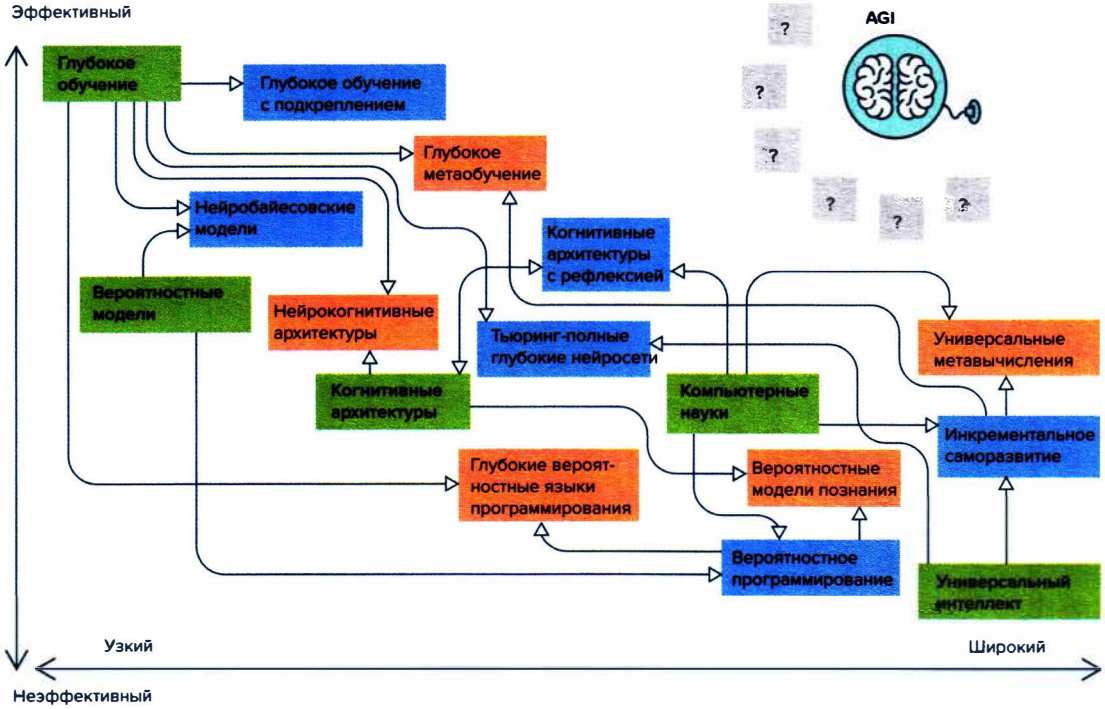


Рис. 17

Приближаясь к искусственному интеллекту

С попыткой решения проблемы синтеза логики и вероятности, статистической двусмысленности и необходимости совместного использования нечетких и экспертных знаний связано создание целой серии различных подходов. Одними из наиболее простых являются байесовские сети, для которых можно установить последовательность причинных связей, что дает этим сетям простую логическую интерпретацию.

Одним из наиболее общих подходов, реализующим синтез логики и вероятности, считается вероятностное логическое программирование (Probabilistic Logic Programming). В частности, язык ProbLog расширяет Prolog значениями вероятностей для фактов и правил и позволяет распространять эти вероятности на заключения, полученные в ходе логического вывода. ProbLog — это система моделирования задач, которые представляются в виде группы правил с вероятностным исходом.

Этим вероятностное логическое программирование заметно отличается от функционального, позволяя вероятностно рассуждать над общими понятиями, в частности, используя кванторы[61]. Однако это не дает полного решения проблем синтеза логики и вероятностей и статистической двусмысленности.

Достоинства вероятностного программирования

> **К достоинствам данного подхода прежде всего относится возможность верификации самих моделей человеком либо машиной.**

Кроме того, каждое отдельное решение, полученное вероятностной системой, можно объяснить в терминах онтологии

---

[61] Общее название для логических операций, ограничивающих область истинности какого-либо предиката (утверждения о субъекте). Например, «все X обладают свойством Y» или «существуют X, которые обладают свойством Y».

или генеративной модели, лежащей в основе соответствующей семантики. Последнее может оказаться решающим фактором с точки зрения решения задачи «объяснимого ИИ» («explainable AI», или XAI) — в свете современного права в области вычислительных систем и больших данных (законодательные акты вроде GDPR в Евросоюзе[62]), а также с точки зрения построения систем обеспечения критической инфраструктуры или обеспечения безопасности, где возможность верификации и аудита принимаемых решений является жестким условием. Необходимость объяснений особенно важна для предсказаний, диагностики и принятия решений, которые являются ключевыми задачами искусственного интеллекта.

Другим важным достоинством систем, основанных на вероятностном моделировании, является возможность их загрузки экспертными данными, касающимися таких предметных областей и задач, где машинное обучение на основе нейросетевых методов невозможно либо экономически неприемлемо, — например, там, где нужна прозрачность логики решения (скажем, при принятии решений о выдаче банковских кредитов). В этом случае декларативное либо процедурное знание, оформленное в виде семантических моделей, может быть загружено в систему исполнения вероятностных моделей для дальнейшего использования либо дообучения. Примечательно, что сочетание данной возможности с принципами «объяснимого ИИ» позволяет передавать модели не только между системами ИИ и людьми, но и между самими различными системами.

Объяснимый ИИ, конечно, является прерогативой не вероятностного, а символьного подхода, но вероятностный подход

---

[62] Действующее на территории ЕС общее положение о защите данных, в котором закреплено, что любой гражданин имеет право запросить объяснение, почему алгоритм принял относительно него то или иное решение. Это ограничивает возможность применения blackbox-решений.

позволяет перенести его также на субсимвольный уровень, одновременно предоставляя единый фреймворк для нейросимвольной интеграции.

Недостатки вероятностного программирования

К недостаткам вероятностного программирования на сегодняшний день можно отнести относительно низкую производительность — с точки зрения как скорости, так и качества решения задач, когда существующие системы вероятностного моделирования в целом пока проигрывают нейросетевым в силу более простой вычислительной реализации последних на существующих аппаратных средствах (если, конечно, говорить про задачи, в которых нейросети применимы).

Еще более серьезной проблема производительности оказывается для задачи обучения вероятностных моделей. Обучение функциональных генеративных моделей требует выхода за рамки понятийного аппарата теории вероятностей и обращения к методам метаэвристического поиска или теории метавычислений. Для логического вероятностного программирования тоже требуются весьма изощренные методы. В то же время различные методы тренировки «нейронных сетей» на основе обратного распространения сейчас хорошо проработаны на практике.

Кроме того, ни проблема статистической двусмысленности, ни проблема синтеза логики и вероятности в настоящее время не решены в полной мере, и их невозможно решить в рамках традиционного подхода к вероятностному логическому программированию. Сам логический вывод должен быть заменен на вероятностный, как это сделано в функциональном вероятностном программировании (и ранее — в байесовских сетях), но с поддержанием вывода над декларативными знаниями, как в логическом вероятностном программировании. Для решения этих проблем требуется принципиально другой математический аппарат. Существуют работы российских

исследователей[63], где определяются максимально специфические правила, для которых можно доказать, что индуктивно-статистический вывод по ним непротиворечив. Это означает, что из таких правил мы не получим противоречивых предсказаний или решений индуктивно статистическим выводом. Эти правила можно найти с помощью специального семантического вероятностного вывода.

Текущие проекты

Вот некоторые из наиболее полных и известных реализаций подхода на основе вероятностных моделей, имеющих отношение к AGI.

- Фактор-графы в когнитивной архитектуре Sigma (разрабатываемой под руководством П. Розенблюма).
- Уже упомянутая система неаксиоматического вывода NARS (Non-axiomatic Reasoning System), позволяющая решать задачи логического вывода в условиях ограниченных ресурсов таким образом, что качество решения задачи, выраженное мерой «уверенности» в результате, будет зависеть от выделенных ресурсов.
- Сети вероятностной логики (Probabilistic Logic Networks — PLN), первоначально предложенные Б. Герцелем и имплементированные в рамках проекта OpenCog.
- Язык функционального вероятностного программирования Gen, который служит для построения вероятностных моделей познания, что помещает данный проект непосредственно в контекст AGI. Отличительной особенностью Gen является введение в него метапрограммирования вывода[64].

---

[63] Vityaev, E., Odintsov, S. How to predict consistently? // Trends in Mathematics and Computational Intelligence In: Studies in Computational Intelligence, 796.

[64] Метапрограммирование — вид программирования, связанный с созданием программ, которые порождают другие программы как результат своей работы.

- Семантическое моделирование с вероятностным выводом, основанное на задачном подходе и реализованное в программной системе Discovery, которое решает следующие задачи:
  1) обнаружение знаний в виде максимально специфических правил при использовании семантического вероятностного вывода;
  2) непротиворечивое предсказание и принятие решений при использовании максимально специфических правил;
  3) моделирование целенаправленного поведения аниматов в соответствии с теорией функциональных систем[65] физиолога Петра Анохина.

Компьютерные науки

Введение

Исследования в области искусственного интеллекта в целом опираются на математику, но для общего ИИ особое значение приобретают направления, связанные с областью компьютерных наук, включая математическую логику, теорию алгоритмов и сложность вычислений, теорию информации, верификацию программ и многое другое. А вопрос о возможности создания мыслящих машин тут же заставляет рассмотреть соотношение между мышлением и вычислимостью.

---

[65] Функциональная система — это система разнородных физиологических составляющих, все части которой содействуют получению определенного полезного результата, что предполагает наличие определенной модели будущего и проактивное поведение.

С точки зрения AGI возникает вопрос, как на имеющейся аппаратной базе создавать системы, способные эффективно решать широкий круг задач. Естественно, для ответа на этот вопрос оказываются необходимыми не только знания о том, какие задачи являются в принципе разрешимыми, но и сведения о более тонких результатах из области математики и компьютерных наук.

Концепции вычислимости в искусственном интеллекте

При решении человеком задач с помощью компьютера стандартная последовательность действий выглядит следующим образом: формулируется задача, ищется алгоритм ее решения, формализованное описание алгоритма передается программисту для написания кода, пишется и отлаживается программа, планируются и осуществляются вычисления, полученные данные интерпретируются заказчиком, и если они удовлетворительны с точки зрения некоторого критерия, то принимается решение, что задача решена. Если же результат неудовлетворителен, то производится анализ предпринятых действий и в них вносятся соответствующие изменения. Заметим, что такие изменения могут коснуться практически всех этапов процесса решения задачи, вплоть до этапа ее постановки.

При этом ошибки, нечеткости или неточности, допущенные при формулировке задачи, могут повлечь за собой весьма дорогостоящие последствия, что налагает на постановщика задачи особую ответственность.

> **Возникает вопрос: а существует ли ситуация, когда правильно поставленная задача фактически означает ее полноценное решение?**

Поскольку речь идет о решении человеком задач с помощью компьютеров, то этот вопрос можно поставить иначе:

существует ли возможность формулировать задачу таким образом, чтобы из этой формулировки компьютер мог автоматически извлечь требуемое решение? И не только извлечь, но и объяснить? Поиском удовлетворительных ответов на эти вопросы и занимается область ИИ.

В определении общего искусственного интеллекта речь идет об искусственных системах, способных по спецификации задачи самостоятельно, то есть автоматически, синтезировать алгоритм ее решения. Сразу же отметим, что проблема автоматизации процесса решения интеллектуальных задач естественно распадается на две подпроблемы: автоматизацию вычислений и автоматизацию рассуждений. И искать ответ на поставленные выше вопросы можно, решая отдельно как проблему автоматизации вычислений, так и проблему автоматизации рассуждений. В частности, большой пласт работ в области искусственного интеллекта посвящен исследованию формальных языков и алгоритмической сложности проблем, связанных с вычислениями либо рассуждениями на основе входных спецификаций задач, представленных в таких языках. Далее мы будем обсуждать исходную проблему автоматизации решения задач, делая акцент на вопросе автоматизации вычислений, а точнее, мы будем осуществлять поиск ответов на поставленные выше вопросы путем поиска и анализа существующих моделей, формализующих понятие «вычислимость». Вычислимые функции — это функции, которые могут быть реализованы на машине Тьюринга.

Фундаментальным изучением понятия вычислимости занимается математическая логика (то есть наука о принятых в математике способах рассуждений) в рамках трех основных подходов: императивного, аксиоматического (лежащего в основе декларативного программирования) и теоретико-модельного. Ниже мы подробнее разберем первые два, поскольку именно они в основном используются в искусственном интеллекте.

Императивный подход

Императивный подход[66], нашедший свое практическое воплощение в традиционных технологиях и языках программирования, имеет в своем арсенале большое число моделей вычислений: машина Тьюринга, машина Поста[67], алгорифмы Маркова, лямбда-исчисление Черча[68], сети Петри[69] и другие, в которых вычислимость предстает как некий процесс управления состоянием памяти некоего абстрактного вычислителя. При описании задач и при их решении в виде алгоритмических процессов в императивном подходе мы всегда имеем дело с двумя видами сущностей:

- **данными, то есть с тем, что мы хотим обрабатывать, преобразовывать, чем управлять;**
- **операциями, то есть с действиями над данными.**

Очевидно, что любой язык формулирования задач и любой язык, в терминах которого мы записываем ее решение, должен предоставлять возможность описывать эти сущности. Для этого такие языки обычно содержат, с одной стороны, элементарные выражения, представляющие исходные примитивные данные и операции над ними (базис языка), а с другой — набор специальных средств конструирования из этих простых объектов более сложных конструкций, которые могли бы

---

[66] Императивная программа предписывает, как конкретно должна решаться задача, — в отличие от декларативной, где прописан только желаемый результат, но не способ его достижения. Примеры императивных языков программирования — C, Java и Python.

[67] Машина Поста — абстрактная вычислительная машина, предложенная математиком Эмилем Леоном Постом.

[68] Система, разработанная американским математиком Алонзо Черчем для формализации и анализа понятия вычислимости.

[69] Сети Петри — математический аппарат для моделирования динамических систем, в которых время изменяется дискретно. Впервые описаны Карлом Петри в 1962 г.

рассматриваться как единые сущности. Это позволяет манипулировать данными конструкциями с помощью средств того же языка. Но при императивном программировании мы вынуждены разделять и отдельно хранить данные и операции.

Аксиоматический подход

Помимо императивных языков программирования, искусственный интеллект активно пользуется декларативными языками программирования — функциональными и логическими. В то время как императивный язык описывает конкретное действие с известными входными параметрами, декларативный описывает законы взаимодействий (аксиомы), не вдаваясь в частности. Предполагается, что компьютер сам способен логически вывести из аксиом верное решение. Примеры декларативных языков — LISP, Haskell, Scala, R.

> **Когда мы описываем задачу, которую будет решать компьютер, формулировка должна быть свободной от неточностей и двусмысленностей.**

Так что тут имеет смысл применять по возможности строгие, формальные конструкции. При этом человеческий опыт подсказывает, что нам удобнее задавать правила, по которым надо действовать, а не писать пошаговую инструкцию для компьютера, составлять которую довольно мучительно. Допустим, вам надо поставить рабочие задачи перед менеджером супермаркета. Проще сформулировать их в виде конструкций «если… то»: если образовалась очередь, то нужно открыть другую кассу, если кто-то разбил банку с консервами — позвать уборщицу, и т.д. А теперь представьте, что вам надо пошагово расписать весь его рабочий день. Не самая простая задача. Но в отдельных случаях это необходимо — как в жизни (например, если вы даете задачу проверить готовность самолета к рейсу, то нужны четкие чек-листы), так и в программировании. Кроме того,

декларативный подход предполагает, что наш менеджер супермаркета (или программа) знает, что такое очередь, как открывать другую кассу и т.д. А эта информация уже закладывается через императивное программирование. С другой стороны, императивные языки программирования высокого уровня, как правило, имеют достаточно развитые средства написания повествовательных, декларативных конструкций. Используя такие средства, можно в отдельных случаях, когда уже заранее имеется некая компьютерная программа решения целого класса задач (шаблон), формулировать конкретную задачу из данного класса не как последовательность инструкций, а как совокупность декларативных условий.

> **Декларативный способ решения задачи выглядит намного привлекательнее, поскольку вместо указаний о том, как решать задачу, мы формулируем лишь описание того, *что* следует решать.**

Поэтому вполне закономерным является вопрос о том, можно ли создать декларативный язык постановки задач, который соответствовал бы следующим критериям:

- опирался бы на общепризнанную и развитую математическую базу;
- был бы максимально приближен к постановщику задачи, удобен и комфортабелен для формулировок задач и способен адекватно отражать в постановке исходную семантику используемых знаний;
- был бы пригоден для формулировки по возможности максимально широкого класса практических задач;
- допускал бы эффективную интерпретацию своих конструкций как набора команд для компьютера.

Заметим, что в предельном случае этот вопрос звучит как вопрос о возможности создания универсального декларативного

языка с добротной математической базой, пригодного для декларативного описания любой (!) задачи, имеющей алгоритмическое решение, то есть язык должен быть Тьюринг-полным.

Функциональное программирование

Основной вклад в развитие парадигмы функциональных языков внес язык LISP, который на протяжении многих лет оставался наиболее популярным функциональным языком программирования. Этот язык использовался в большом числе исследований и разработок в области ИИ при решении таких задач, как обработка естественного языка, создание экспертных систем, распознавание образов и многие другие. Но, как мы уже говорили, декларативные языки на самом деле не являются чисто декларативными, поскольку в них разрешается применять императивные конструкции. Это приносит определенную практическую пользу, но приводит к излишней сложности используемых конструкций. То же самое происходит и при добавлении в императивные языки функциональных средств. Попытки создания языка, который совмещал бы в себе сильные стороны функционального и императивного программирования, продолжаются до сих пор.

> **В функциональном программировании действия представляются в виде функций, которые одни данные (аргументы) преобразуют в другие данные (значения).**

При этом, в отличие от императивных языков, значения функций однозначно определяются их аргументами и не зависят как от истории вычислительного процесса, так и от внешних по отношению к вычислительному процессу сигналов и состояний. В декларативном языке значение функции может зависеть от разных внешних параметров.

Общая схема формулирования и решения задач в функциональном программировании сводится к следующему:

- функциональная спецификация задачи, то есть написание ее исходных условий в виде конечного набора определений функций;
- написание запроса и задание его параметров;
- вычисление значений данного запроса.

То есть нам задается задача и ее условия в виде набора определений функций. Эти определения компьютер воспринимает как аксиомы. Ему надо свести воедино все правила и логические связи между ними и вычислить оптимальное решение задачи.

Логическое программирование

Похоже, но несколько иначе, ставятся и решаются задачи в другой парадигме декларативного программирования — в логическом программировании (ЛП). Логическая программа представляет собой формульное описание вычислимого отношения между параметрами в терминах формального логического языка. Если функциональное программирование в качестве «строительных блоков» использует функции, то логическое программирование — предикаты. Хотя предикат можно трактовать как логическую функцию, но разница между параметрами, объявленными входными, и параметрами, рассматриваемыми как выходные, в логической программе достаточно условна, что и отличает парадигму логического программирования от функционального. Дело в том, что вычисление функции всегда является строго направленным процессом: мы подаем на вход значения аргументов функции и на выходе должны получить результат, то есть вычисленные значения функции. В логических же языках допускаются ситуации, когда вначале указывается значение выходного параметра и ищется такое значение входного, при котором имеет место заданное отношение. Другое важное отличие логического программирования от функционального — это неодно-

значность логических вычислений, поскольку удовлетворяющих исходному запросу значений переменных у логической программы может оказаться несколько.

> **В логическом программировании исходные условия задачи задаются конечным набором логических правил и фактов, записанных в виде логических формул.**

Запрос также представляет собой логическую формулу этого языка, которая может содержать переменные, а цель решения задачи — попытка либо доказать, что запрос является логическим следствием аксиом, либо убедиться в обратном.

Первые попытки реализовать отдельные идеи логического программирования делались еще в начале 1960-х гг. Значительный вклад в его становление внес британский философ и логик Джон Алан Робинсон, создавший метод резолюций — ключевой метод в математической логике для проверки того, являются ли некоторые утверждения логическим следствием других утверждений, посылок или гипотез. Используя этот метод, в 1971 г. французские исследователи Аллен Колмероэ и Филлип Руссель создали первый язык логического программирования — «Пролог» (Prolog — PROgrammation en LOGique). Пик популярности идей логического программирования пришелся на 80-е гг. прошлого столетия.

Уже с момента появления логического и функционального языков программирования неоднократно предпринимались попытки объединить эти две декларативные парадигмы. Как правило, главной целью всех предлагаемых вариантов объединения была возможность использования более широкого набора стратегий управления вычислениями. Однако возникли сложности с единообразием используемых структур данных и их типизацией.

Кроме того, у языков декларативного программирования есть общий недостаток. С одной стороны, чем меньше

ограничений накладывается на допустимые способы построения аксиоматических теорий (наборов правил), тем проще пользователю специфицировать задачу. Но, к сожалению, когда мы даем компьютеру свободу интерпретировать теорию, больше шансов, что он сделает это неэффективно и получит неоптимальное решение. По крайней мере, если поиск решения реализуется как поиск доказательства в формальной системе. Это противоречие заставляет сторонников и апологетов декларативного программирования искать разумный компромисс между удобством и выразительностью существующих декларативных языков и их эффективностью. Такой поиск может осуществляться либо в рамках существующих парадигм функционального и логического программирования, либо в их критическом осмыслении и, возможно, создании новой парадигмы.

Формальные методы в ИИ

Формальные методы, представляют собой группу техник для спецификации, разработки и верификации программного и аппаратного обеспечения. Они базируются на формальных спецификациях программ. Спецификация программы — точная и полная формулировка задачи, содержащая информацию, необходимую для построения алгоритма решения. Формальная спецификация — это спецификация, записанная в виде набора строгих математических формул на одном из формальных языков.

Формальные методы реализуются в разнообразных инструментах анализа и верификации — от тестирования и статического анализа программ до дедуктивной верификации и программного синтеза. Дедуктивная верификация — это проверка соответствия программы определенным критериям (условиям верификации), реализуемая автоматическими или интерактивными инструментами доказательства теорем. Она дает абсолютную гарантию корректности программы относительно

спецификации, однако сложна, требует много времени и высокой квалификации, поэтому обычно применяется для небольших критических фрагментов программ с высокой ценой ошибки. Наиболее сложным и сильным методом является программный синтез, позволяющий автоматически построить программу по ее спецификации.

Как уже было сказано выше, инструменты доказательства теорем подразделяются на автоматические и интерактивные. SAT[70]- и SMT[71]-решатели автоматически ищут доказательство теорем. Популярных SMT-решателей около 30. Их мощь с каждым годом возрастает. Это решатели Alt-Ergo, CVC3, CVC4, Z3 и др. Системы интерактивного доказательства — Coq, PVS, HOL и т.п. — предоставляют пользователю удобный интерфейс для доказательства теорем.

Формальные методы определяют базис популярной ныне модельно-ориентированной (model-based, model-driven) технологии в программной инженерии. Для создаваемой программы конструируется и верифицируется модель. В традиционной технологии разработка большой программной системы сопровождается наличием множественных нестыковок и разломов во внутренних интерфейсах системы. В модельно-ориентированной технологии внутренние интерфейсы выстраиваются строго в соответствии с моделью, что принципиально повышает надежность программной системы. Примером является разработка и верификация модели политики безопасности управления доступом в операционных системах, в том числе для Astra Linux Special Edition,

---

[70] SAT (satisability) — задача выполнимости булевых формул: у нас есть булева формула, и мы хотим узнать, есть ли такие значения переменных, при которых она истинна.

[71] SMT (satisfiability modulo theories) — задача выполнимости формул в теориях. В отличие от задачи выполнимости булевых формул, SMT-формула содержит вместо булевых переменных произвольные переменные, а предикаты — это булевы функции от этих переменных.

ориентированной на применение в военных и правительственных организациях.

> **В настоящее время при массовой эксплуатации систем искусственного интеллекта периодически стали возникать аварийные ситуации, вызванные ошибками в программировании. Повысились требования к надежности и безопасности систем ИИ, отраженные, в частности, в изменившихся правилах их сертификации.**

Это лишь одна из причин возрастания интереса к применению формальных методов в сфере ИИ.

В последние годы все чаще появляются серьезные работы по применению формальных методов в задачах искусственного интеллекта. Одновременно на конференциях по формальным методам увеличилось число работ, ориентированных на искусственный интеллект. Осознание этого привело к появлению новых международных конференций (воркшопов) по формальным методам в контексте ИИ:

- Artificial Intelligence and Theorem Proving (AITP) 2016–2019;
- Formal Methods and AI (FMAI) 2018–2019;
- OVERLAY2019: First workshop on Artificial Intelligence and fOrmal VERification, Logic, Automata, and sYnthesis.

Новые конференции пока набирают силу. Их тематика еще не сложилась и довольно пестра. Рассматриваются самые разные вопросы на стыке ИИ и формальных методов: методы верификации систем искусственного интеллекта, алгоритмы и методы доказательства свойств систем ИИ, безопасность и защищенность от вредоносного воздействия систем ИИ, вероятностные модели, автоматическое планирование во времени и другие вопросы.

Написание формальной спецификации проверяемых свойств системы искусственного интеллекта часто оказывается крайне сложным.

> **Спецификация систем машинного обучения вызывает особые трудности, обусловленные, в частности, применением вероятностной логики.**

Здесь необходима разработка новых методов абстракции от механизма обучения (в сочетании с механизмом объяснений).

Доказательная оценка качества построения программных систем, основанная на символьных методах, позволяет оценивать устойчивость нейросетевых программ к возмущениям в данных и генерировать примеры уязвимости, когда небольшие возмущения приводят к ложному результату.

Построение формальных спецификаций моделей и их верификация доступны только специалистам по формальным методам, которых очень мало в нашей стране. Эта специальность требует высоких способностей и многоэтапного длительного обучения. Если мы хотим сделать системы ИИ надежнее и безопаснее, потребуется обеспечить подготовку таких специалистов.

Роль компьютерных наук в AGI

Можно ожидать, что по мере более глубокого понимания ключевых свойств и границ применимости имеющихся подходов узкого искусственного интеллекта появится спектр моделей ИИ с математическими основаниями, в рамках которых будут развиваться основные направления AGI. Одним из важных вопросов с точки зрения AGI является вопрос выразительности тех или иных моделей машинного обучения с точки зрения реализуемых ими вычислений. Например, в глубоком обучении рассматривается вопрос о Тьюринг-полноте тех или иных моделей

нейронных сетей. При этом ряд заблуждений в этой области связан с нестрогостью понятия «нейронная сеть». Например, в работе известной исследовательницы Хавы Сигелман показана Тьюринг-полнота одной из «нейросетевых» моделей, имеющей, однако, весьма условное отношение к тем искусственным нейронным сетям, которые используются в машинном обучении. Тем не менее этот результат пытаются тиражировать на все рекуррентные нейросетевые модели, упуская из вида тот факт, что такие нейронные сети соответствуют другим моделям вычислений, не являющихся Тьюринг-полными. Есть попытки сформулировать более тонкий результат, что любую машину Тьюринга с ограниченной памятью и временем вычислений можно проэмулировать в рекуррентной нейронной сети. Однако при этом не учитывается размер получаемой сети, который может быть экспоненциальным от размера описания входной машины Тьюринга, причем должны меняться и сами веса связей (то есть фактически это будут разные сети), тогда как в машине Тьюринга будет изменяться только размер ленты.

> **Непосредственно востребованными результаты из области компьютерных наук оказываются в рамках такого подхода к AGI, как универсальный алгоритмический интеллект.**

Действительно, базовая модель универсальной индукции имеет прямое отношение к алгоритмической теории информации. Также и в рамках модели AIXI «широкий диапазон сред» задается через понятие алгоритма — как класс всех вычислимых сред. При этом возникают вопросы о невычислимости самих моделей универсального интеллекта. Такие упоминавшиеся в разделе про универсальный интеллект модели, как AIXItl и машина Геделя, интенсивно опираются на теорию дедуктивного вывода. При этом при попытке практической реализации машины Геделя возникает необходимость рассмотреть вопросы, относящиеся сугубо к области компьютерных

наук. В области универсального алгоритмического интеллекта эпизодически используются самые разные идеи из компьютерных наук, например ленивые вычисления[72].

Дизайн когнитивных архитектур также тесно связан с дизайном языков программирования. Почти каждая когнитивная архитектура включает свой язык (например, Atomese для OpenCog, Narsese для OpenNARS и т.д.). Вполне очевидна связь компьютерных наук с дизайном языков вероятностного программирования, однако эта связь может быть весьма обширной. В частности, одной из последних тенденций в функциональном вероятностном программировании является уже упоминавшееся метапрограммирование вывода, а несколько раньше рассматривались методы анализа программ в целях повышения эффективности вывода.

В целом можно сказать, что на настоящий момент подхода к AGI на основе компьютерных наук пока не существует, так как нет систематических попыток использовать результаты из этой области к проблеме создания AGI. Однако представляется возможным, что подход к AGI со стороны именно математики и компьютерных наук является наиболее естественным, поскольку возникновение самих компьютеров как машин, способных эмулировать любой автомат, обязано развитию теории алгоритмов, а создание систем AGI, способных находить алгоритм решения любой разрешимой задачи, может быть обязано компьютерным наукам, систематическое применение результатов изысканий которых к AGI представляется актуальным.

---

[72] Применяемая в некоторых языках программирования стратегия вычисления, согласно которой вычисления следует откладывать до тех пор, пока не понадобится их результат

ГЛАВА 4. ВАРИАНТЫ ВОПЛОЩЕНИЯ

Введение

Выше мы говорили о том, как общий интеллект может быть устроен изнутри и какие принципы могут быть положены в его основу. Но как именно он будет создан и какую форму приобретет? В научной фантастике, особенно более старой, вариантом, встречающимся чаще всего, является робот. Действительно, говоря о разуме — искусственном или нет, — мы представляем если не личность, то некий целостный агент, субъект, отделяемый от других субъектов и наделенный своим телом. Робототехника активно развивается и вполне может воплотить эти фантазии в реальность.

В то же время такое направление в ИИ, как обучение с подкреплением, всецело концентрируется на обучении интеллектуальных агентов, меньше внимания уделяя тому, как могут быть устроены их тела, но также предполагая, что агент

помещен в некую среду, где он может совершать действия и получать сенсорную информацию.

Однако ИИ порой рисуется не как чудо инженерной и научной мысли, спроектированное и воплощенное от начала и до конца кем-то конкретным в виде интеллектуального агента или робота, а как нечто, что возникает само по себе. Конечно, было бы крайне наивно полагать, что ИИ может возникнуть сам на пустом месте. Но если создать подходящие условия, это вполне может произойти, ведь и человеческий разум возник эволюционно, а не в результате проектирования. Неудивительно, что и в рамках ИИ существует целое направление эволюционных алгоритмов, подражающих биологической эволюции.

Возникшая же в XX в. наука о самоорганизации — синергетика — пролила свет на процессы возникновения сложных систем не только в ходе эволюции с ее конкретными генетическими механизмами, но также в физике, химии и даже экономике. Если процессы самоорганизации вездесущи, то они могут сыграть роль и в возникновении общего ИИ. Может ли общий ИИ воплотиться в форме сети взаимодействующих друг с другом алгоритмов ограниченной сложности или в форме коллективного разума агентов с ограниченным уровнем интеллекта, взаимодействующих через интернет? Экосистемный подход к общему ИИ предполагает положительный ответ на этот вопрос и фокусируется на создании условий, способствующих самоорганизации как при создании компонентов ИИ, так и в процессе их работы при взаимодействии друг с другом.

Так каким же образом воплотится общий ИИ — в теле конкретного робота или как глобальный интернет-разум? Или, может, произойдет что-то еще, например оцифровка человеческого сознания? Попробуем разобраться с этими вопросами детальнее.

Обучение с подкреплением как путь к общему интеллекту

Как это работает

У животных (не исключая хомо сапиенсов) распространено обучение через ожидание вознаграждения или наказания, которые ассоциируются с определенными действиями. Мы запоминаем, какое поведение приводит к тому, что мы получаем пищу, секс или еще что-то приятное и полезное, а после какого нас ждут страдания. Если нам удастся воплотить такие механизмы обучения в ИИ в сочетании с механизмами мультимодального сенсорного восприятия (зрения, слуха, осязания и т.д.), это может привести к прорыву в построении систем, демонстрирующих мышление на человеческом уровне. В этой связи обучение с подкреплением (*reinforcement learning, RL*) может стать одним из главных подходов на пути к общему искусственному интеллекту (рис. 18).

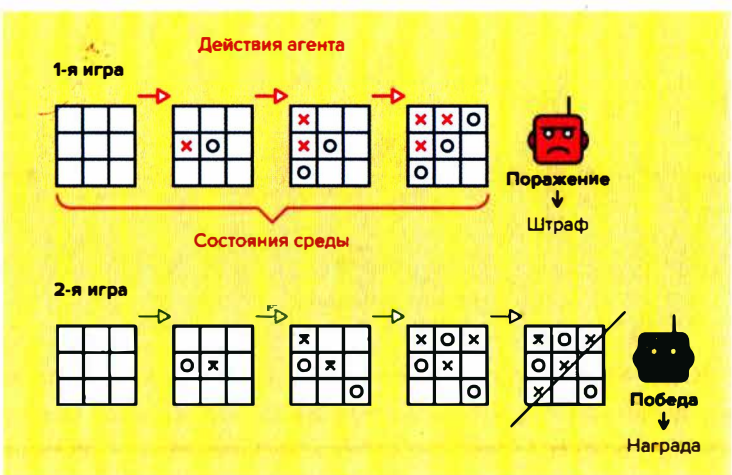


Рис. 18

Обучение с подкреплением

Как мы уже говорили, цель обучения с подкреплением — обучить агента, взаимодействующего с неизвестной средой, такой стратегии поведения, которая максимизирует суммарное вознаграждение от среды. Например, на заре создания компьютерных игр существовала аркадная игра Breakout: игрок управлял дощечкой в нижнем поле; этой дощечкой нужно было отбивать мячик так, чтобы он, с одной стороны, не падал, а с другой — разбивал кирпичики в верхней части поля (рис. 19).

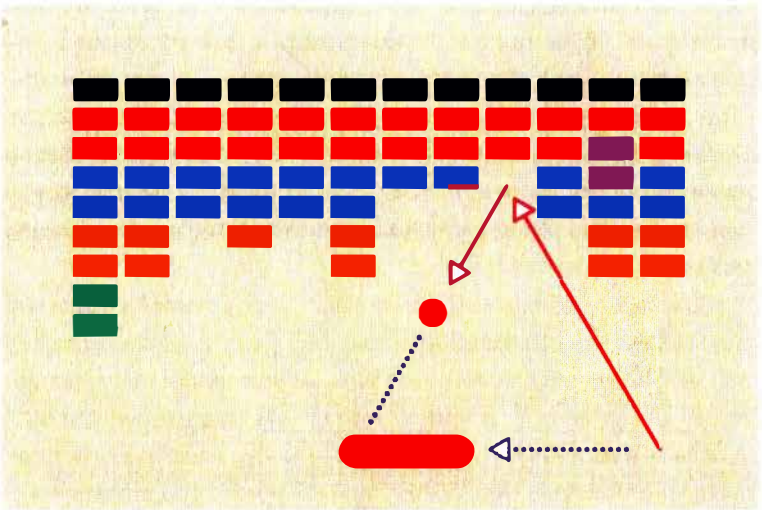


Рис. 19

Игра в Breakout

Когда в эту игру играет человек, он постепенно учится двигать дощечкой так, чтобы не терять мячик и получать больше очков за разбитые кирпичики. Но принцип выбора удачной стратегии кажется интуитивным: мы не осознаем, как конкретно двигаем дощечку и какую обратную связь получаем на каждое скорректированное движение. Как бы это выглядело

для компьютера? Он принимает какое-то решение о положении дощечки, дощечка отбивает мячик, мячик разбивает кирпичики, агент получает награду (баллы). Его задача — найти стратегию поведения, которая приносила бы максимум баллов.

Тут нет размеченной внешним экспертом обучающей выборки, и агент может самостоятельно исследовать среду методом проб и ошибок.

> **В идеале агент не должен запоминать каждое действие (потому что предусмотреть все возможные ситуации все равно не получится — их может быть слишком много), а должен пытаться обобщить ситуации, чтобы выходить из них с максимальной выгодой.**

Сейчас обучение с подкреплением используется для автоматической торговли, промышленного моделирования, управления ресурсами предприятий и создания беспилотных автомобилей.

Одна из проблем метода — в том, что агенту достаточно сложно найти оптимальный баланс между исследованием и использованием: чтобы получить максимальное вознаграждение, агент должен и использовать уже найденные удачные действия в известных ситуациях, и пробовать новые действия, пытаясь попасть в новые ситуации. Кроме того, агенту надо понять, какие именно его действия принесли вознаграждение, а среда может реагировать с временной задержкой, и это мешает обнаружению причинно-следственных связей. Так же как, например, в жизни удачно принятое решение может дать результат только через какое-то время.

В случае с Breakout нужно связать выбранное положение дощечки с тем, как потом от нее отлетает мячик и какие баллы за это начисляются. И это еще очень простой вариант. И если промежуток между важным выбором и вознаграждением слишком велик, обнаружить эту связь может быть очень

сложно. При этом среди всех форм машинного обучения обучение с подкреплением наиболее приближено к процессам обучения в живой природе. Оно способствует автономному получению навыков, что дает возможность решать задачи, для которых сложно построить формальные модели, — например, выбирать поведение на дороге с учетом не только правил дорожного движения, но и разнообразного поведения других водителей, погодных условий и этических норм.

Ранние попытки применения обучения с подкреплением были недостаточно масштабируемы и встречались лишь в задачах сравнительно небольшой размерности — например, в обучении игре в нарды. Программа TD-Gammon, созданная в исследовательском центре IBM в 1992 г., лишь немного отставала по мастерству от лучших игроков в нарды того времени. Развитие глубокого обучения позволило существенно расширить круг решаемых задач. Первым громким успехом в глубоком обучении с подкреплением (*deep reinforcement learning*) была разработка модели, способной играть в аркадные игры Atari сравнимо с человеком или лучше, используя в качестве входных данных только изображения экрана и результат игры. Другой широко известный успех последнего времени — разработка гибридной системы для игры в классические настольные игры (шахматы, сеги и го), которая обучалась, исключительно разыгрывая партии сама с собой, и превзошла в мастерстве лучших профессиональных игроков.

Модельные и безмодельные подходы

Математическое описание обучения с подкреплением строится на основе MDP — марковского процесса принятия решений (рис. 20).

> **MDP — это способ представления «динамики» среды, то есть того, как среда будет реагировать на возможные действия, которые агент может предпринять в том или ином состоянии.**

Рис. 20

Марковский процесс на примере автогонок

Состояние — это те условия, которые окружают агента после совершения того или иного шага. (Например, на рисунке 8 среда может принять состояние «Дом». Из него можно перейти в состояния «Бар» или «Работа».) Обычно цель агента — научиться оптимальным образом (например, быстрее всего) попадать из состояния X в состояние Y. Есть функция перехода, которая, учитывая текущее состояние среды и возможные действия агента, выдает вероятность перехода к любому из следующих состояний. И есть также функция вознаграждения. Она выводит вознаграждение, учитывая текущее состояние среды и действие, предпринятое агентом. Агент стремится максимизировать награду на всей траектории. При этом более близкие по времени награды обычно оцениваются выше, чем те, которые ждут в отдаленном будущем (так же, как мы, люди, часто предпочитаем съесть тортик сейчас, чем быть стройными потом).

Сочетание функций перехода и вознаграждения составляет *модель среды*. Когда модель среды известна, выбор

оптимального действия на каждом шаге можно осуществлять на основе функции перехода и функции вознаграждения известными строгими алгоритмами. Однако в обучении с подкреплением модель среды для агента, как правило, неизвестна. Что делать? Казалось бы, нужно выучить модель среды на основе опыта взаимодействия с ней. И т.н. *алгоритмы на основе моделей* именно это и делают — они пытаются в явном виде оценить функции перехода и вознаграждения, чтобы на их основе выбирать наилучшие действия. Такие алгоритмы в процессе обучения не только узнают, какие шаги приведут их к большей награде, но и почему (то есть, по каким законам работает среда).

Однако оказывается, что агент может научиться оптимальной стратегии и без обращения к модели среды. Общая идея здесь заключается в том, чтобы для каждого состояния оценить его совокупную полезность как сумму наград, которые можно получить, начиная из этого состояния. Для такой функция полезности выполняется простое соотношение: величина полезности для каждого состояния равна полезности того состояния, в котором мы окажемся после совершения действия в соответствии с нашей стратегией поведения, плюс непосредственная награда, которую мы получим при переходе из состояния в состояние. Полезность нового состояния при этом обычно берется с дисконтным множителем, а если результирующее состояние неоднозначно, то делается усреднение по возможным исходам. Если стратегия агента фиксирована, то он может взять случайно инициализированную функцию полезности и обновлять ее в соответствии с этим соотношением на каждом шаге взаимодействия со средой. Часто так оценивается полезность не только самих состояний, но и действий, которые агент может в них совершить.

Видно, что функция полезности состояний привязана к стратегии поведения. Но мы ведь можем и улучшить нашу стратегию! Если у нас есть оцененная функция полезности, то агент на каждом шаге может выбирать то действие, которое обладает

наибольшей полезностью. Тем самым он улучшит свою первоначальную стратегию. На самом деле, ему не нужно сначала оценивать функцию полезности, потом улучшать стратегию, потом снова для нее уточнять функцию полезности. Он это может делать одновременно, просто выбирая наилучшее действие в соответствии с текущей оценкой полезности состояний и действий и обновляя оценку в соответствии с получившимся результатом. Для такой схемы, правда, как раз и свойственна проблема исследования-использования, так как первое удачное действие в каждом состоянии агент будет повторять снова и снова, считая его оптимальным и просто не зная, что другие действия могли бы в будущем привести к лучшим результатам. Существуют разные подходы к решению этой проблемы. Наиболее простой (но не лучший) из них — это время от времени совершать случайные действия. Кроме того, оказывается, что можно выучивать функцию полезности для одной стратегии (например, оптимальной), следуя при этом другой стратегии (например, исследовательской).

> **Таким образом, в безмодельном подходе агент будет использовать только значения суммарной полезности состояний.**

Тогда он будет видеть, какие действия для него лучше, хотя и не будет видеть почему. Например, представьте, что вы играете в крестики-нолики — за крестики. Вы не пытаетесь просчитать, как ответит противник на тот или иной ваш ход. Вы просто на каждом ходе смотрите на текущую ситуацию на доске и делаете тот ход, который приведет к лучшей позиции. Была ли позиция лучше, выясняется в конце партии, когда игроки получают подкрепление за выигрыш или наказание за проигрыш. Получив награду или наказание, агент уточнит значение полезности соответствующей позиции. Когда он в следующий раз совершит действие, которое приведет его в позицию с уточненной полезностью, он уточнит полезность предыдущей позиции

(агент не пытается запомнить, какие действия при каких состояниях куда ведут, то есть не пытается построить функцию переходов, но по факту совершения действия у него будет предыдущее и последующее состояние, так что для предыдущего состояния он сможет уточнить значение полезности). Сыграв много партий, агент сможет оценить функцию полезности для каждой позиции для оптимальной стратегии. Он будет знать, что наилучший первый ход — в центр, хотя и не будет знать — почему. И на каждый ответ нолика он будет знать, какой следующий ход приведет его ближе к победе.

Видно, что для чуть более сложных сред, чем крестики-нолики, отдельно запоминать и уточнять полезность каждого состояния или игровой позиции нереалистично. И тут как раз ситуацию спасают методы глубокого обучения, которые заменяют табличное представление функции полезности или описание состояний среды вручную сконструированными признаками (как в TD-Gammon) ее аппроксимацией с помощью нейросети. Оказывается, что обучение такой нейросети может естественным образом сочетаться с уточнением самих значений полезности в ходе взаимодействия со средой.

Сравнительные преимущества модельных и безмодельных подходов

Выучить функцию полезности, зависящую лишь от одного состояния, оказывается заметно проще, чем модель среды, зависящую от пары состояний, да еще и требующую потом перебора для выбора оптимальных действий. Если же говорить про частично наблюдаемые среды, состояния которых не полностью даны в наблюдениях, то построение их моделей становится совсем уж сложной задачей, тогда как безмодельные методы могут сносно работать с минимальными модификациями. Этим и объясняется их более ранний успех.

Однако, несмотря на сходство с биологическими механизмами обучения и впечатляющие достижения, безмодельное

обучение с подкреплением отличается от человеческого мышления рядом особенностей. Важнейшим инструментом познания для человека является построение моделей мира и объяснение с их помощью наблюдаемых данных. Расширяя подобные высокоструктурированные модели и применяя их в различных областях, мы способны выводить причинно-следственные связи даже при малом числе наблюдений (например, в детстве мы уже за несколько эпизодов понимаем, что рисование на обоях приводит к неприятностям). Во многом эта способность определяет феноменальную скорость обучения новым навыкам, которая на данный момент остается недоступной для искусственного интеллекта.

> **Даже алгоритмам, специально разработанным с целью более быстрой адаптации к похожим задачам, все еще требуются миллионы обучающих примеров, в то время как люди достигают тех же результатов за считанные минуты практики.**

При разработке систем искусственного интеллекта исследователи редко сосредоточены на том, как именно алгоритм приходит к решению задачи. Современный искусственный интеллект не руководствуется тем же набором предпосылок, которые свойственны мышлению человека. Среди таких предпосылок — целеполагание и интенциональность, объектность мира и базовые представления о физике, элементарное понимание арифметики и геометрии. На этих концепциях строится наше понимание механизмов мышления человека и животных, и кажется разумным, что общий искусственный интеллект должен строиться на аналогичных принципах.

> **Чтобы думать как люди, машины должны научиться строить причинно-следственные модели мира, которые допускают объяснимость и понимание, а не просто аппроксимировать функцию полезности.**

Они должны основывать процесс обучения на модельных представлениях о физике и психологии, чтобы подкреплять и расширять получаемые знания. Стоит также включать механизмы внимания, внутренней мотивации и эпизодической памяти. Кроме того, важными являются композиционность и способность к метаобучению.

Все эти возможности более естественным образом реализуются в модельном подходе, в котором агент получает возможность отвечать на вопрос «что будет, если...». Если мы в игре Breakout подвинем биту чуть выше, агент в рамках модельного подхода будет способен предсказать, куда шарик отскочит, и сможет выбрать правильное действие. Остается трудный вопрос, как выучить адекватную модель среды, но это возможно, по крайней мере, в принципе. В рамках безмодельного подхода все будущие награды аккумулируются для каждого положения биты, и новое положение будет давать совершенно новые траектории состояний игры, вдоль которых награды нужно будет аккумулировать заново, то есть обучать модель фактически с нуля. То, что при этом будет «чувствовать» модель, сродни тому, что чувствует человек, привыкший, скажем, играть в настольный теннис ракеткой определенных размеров, которому вдруг дали ракетку с более длинной ручкой. В этом смысле безмодельный подход хорошо подходит для описания рефлекса. Но даже в случае с ракеткой человеку не придется обучаться совершенно с нуля.

> **В безмодельном подходе полезной оказывается только информация, имеющая отношение к оценкам полезности состояний и действий, тогда как в модельном подходе используется вся информация для уточнения модели среды.**

Любопытно, что при исследовании процессов формирования рефлексов у животных было обнаружено латентное

обучение, когда животные учились связывать стимулы или формировать навыки без подкрепления. Построение и использование модели среды открывает возможности по решению новых задач в этой среде, для которых переиспользовать результаты обучения безмодельных методов гораздо сложнее.

Правда, системы в рамках модельного подхода получаются более громоздкими и вычислительно затратными, а еще остается вопрос, как потом все-таки вычислять в ней оптимальные решения — число возможных действий получается слишком велико, чтобы их все перебрать. Безмодельный подход позволяет находить выигрышные решения быстрее и точнее, хоть они и будут более «узкими». Чтобы сочетать все плюсы обоих подходов, можно использовать результаты модельных симуляций в качестве входных данных для безмодельного агента, задача которого — научиться интерпретировать неточные предсказания, имитируя процесс воображения.

Есть свидетельства, что люди используют в обучении оба подхода. Например, механизмы безмодельного обучения задействованы в простом ассоциативном обучении (когда мы связываем какое-то условие со значимым для нас последствием — например, если выпить слишком горячий чай, то можно обжечь язык). Но для того, чтобы, например, понять, как вести себя с другим человеком, мы рисуем в голове его ментальную модель с предсказаниями его реакций на разные наши действия в разных контекстах (например, если нас поймали на проступке и мы заплачем, то маму это растрогает, потому что она решит, что мы осознали вину, а папу, наоборот, рассердит, потому что он воспримет это как манипуляцию). При повседневном использовании навыки автоматизируются, вводятся в привычку, вероятно тем самым отражая переход с модельного управления к безмодельному, что позволяет достичь баланса между гибкостью и скоростью принятия решений.

Помимо самого наличия моделей, агенту также необходимо понимание, как ими пользоваться в новых ситуациях и при изменении целей с течением времени. Так что самые большие перспективы — у подходов, которые позволяют управляющим структурам эволюционировать совместно с внутренними моделями.

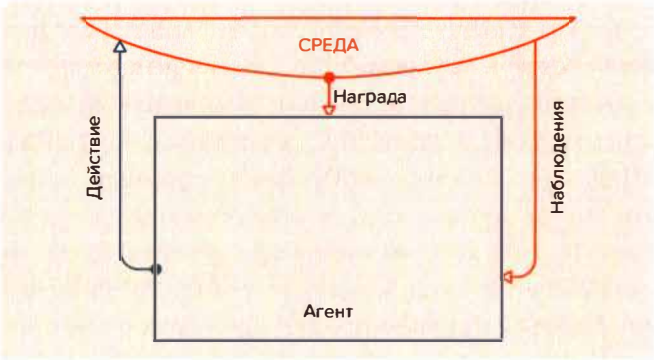
Внутренняя мотивация

В когнитивной психологии выделяют две формы вознаграждения: внешнюю и внутреннюю мотивацию. Первая соответствует изменениям в поведении под воздействием внешних стимулов — например, похвалы или повышения зарплаты. Внутренняя мотивация отвечает за изменения в поведении под влиянием эмоциональных состояний, побуждений, ценностей и опыта — например, если нам любопытно или мы чувствуем себя хорошими, когда выполняем какое-то действие. Наша внутренняя мотивация постоянно меняется в зависимости не только от контекста задачи, но и от ранее полученного опыта. Например, если в вашей семье разделяются определенные ценности, скорее всего, вы переймете какие-то из них. Но если, следуя какой-то ценности (например, честность), вы получите очень болезненный результат (например, разрушите очень важные для вас отношения), у вас может пропасть мотивация и дальше быть честным. И все же внутренняя мотивация считается более действенной, чем внешняя, и играет большую роль в теориях обучения и развития.

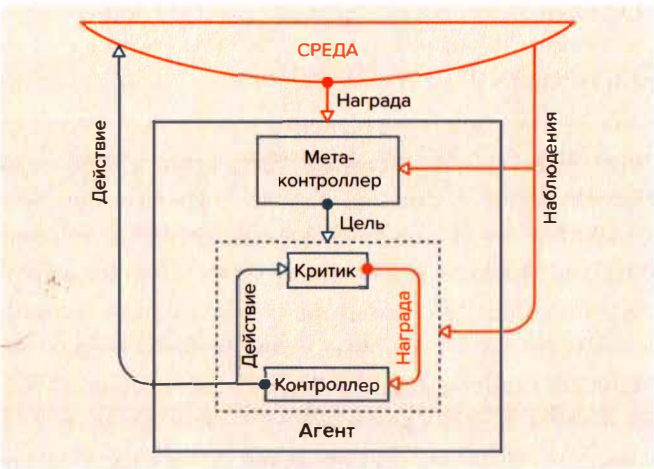
Подход, учитывающий оба вида мотивации, более точно отражает многообразие воздействий, с которыми сталкивается любая обучающаяся система.

> Внешние вознаграждения могут быть переопределены в соответствии с внутренней ценностью для агента, которая может зависеть от текущей цели и его ментального состояния.

Простое



Глубокое иерархическое с внутренней мотивацией



Метаконтроллер учится политике постановки целей в среде, используя глубокую нейросеть. Если агент достигает цели, критик вознаграждает контроллера.
Контроллер учится политике выбора действий, ведущих к достижению целей, используя глубокую нейросеть.

Источник: Tejas D. Kulkarni et al. Hierarchical Deep Reinforcement Learning: Integrating Temporal Abstraction and Intrinsic Motivation, 2016

Рис. 21

Обучение с «внутренним вознаграждением»: модель Теджаса Кулкарни

Например, если вы разочаровались в обществе потребления, высокая зарплата может перестать быть для вас хорошим мотиватором при выборе работы. В обучении с подкреплением центральный элемент мотивации — внешние вознаграждения, а внутреннюю мотивацию начали активно использовать лишь недавно, хотя теории Юргена Шмидхубера (директора Института по изучению искусственного интеллекта Далле Молле в Лугано), которая легла в основу этих разработок, уже больше 25 лет. Если ее удастся встроить, ИИ сможет гибко адаптироваться и корректировать свои цели с течением времени, по-разному расставляя приоритеты по мере приобретения новых знаний и навыков и внесения изменений в требования задачи.

Структура и иерархия стратегий

Многие задачи имеют естественную иерархическую структуру: на верхнем уровне иерархии описываются этапы достижения конечной цели, а на более нижних уровнях определяется последовательность действий для прохождения промежуточных этапов. Мы часто сталкиваемся с этим при планировании своих действий. Вначале мы ставим более общие цели (например, накачать пресс), которые затем разбиваем на более мелкие задачи (например, каждый день делать зарядку из десяти упражнений), а те, в свою очередь, — на подзадачи (каждое упражнение предполагает определенный набор действий).

Таким структурированием задач занимается иерархическое обучение с подкреплением (*hierarchical RL*). Одна из первых работ[73] в этой области появилась в конце 1990-х. Ее авторы ввели понятие опций — многошаговых последовательностей действий, которые доступны для выбора наряду с примитивными действиями. Например, есть действия «порезать морковку», «порезать свеклу» и т.д., и из них можно сложить

[73] Sutton, Precup, and Singh 1999.

Рис. 22

Пример иерархической задачи: поздка из Москвы в Петербург

опцию «сварить борщ». Такой подход позволяет высокоуровневым стратегиям фокусироваться на главных целях и делегировать подстратегиям управление на отдельных этапах. Самое интересное происходит, когда на подзадачах усваиваются правила, которые применимы для достижения других целей.

В реляционном обучении с подкреплением (*relational RL*) основная идея заключается в обучении функций, ориентированных на сущности и отношения между ними, что улучшает способность модели к обобщению и дает возможности интерпретировать ее решения, то есть объяснить их логику понятным для человека образом. Один из примеров успешного применения — работа Замбальди[74], где был использован механизм «самовнимания». Это метод, который помогает определить связи внутри входящих данных — например, в предложении «The animal didn't cross the street because it was too tired» он поможет определить, к какому слову относится местоимение «it». Создателям удалось добиться того, что в четырех из шести

[74] Zambaldi et al. 2018.

мини-игр StarCraft II модель продемонстрировала уровень навыка выше гранд-мастера.

Метаобучение с подкреплением исследует проблему быстрой адаптации модели к новым задачам. Эта область является одной из многообещающих с практической точки зрения, поскольку такой механизм фактически снимает необходимость разработки отдельных алгоритмов для каждой задачи и смещает акцент на создание задач для обучения быстрой адаптации. Чтобы быстрее продвинуться в этой сфере, полезно иметь возможность автоматически формировать такие задачи.

В контексте общего ИИ интерес представляют не столько конкретные безмодельные методы обучения с подкреплением, уже нашедшие разнообразные применения на практике, сколько их многочисленные возможные модификации в совокупности, в особенности включающие одновременное использование как моделей среды, так и безмодельных компонентов, памяти, мотивации и т.д.

> **В результате обучение с подкреплением не столько дает ответ на вопрос о внутреннем устройстве общего ИИ, сколько задает воплощение для него в форме автономного агента, взаимодействующего с неизвестной средой.**

Хотя такая постановка и весьма общая, в ее рамках отдается предпочтение системам определенного рода. Скажем, никто не пытается отталкиваться от обучения с подкреплением при рассмотрении глобального разума как возможного воплощения общего ИИ. Так какого рода эти системы? Чаще всего это или агенты в виртуальных средах или играх, или их аналоги в реальном мире — роботы

Робот как воплощение AGI

Термин «робот» появился в нашем лексиконе примерно на 30 лет раньше, чем «искусственный интеллект». А идея «умных искусственных помощников» насчитывает тысячелетия. И в массовом сознании ИИ часто ассоциируется с роботами. Но связь эта не столь однозначна, как может показаться. Робототехника во многом развивается сама по себе, а искусственный интеллект — сам по себе, поскольку из-за стремительного развития микроэлектроники и значительно более медленного развития материальной базы для создания роботов, эти научные области стали стремительно расходиться в стороны. Быстрый прогресс машин в работе с абстрактными понятиями, в математике, порождал ожидания успеха в создании «сильного ИИ», подпитывал внимание к функционалистским концепциям. Термин «интеллектуальная робототехника» стал интегрирующим ответом, попыткой преодолеть междисциплинарные противоречия, растущие среди исследователей по искусственному интеллекту и робототехнике. Понятно, что без ИИ роботы останутся лишь автоматами, выполняющими рутинные операции, а не андроидами из научной фантастики. Но может ли появиться искусственный интеллект без роботов? И если да, то насколько нам нужен такой ИИ? Роботы нередко рассматриваются как испытательный полигон, мотивационная задача для искусственного интеллекта — полезная, но опциональная. Но, может быть, создание ИИ — не цель, а средство? А цель — именно создание умных роботов?

> **Ряд авторов высказывают идею, что для создания общего искусственного интеллекта и его последующего саморазвития без физического воплощения никак не обойтись.**

Ведь человеческий мозг развивается только при телесном взаимодействии со средой. Но тут надо определиться

с тем, что мы посчитаем достаточным физическим воплощением для компьютера. В последнее десятилетие прошлого века Родни Брукс (извесный робототехник, бывший профессор МТИ и основатель компании Rethink Robotics) предложил применить бихевиористические методики к созданию искусственного интеллекта и роботов, который он назвал «ситуативной робототехникой». Согласно этой методологии, интеллектуальный агент помещается в конкретную физическую среду и действует в нем самостоятельно, опираясь лишь на заложенные в нем поведенческие правила, сенсорную систему и набор актуаторов. Это было полностью противоположно ранее принятому подходу, в котором экспертные системы, основываясь на логико-символическом подходе и репрезентация окружающего мира, делали логические вычисления и принимали оптимальные решения, основываясь на запрограммированных знаниях. Следуя этому подходу, любое техническое изделие человека, действующее в физическом мире, является интеллектуальным роботом (воплощенным ИИ), если обладает тремя способностями, выполняемыми одновременно:

1) чувственно воспринимать окружающий мир или его элементы, используя сенсоры;
2) самостоятельно понимать и обрабатывать получаемую сенсорную информацию о внешнем мире, используя ее для создания и адаптации модели окружающего мира и своего поведения. Например, сунуть механическую руку в ведро с водой, понять, что попадание воды нарушает работу руки, и в дальнейшем избегать контактов с водой (или изобрести новую водостойкую конструкцию руки). Строго говоря, это требование скорее не для физической оболочки, а для «населяющего» ее интеллекта (или для скоординированности интеллекта с оболочкой);

3) действовать, непосредственно изменяя окружающий мир через физический контакт с объектами в соответствии с моделью своего поведения. То есть, условно, если компьютер сам способен вкрутить лампочку, то он воплощен, а если ему приходится давать задание человеку это сделать, то он не воплощен.

Для наглядности приведем примеры воплощенного и невоплощенного Narrow AI. Физическое воплощение виртуальной телеведущей Елены, разработанной в лаборатории робототехники Сбербанка, очень ограниченно: она «живет» в компьютере и у нее нет тела, которое могло бы воспринимать окружающий мир с помощью органов чувств и непосредственно манипулировать материальными объектами. Строго говоря, программа Елены все же меняет физический мир — на уровне пикселей на мониторах. Кроме того, выступление виртуальной телеведущей может привести к реальным изменениям опосредованно — например, если, посмотрев ее телепередачу, люди бросятся менять валюту, это повлияет на экономику. Эти изменения будут гораздо существеннее, чем те, которые доступны, скажем, роботу-пылесосу. Да и с органами чувств не все так просто. Кто-то может смело сказать, что клавиатура компьютера или экран телефона вполне подпадают под понятие сенсора, регистрирующего физическое воздействие, и в этом смысле «органы чувств» у Елены есть.

И ученые спорят, можно ли считать, что у ноутбука есть тело[75]. Но если уж считать, что любой агент воплощенный, иначе бы с ним вообще не могли взаимодействовать, то само понятие воплощенности станет бесполезно. Так что будем считать Елену невоплощенной, подразумевая под этим, что ее возможности по взаимодействию с миром в отсутствие человека

---

[75] Wang P. Embodiment: Does a laptop have a body? Proc. AGI Conf. 2009.

близки к нулю и ее архитектура не предполагает их самостоятельного расширения.

Другой проект этой же лаборатории — робот-курьер — имеет физическое тело и может непосредственно взаимодействовать с материальными объектами (в какой-то степени), так что мы можем назвать его воплощенным (рис. 23). Но ни ведущая, ни робот не создают модель окружающего мира и не выстраивают сложное поведение на ее основе. Елена не составляет свои выступления исходя из широкой повестки и ожиданий зрителей, а генерирует их на базе готовых

1. Чувственное восприятие

2. Осмысление

3. Самостоятельное действие

Рис. 23

Условия воплощенности искусственного интеллекта

новостных текстов, подбирая подходящие по контексту интонации и эмоции. Робот-курьер способен самостоятельно передвигаться по кабинетам и объезжать препятствия, опираясь на сенсорную информацию, но это не означает, что у него в голове есть представление о происходящих в офисе процессах и закономерностях, так что воплощенным интеллектом его назвать нельзя. Тем не менее, нам он кажется ближе к этому. Почему?

Проблема воплощенности связана с обсуждавшейся выше проблемой привязки символов.

> Смысл слов и фраз — в стоящей за ними реальности. Эта связь реализуется через модель окружающего мира. Может ли компьютер реконструировать модель мира, общаясь только посредством сообщений на естественном языке?

Ученые дискутируют, насколько это возможно в принципе, но никто уж точно не пытается этого делать на практике. Именно поэтому у невоплощенной Елены нет практически никаких шансов удовлетворять нашему пункту 2, тогда как у робота-курьера такая возможность есть хотя бы потенциально.

На данном этапе развития ИИ и робототехники мы видим ограниченность и на уровне интеллектуальности, и на уровне воплощенности. С одной стороны, искусственные оболочки роботов все еще отста ют от живых тел по многим показателям (ниже энергоэффективность, меньше степеней свободы, ниже скорость обработки данных с сенсоров и т.д.). С другой стороны, даже самые прогрессивные механические конструкции, сопоставимые по эффективности с людьми или превосходящие их (например, роботы, которые решают задачи по ликвидации последствий стихийных бедствий и техногенных катастроф), в большинстве своем управляются людьми-операторами, поскольку искусственного интеллекта,

которым имело бы смысл «населять» такие «тела», пока не появилось.

Мы снова и снова наблюдаем за тем, как разработчики ИИ, находя работающее при некоторых условиях решение, делают предположение, что при вложении немного больших усилий и немного большего объема данных это решение заработает при любых условиях. Точно так же и разработчики роботов делают предположения, что решения, найденные для узкой задачи управления положением тела в среде, заработают при любых условиях. А это совершенно необязательно так. Например, легко обучить беспилотный автомобиль придерживаться полосы движения на незагруженной дороге в солнечный день. Обучить его тому же в условиях тропического ливня в мегаполисе гораздо труднее. И эта проблема присуща не только беспилотным автомобилям. Разработчики пока не заостряют особого внимания на проблеме робастности, то есть нечувствительности к небольшим отклонениям. Но, например, никто не купит домашнего робота, который доносит дедушку до кровати четыре раза из пяти.

Может ли физическое воплощение помочь развитию AGI?

Сторонники концепции «воплощенного интеллекта» предполагают, что наш, человеческий, интеллект не развился бы до столь высокого уровня без наличия у нас тела. А значит, есть вероятность, что физическое воплощение (необязательно в виде биологического тела, подобного человеческому) может оказаться необходимым для развития систем общего ИИ или хотя бы значительно расширить спектр их возможностей.

Ниже приведены гипотетические преимущества «робо-AGI».

Восприятие. Интеллект человека использует ощущения тела для познания окружающего мира. И в том случае, если

мы стремимся к созданию общего искусственного интеллекта, способного к эффективным действиям в любой ситуации в знакомой нам физической реальности, то нам неизбежно необходимо дать машине возможность к восприятию окружающего мира (зрение, слух, осязание, ощущение температуры, ощущение положения в пространстве и т.д.). Но сенсоры не обязательно должны быть интегрированы в одно-единственное «тело» робота. Вполне можно представить себе ситуацию, когда сенсоры одного робота могут быть сенсорами другого, а искусственный интеллект в облаке способен «видеть» глазами роботов, удаленных от него на миллионы километров.

Здравый смысл. Обладая телом и исследуя окружающий мир с его помощью, ИИ сможет сочетать теоретические познания о мире с непосредственными ощущениями. Например, понимать, почему лучше не заходить в горящий дом и не ехать поперек проезжей части с интенсивным автомобильным движением. Здесь же упомянем и перенос обучения: если ИИ обжигает часть своего физического тела в открытом огне, он узнает, что другие предметы тоже меняют свойства при контакте с огнем. Накопленный сенсорный опыт гипотетически позволяет делать полезные обобщения: например, если квадратные окна с прозрачными стеклами и продолговатыми ручками открываются, то круглое окно с затемненным стеклом и округлой ручкой тоже, скорее всего, открывается. Но если, приложив усилие, сопоставимое с усилием человека, робот не может открыть окно, то высока вероятность, что оно заблокировано.

Правда, в этом рассуждении мы снова исходим из того, как работает человеческий мозг. Воспоминание о боли от ожога в нашем мозге срабатывает быстрее и эффективнее, чем теоретическое знание о том, что прикосновение к горячим предметам может повредить наше тело. Но эта разница значима только тогда, когда нам надо быстро

импровизировать — например, спасти свою кошку из пожара. Если бы мы разрабатывали экипировку для пожарных, нам было бы достаточно теоретических знаний. Так что далеко не для всех задач уровня AGI такой здравый смысл будет иметь значение.

Развитие высших когнитивных процессов. Исследования когнитивистов в области воплощенного познания предполагают, что осознание намного более зависимо от физического тела, чем это считалось ранее[76].

> Некоторые ученые утверждают, что человеческое тело непосредственно участвует в процессе обучения, причем даже с формированием абстрактных концепций[77].

И что даже символьные манипуляции активируют в мозге природные перцептуомоторные (описывающие движение конечностей в ответ на получаемые ощущения) схемы. В книге «Откуда взялась математика» Джордж Лакофф и Рафаэль Нуньес[78] детально рассматривают и приводят примеры того, как математические концепции могут быть глубоко встроены в тело человека и его взаимодействие с окружающим миром. Например, шаги вперед и назад по дороге ассоциируются у нас в уме со сложением и вычитанием.

Но на то, чтобы выйти на абстрактный уровень современной математики, человечеству потребовались тысячелетия. И возможно, существует какой-то более быстрый способ. К тому же есть много примеров того, как «наивная физика» мешает нам развивать научную картину мира. Время интуитивно понятно, но теория относительности родилась

---

[76] https://www.researchgate.net/publication/255635498_A_Short_Primer_on_Situated_Cognition

[77] https://psycnet.apa.org/record/2015-03471-018

[78] http://www.cogsci.ucsd.edu/~nunez/web/FM.PDF

из того, что Эйнштейн поставил это интуитивное, наивное понимание под вопрос. Поэтому важно определиться, с какой целью мы хотим создавать AGI. Если для того, чтобы заменять AGI-системами живых работников, то, наверное, пригодится стиль мышления, похожий на человеческий. Но если мы хотим, чтобы общий искусственный интеллект помог человечеству решить проблемы, с которыми оно пока не справляется, вряд ли стоит загонять его в рамки «человекообразности».

Мастерство. Строго говоря, мастерское исполнение каких-то действий не является ни необходимым, ни достаточным критерием для общего искусственного интеллекта. С одной стороны, можно мастерски делать что-то в рамках узкого ИИ, с другой — ничто не мешает эффективно решать разноплановые задачи в разных средах, используя труд исполнителей-людей. Но, наверное, возможность мастерски осваивать разные навыки принесла бы AGI-системам дополнительные возможности. Загвоздка в том, что мы, люди, овладеваем мастерством только эмпирически, причем мы, как правило, не осознаем те законы, по которым это мастерство работает. Например, велосипедист не знает формул, по которым вычисляет оптимальный угол поворота руля для сохранения равновесия, и владение этими формулами вряд ли помогло бы ему ездить лучше.

> Известный парадокс Маравека говорит о том, что компьютер проще научить играть в шахматы лучше, чем это делают люди, чем научить робота (и тот же компьютер) ловить мячик.

Происходит это потому, что наш физический мир основан на невероятно сложных, но понятных нам законах. Эти же законы мы должны тщательно записать в понятную компьютеру форму, чтобы он смог повторить хоть что-то из того, что доступно для детей.

Правда, и в этом случае речь идет о человекоподобном ИИ. То, что нам не удается получить навык на основе теоретической информации, еще не означает, что это в принципе невозможно. Другое дело, что составлять инструкции для ИИ могут пока что только люди со своей спецификой мышления (возможно, в будущем ИИ будут передавать навыки друг другу каким-то иным способом).

Способность понимать людей. Опыт физического контакта с живыми существами дает понимание их уязвимости. Робот должен уметь вести себя так, чтобы во время физического контакта с незнакомым живым существом не причинять ему вреда. Кроме того, опыт взаимодействия учит соблюдать социальные и культурные нормы: не шуметь, когда спит младенец, помогать пожилым людям и людям с ограниченными возможностями. Тут понадобится некий предустановленный этический кодекс, однако многие непростые моральные выборы мы способны сделать только на практике, используя свой личный опыт взаимодействия с миром.

Правда, не факт, что непосредственное обучение на собственных ошибках окажется наилучшим вариантом с точки зрения выгод и издержек (и гуманности по отношению к живым существам). А продуктивное взаимодействие с другими субъектами, из которого вырастает способность к эмпатии, можно симулировать и в искусственной среде. Кроме того, в ходе обучения с подкреплением можно сформировать у агента стремление не только к наградам, но и к определенным альтруистическим ценностям (которые в обучении с подкреплением будут определяться через предпочитаемое состояние среды — например, мира во всем мире).

С другой стороны, Бен Герцель и другие исследователи считают, что если мы хотим создать AGI, способный к когнитивной эмпатии (то есть умеющий строить ментальную модель другого субъекта, понимать его мотивацию и прогнозировать поступки, а не просто не причинять ему вред), ему

потребуется эмпирический опыт, схожий с нашим. И тело, которое хотя бы отдаленно напоминает человеческое, — хорошее решение для этого. В сознании человеческого ребенка сложным образом смешиваются разные виды данных, и на их основе формируются разные структуры, категории, цели, ценности и т.д.

Выводы

Наш мозг всегда развивался в контексте тела, взаимодействующего с внешним миром для обеспечения своего выживания. И некоторые исследования ссылаются на необходимость получения машинами собственного опыта для обучения. Другие ученые предполагают, что достаточное понимание физических объектов можно сформировать чисто через теоретическое обучение. Но тут возникает вопрос, как ИИ, прошедший «теоретическое обучение», будет далее расширять хотя бы теоретические знания — свои и человечества. Откуда он будет брать новые данные? Если мы его создали только для того, чтобы он мог изучать имеющиеся теоретические знания, то максимум, что он сможет сделать, — это прийти к каким-то новым выводам в рамках существующих моделей и представлений. Если мы хотим, чтобы он имел возможность учитывать новые экспериментальные данные, то возникает вопрос, по каким принципам он будет это делать, как он будет соотносить новые данные с априорными знаниями.

AGI вполне может быть «бестелесным теоретиком» — но тогда ему придется передать эксперименты в руки людей и получать от них собранные данные, чтобы уточнять свои теории и модели. Очевидно, он будет зависеть от людей, и это будет его ограничивать. Но и в таком состоянии он может быть общим интеллектом. Так что речь скорее о том, что робототехника может дополнить и развить возможности AGI, а не о том, что без нее создание общего интеллекта в принципе невозможно.

Значимую роль в обсуждении AGI в робототехнике сыграла серия публикаций Родни Брукса, вышедшая в середине 1980-х[79]. В целом разработки в этой области можно разделить на три уровня. Первый — философские труды, затрагивающие вопросы природы мышления, отличий робота от человека и робоэтики. Второй — теоретические исследования, а третий — практические попытки реализовать AGI и объединить его с робототехникой. Примеры практических разработок — фреймворк OpenCog, уже упоминавшийся в главе про подходы к достижению AGI[80], и общая когнитивная архитектура SOAR[81], главная задача которой — объединение в одном ресурсе всех возможностей интеллектуального агента, от шаблонных задач до заданий с многовариантным выбором.

Эволюционный подход

Общие принципы эволюционного подхода

Вариации и отбор, действующие в природе, привели к появлению огромного разнообразия живых существ с очень разным уровнем когнитивных способностей — от австралийских жуков вида Julodimorpha bakewelli, которые испытывают трудности даже с продолжением рода (оказалось, что самцы этих жуков по ошибке спариваются с пивными бутылками вместо самок), до новокаледонских воронов, способных изготавливать составные инструменты для решения незнакомых

---

79 https://apps.dtic.mil/dtic/tr/fulltext/u2/a174364.pdf

80 https://opencog.org/2015/07/opencog-partners-with-hanson-robotics-to-work-toward-human-like-robots/

81 https://soar.eecs.umich.edu

задач (например, собрать из коротких палочек орудие, чтобы достать угощение из прозрачной коробки с отверстием). Человеческий интеллект — пока что самый универсальный на Земле — тоже продукт биологической эволюции. Значит, в результате эволюции может возникнуть универсальный интеллект, и мы можем сделать этот процесс более контролируемым и направленным, используя эволюционные принципы для отбора искусственных интеллектуальных систем.

> **Эволюционные алгоритмы — это собирательное название разных методов, берущих за основу базовые элементы дарвинистской теории: наследственность, изменчивость и естественный отбор.**

Посмотрим, как это работает, на примере генетических алгоритмов.

Представим, что нам необходимо создать мост через реку, который обладал бы наилучшими характеристиками. Например, мы посчитаем, что мост нам удался, если он обладает и большой надежностью, и низкой стоимостью. Если бы мосты были живыми организмами в своей естественной среде обитания, то из всех вариантов мостов через данную реку самые хорошие мы бы назвали «жизнеспособными». В генетических алгоритмах жизнеспособность обычно задается количественно. Тогда она называется функцией приспособленности, а задача сводится к поиску решения, для которого значение функции максимально (или минимально, если так естественнее для задачи).

Чтобы создать такое решение, нам надо выбрать, в каком виде мы его представим, закодируем. Его аналог в биологической эволюции — это конкретная особь какого-то вида (например, полевка обыкновенная). Характеристики особи зависят от ее генов, поэтому мы можем считать, что параметры полевки как эволюционного решения кодируются в ее генотипе

с помощью ДНК. По аналогии с природой в генетических алгоритмах решения кодируются в виде двоичных последовательностей, играющих роль цепочек ДНК. Далее придумывается, как двоичную последовательность развернуть в особь, например в конструкцию моста, для которой уже можно посчитать функцию приспособленности. Хотя такое преобразование приходится придумывать отдельно для каждой задачи — для мостов одно, для позиций в шахматах совсем другое, — зато потом с этим универсальным представлением можно работать одинаково. Эта работа выполняется *генетическими операторами*.

Они задают правила, по которым создаются генотипы потомков из генотипов родителей. Вот основные генетические операторы:

- **мутация — это случайное изменение в бинарной последовательности. Обычно одна мутация меняет 0 на 1 или 1 на 0 в одном случайном месте. В результате меняется какая-то характеристика особи, зависящая от этого гена, например ширина моста или наличие какой-то балки, но если «фенотип» решения сложным образом зависит от генотипа, то изменений может оказаться и больше;**
- **скрещивание — это рекомбинация, или «перемешивание», генов родителей при создании генотипа потомка, которое в природе происходит через разрыв и соединение разных молекул. Это дает новые комбинации генов и, соответственно, тоже меняет характеристики особи. В генетических алгоритмах существуют разные варианты реализации аналога этого процесса;**
- **отбор — выбор особей для следующей популяции.**

После того как мы определили функцию приспособленности, принцип кодирования и то, какие генетические операторы мы будем использовать, можно начать моделировать эволюционный отбор. Сперва надо создать начальную

популяцию искусственных особей и оценить каждую из них при помощи функции приспособленности. Затем из популяции выбирается первый родитель. Обычно либо случайно берется любая особь из популяции, либо вероятность выбрать ту или иную особь прямо пропорциональна ее функции приспособленности. Как в природе: чем более ты приспособлен, тем выше твои шансы оставить потомство. Потом к родителям применяют оператор рекомбинации, который на их основе генерирует новые решения (потомков). Затем оператор мутации с некоторой вероятностью меняет отдельные гены у потомков. Дальше из получившегося потомства (иногда вместе с родителями) отбираются самые перспективные особи для новой популяции. Чаще всего это происходит либо по принципу пропорционального отбора (чем выше приспособленность особи, тем больше вероятность, что она останется в следующей популяции), либо по принципу элитного отбора (отбираются только самые приспособленные особи).

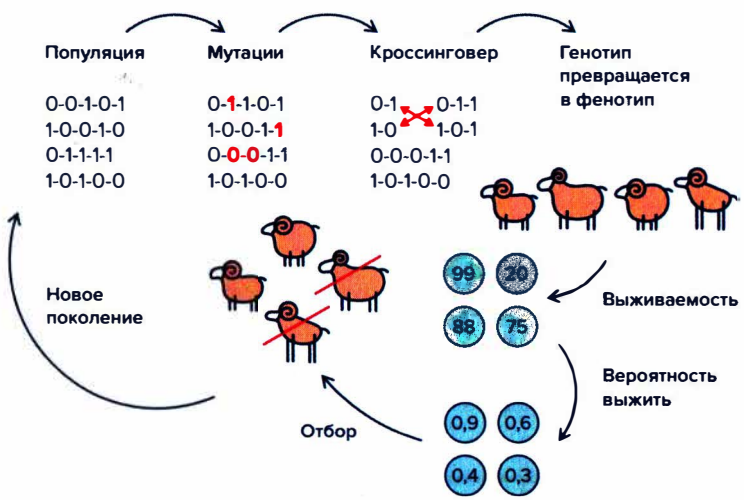


Рис. 24

Эволюционный алгоритм

В новой популяции цикл повторяется. Через несколько циклов мы получаем решения, намного лучше приспособленные к заданным условиям.

Другие ключевые направления эволюционных вычислений

Генетическое программирование, в котором в качестве эволюционирующих особей выступают программы. Роль генотипа может играть вычислительный граф[82], на параметры и структуру которого воздействуют мутации и рекомбинация. В роли эволюционирующих особей могут выступать как алгоритмы решения конкретных задач, так и алгоритмы обучения или даже интеллекта целиком, в связи с чем генетическое программирование представляет особый интерес для AGI.

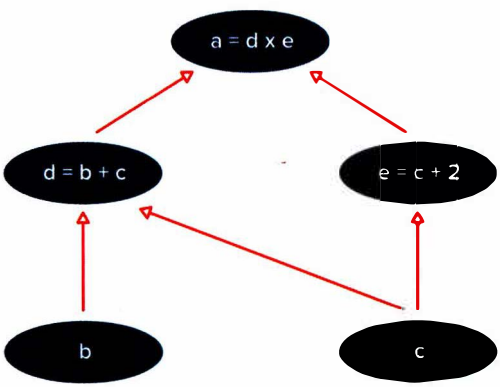


Рис. 25

Простой вычислительный граф для выражения a = (b + c) · (c + 2)

[82] Граф состоит из двух множеств — множества вершин и множества ребер, причем для каждого ребра указана пара вершин (узлов), которые соединяются этим ребром. Вычислительный граф — это граф, где узлы соответствуют операциям и соединены в том порядке, в котором производятся вычисления (см. рис. 26).

Эволюционные стратегии схожи с генетическими алгоритмами, но в них генотипы особей не кодируются в виде битовых строк, а представляются векторами, или массивами, вещественных чисел, что позволяет к таким «генам» применять арифметические операции при скрещивании и мутациях.

Эволюционное программирование раньше выделялось в самостоятельное направление эволюционных алгоритмов. Однако, во-первых, впоследствии оно оказалось сильным упрощением генетического программирования, так как в нем в качестве особей также выступают программы, но с фиксированной структурой. Эволюционируют только их числовые параметры. А во-вторых, с точки зрения реализации эволюционное программирование оказалось частным случаем эволюционных стратегий.

С начала 2000-х гг. в отдельное направление стала выделяться **нейроэволюция** — эволюционная оптимизация нейронных сетей. Самые интересные подходы здесь связаны с тем, что в процессе эволюции может меняться размер нейросети. Если задача усложняется, то сетка начинает расти, включая новые нейроны, позволяющие получить устойчивое решение. И наоборот, в функцию приспособленности можно заложить член, отдающий предпочтение меньшему объему «мозга». Ведь чем больше нейронная сеть, тем она более сложная, медленная и склонная к переобучению (то есть к запоминанию конкретных примеров и нюансов без обобщения).

Эволюционные алгоритмы успешно применяются в различных областях науки и индустриальных приложениях более 30 лет. Сегодня много внимания привлекает направление, связанное с автоматическим поиском архитектуры нейросетей (Neural Architecture Search) — то есть алгоритмами автоматического проектирования систем машинного обучения. Перспективные исследования проводятся в корпоративных лабораториях Google Brain и Uber AI.

Сильные стороны эволюционного подхода

Главное преимущество эволюционного подхода — его креативность. В наиболее часто используемых в машинном обучении оптимизационных алгоритмах, таких как градиентный спуск, мы не можем выбирать структуру решения, а лишь подстраиваем параметры решения с фиксированной структурой. Хотя некоторые типы эволюционных алгоритмов имеют такое же ограничение, но в общем случае мы задаем правила построения архитектуры (какие слои можно брать, например) и дальше это все эволюционирует, самостоятельно подстраиваясь под решение задачи. Кроме того, тут не требуется, чтобы функция приспособленности была дифференцируема, что значительно расширяет класс задач и упрощает оптимизацию. Так что эволюционный поиск более универсален, чем градиентный спуск, и открывает путь к решениям с практически неограниченной сложностью. Если из одноклеточных организмов благодаря эволюции постепенно возникли существа с развитым интеллектом, это дает надежду, что мы можем добиться сопоставимого прироста сложности при применении эволюционных методов in silico[83]. Другой плюс эволюционных алгоритмов — то, что они очень легко и хорошо распараллеливаются: мы можем разбить популяцию на части и рассчитывать приспособленность особей в них параллельно, что ускоряет получение новых решений.

Недостатки и ограничения эволюционного подхода

Эволюционный алгоритм генерирует много случайных решений, а это значит, что большая часть вычислительных ресурсов тратится на генерацию и оценку решений, которые потом не пригодятся. При этом есть другие методы метаэвристиче-

---

[83] То есть «в кремнии» — по аналогии с in vitro (лат.), что означает «в пробирке».

ской оптимизации[84] (например, метод имитации отжига, алгоритм «муравьиной колонии», поиск с запретами и ряд других), и некоторые из них тоже позволяют работать на уровне структуры моделей, хорошо распараллеливаться, работать с недифференцируемыми функциями и т.д. Опыт показывает, что, хотя в ряде задач эволюционные вычисления превосходят альтернативы, в других задачах они могут уступать имитации отжига или даже случайному поиску.

Кроме того, аналогия между эволюционными алгоритмами и естественной эволюцией на самом деле очень поверхностна. Естественная эволюция изобрела множество разных механизмов (управление мутациями, генные сети, различные эпигенетические механизмы и т.д.), а стандартные эволюционные алгоритмы до сих пор используют лишь одну простейшую эвристику, предзаданную человеком. Время от времени происходят попытки промоделировать аналог того или иного механизма естественной эволюции, однако без громких практических успехов.

Тенденции и перспективы эволюционного подхода для AGI

Несмотря на то что эволюционные алгоритмы вряд ли можно считать универсальным решением, они в среднем превосходят другие методы по широте применения или эффективности поиска, будучи при этом простыми в исполнении. Они играют важную роль в различных подходах к AGI и используются в глубоком обучении, универсальной индукции, вероятност-

---

[84] Метаэвристика — это эвристика, не опирающаяся явно на специфику конкретной задачи, но при этом часто неплохо работающая во многих задачах. В играх в качестве метаэвристики может выступать правило: «Предпочитай ход, после которого у соперника остается меньше альтернативных ходов», — которое не ссылается на правила игры и которое работает часто, но не всегда. В задачах поиска скрещивание решений — это тоже метаэвристика.

ных моделях, когнитивных архитектурах. Развитие принципов эволюционного метаобучения на фоне прогресса в аппаратном обеспечении может привести к тому, что эволюционные алгоритмы будут играть одну из ведущих ролей в области общего ИИ. Но могут ли они использоваться не внутри систем AGI в качестве одного из их компонентов, а снаружи — для поиска и оптимизации самих интеллектуальных агентов?

Искусственная жизнь

Программы, которые закладываются в аниматов и когнитивных роботов, очень сложны для написания. Одному агенту в процессе обучения сложно получить необходимый объем информации, чтобы как следует приспособиться к среде. В природе эта информация накапливается в процессе эволюции, который затрагивает множество видов и позволяет выжить наиболее приспособленным.

> **Было бы здорово смоделировать процесс эволюции какого-то виртуального мира на компьютере, где искусственные существа будут эволюционировать и умнеть.**

Но эволюция шла миллиарды лет и затрагивала миллионы видов и неисчислимое количество особей, существовавших одновременно, и даже на компьютере, где мы можем «сжать» сутки и годы, это все равно займет очень много времени. Либо же нам придется делать эволюцию гораздо более «умной», но будет ли разработка умной эволюции проще, чем сразу умного интеллектуального агента?

Тем не менее попытки воспроизводить процессы эволюции в виртуальном мире все равно приносят пользу. Это направление исследований называется «искусственная жизнь». Как правило, конструируется небольшой виртуальный мир с простыми законами. Основная цель — раскрыть, смоделировать и воспроизвести принципы организации процесса

биологической жизни и процесса ее развития в ходе эволюции. Но это упрощенные модели, без попытки точно воспроизвести реальные биологические механизмы. Например, виртуальный мир может выглядеть как поле, разбитое на клетки (рис. 26). Клетка может быть пустой, в ней могут находиться растение (*), хищник (Х) или травоядное (Ж). Травоядное животное получает информацию о том, что находится в смежных клетках, и может принять решение переместиться или вступить во взаимодействие с этими объектами (например, травоядное может съесть растение или убежать от хищника).

Способностью к развитию можно наделить один вид или несколько. Как правило, эволюционируют не физические параметры животных, а программы управления (например, при эволюции червяков в 3D-симуляции для начала нам будет важно, чтобы они научились ползать как червяки, а не чтобы

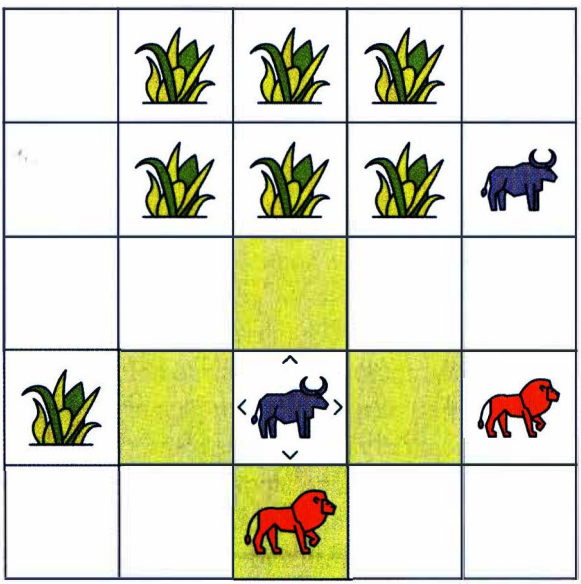


Рис. 26

Пример фрагмента «искусственного мира»

отрастили ноги или крылья). Такие эксперименты ставятся для проверки гипотез об эволюционных механизмах. Например, можно проверить роль разных эмоций в выживании.

Один из наиболее интересных (и значимых в контексте AGI) вопросов, которые могут быть поставлены в рамках направления «искусственная жизнь», — это вопрос о том, в любом ли мире мог возникнуть интеллект? Не в плане физической выживаемости, а в плане условий для выбора, которые ставит среда. Судя по экспериментам на виртуальных мирах, подходящий мир не должен быть слишком простым (в этом случае будет слишком мало условий для развития) или слишком сложным (если найти закономерности, на которых можно выстроить эффективное поведение, будет слишком трудно или невозможно, то агенту выгоднее выбрать случайное поведение). В идеале он должен постепенно усложняться — как постепенно усложняются условия в процессе обучения ребенка. В живой природе усложнение дают другие виды.

> **Искусственная жизнь дает более конкретное понимание того, как интеллект мог возникнуть в процессе эволюции. Но в то же время эксперименты в этой области делают очевидным тот факт, что эволюционное создание AGI с нуля — вычислительно невыполнимая задача.**

Возможно, процесс разработки систем AGI сродни эволюционному: есть и отбор более приспособленных решений, и скрещивание различных когнитивных архитектур или нейросетевых моделей с изменчивостью отдельных компонентов системы (например, когнитивная архитектура OpenCog «скрещивалась» и с NARS, и с OpenPsi, которые впоследствии «мутировали» не так, как оригиналы). Однако все это делается вручную людьми. А значит, создание AGI мы не может отдать на откуп искусственной эволюции и именно мы, люди, определяем форму воплощения AGI как конкретной особи или вида.

AGI как экосистема: воплощение через платформенные решения

Еще один вариант воплощения общего искусственного интеллекта — это экосистема ИИ-сервисов. Движение в сторону общего ИИ требует все более и более широкоприменимых алгоритмов. Сегодня каждый тип интеллектуальных систем достаточно хорошо решает некоторый спектр задач, но не справляется с задачами из соседних областей, тогда как более универсальные методы пока плохо масштабируются на задачи реального мира. Наиболее очевидное, хотя и не единственно возможное, решение этой проблемы — создание гибридных систем ИИ, оптимально сочетающих сильные стороны множества разработанных типов алгоритмов.

Работа над гибридными системами ИИ — сложная задача. Небольшие коллективы обладают ограниченным разнообразием компетенций в разных направлениях и нередко концентрируются на узкой области, чтобы быть конкурентоспособными. Кроме того, в небольших исследовательских группах много времени тратится на воспроизведение наработок, полученных коллегами, что значительно замедляет прогресс в развитии новых идей. При этом даже создание крупных групп в специализированных национальных и корпоративных исследовательских центрах не позволяет максимально эффективно объединять наработки.

> **Возможно, создание гибридного общего ИИ можно сравнить с проектом по созданию первой атомной электростанции или запуском человека в космос, что требовало координированных усилий большого числа ученых и инженеров.**

Но, возможно, развитие интеллектуальных систем до уровня общего ИИ будет больше напоминать развитие интернета. И тогда более эффективным будет не централизованное

решение с полным пониманием того, что и как делается, а создание экосистемы из разных исследовательских групп с обеспечением их условиями для эффективного обмена идеями и кодом. У разработчиков должна быть возможность быстрого прототипирования — а для этого надо стандартизировать фреймворки искусственного интеллекта и публиковать исходные коды в открытых репозиториях. Кроме того, должны поощряться проекты, направленные на создание не узкоспециализированных систем, а алгоритмов, умеющих на приемлемом уровне решать широкий спектр задач. В той или иной мере этот сценарий уже начинает реализовываться.

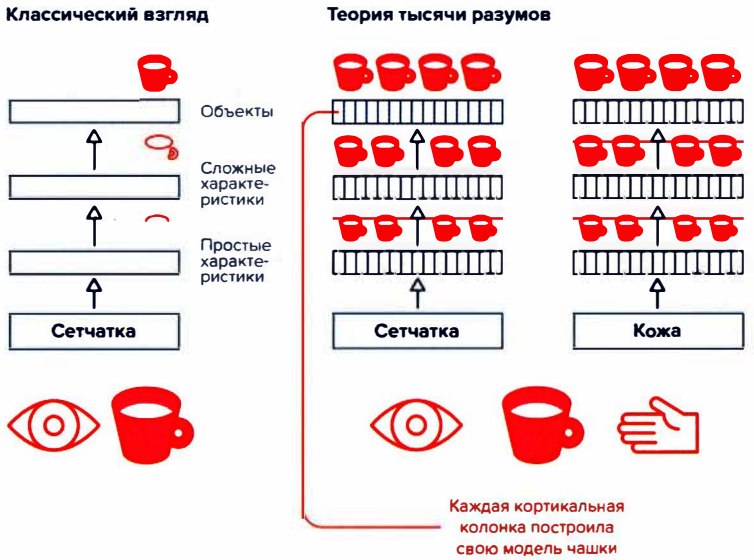
> **В пользу экосистемности говорят не только практические соображения, но и современные теории интеллекта как следствия эмерджентности, то есть появления у системы свойств, не присущих ее элементам в отдельности.**

В частности, концепция «Общества разума» Марвина Минского — одного из ученых, стоявших у истоков ИИ, — предполагает, что интеллект человеческого уровня может проявляться как сложное целое, возникшее при взаимодействии более простых агентов, каждого из которых нельзя назвать разумным.

Можно провести аналогию с нашим телом: каждая клетка выполняет довольно простую функцию, но вместе они составляют сложный организм, способный играть на скрипке или выполнять акробатические трюки. Эта концепция находит подтверждение в последних работах изобретателя Джеффа Хокинса и его коллег, сформулировавших «Теорию тысячи разумов» на основе своих исследований в нейрофизиологии и компьютерном моделировании[85]. Согласно этой теории, главную

---

[85] https://numenta.com/neuroscience-research/research-publications/papers/a-framework-for-intelligence-and-cortical-function-based-on-grid-cells-in-the-neocortex/



Источник: Jeff Hawkins et al. A Framework for Intelligence and Cortical Function Based on Grid Cells in the Neocortex, 2019

Рис. 27

Как наш мозг воспринимает чашку кофе: классическая модель и модель Джеффа Хокинса

роль в возникновении интеллекта играют так называемые нейроны решетки в коре головного мозга — в более древних областях мозга эти нейроны активируются, когда мы пересекаем узлы воображаемой координатной сетки в пространстве (они помогают нам ориентироваться на местности и лавировать между находящимися на ней предметами). А с помощью нейронов решетки мы изучаем окружающий мир на более абстрактном уровне и строим в уме модели разных объектов. Каждая решетка создает свою модель, а их совмещение дает сложную картину мира.

Соответственно, у гибридной системы, совмещающей разные типы алгоритмов, больше шансов выйти на нужный уровень эмерджентности. Об этом же гласит и гипотеза

когнитивной синергии Бена Герцеля. Согласно этой гипотезе, разнообразные подсистемы общего ИИ, работающие с информацией разного вида, должны взаимодействовать таким образом, чтобы помогать друг другу в преодолении комбинаторного взрыва[86]. Это гипотезу косвенно подтверждает то, насколько сложно оказалось оценивать частичный прогресс на пути к общему ИИ. По мнению Герцеля, дело в том, что даже разработанная на три четверти система может не проявить свойства когнитивной синергии и вести себя как система узкого ИИ.

При этом для успеха в создании платформенных решений важно наличие репутационной системы, действующей в масштабе всего «общества разума»[87]. Такая система будет обеспечивать эволюционный отбор наиболее эффективных подходов и приводить к наиболее выигрышным для всех социальным договоренностям. Поскольку платформа будет объединять множество участников с разными уровнями доверия, важно снизить риск воздействия со стороны наиболее некомпетентных и недобросовестных «звеньев».

Примеры проектов

Гибридные proto-AGI системы можно разделить по глубине их интеграции. В частности, глубоко интегрированными гибридными системами являются некоторые когнитивные архитектуры. Благодаря открытости их кода можно говорить о том, что вокруг них формируется экосистема для создания общего ИИ.

Один из уже упоминавшихся примеров такой когнитивной архитектуры — OpenCog — проект, направленный на создание

---

[86] Термин, используемый для описания эффекта резкого («взрывного») роста временной сложности алгоритма при увеличении размера входных данных задачи.

[87] A. Kolonin, B. Goertzel, D. Duong, M. Ikle. A Reputation System for Artificial Societies. arXiv:1806.07342 [cs.AI].

платформы искусственного интеллекта (метаархитектуры для AGI-исследований) с открытым исходным кодом. Проект объединяет ряд технологий для создания интеллектуальных систем: гиперграфовую базу знаний, систему сопоставления подграфов для извлечения знаний из базы, унифицированную машину для построения цепочек рассуждений на основе знаний, реализованную на ее основе концепцию вероятностных логических сетей, систему эволюционного программирования, экономические сети для управления вниманием, систему мотивации OpenPsi, модули для работы с естественным языком, интеграцию с сетями глубокого обучения и ряд других компонентов, взаимодействующих друг с другом через общий гиперграф знаний.

Подобные проекты, хотя и обладают некоторой долей экосистемности, из-за глубокой интегрированности компонентов требуют от разработчиков большой вовлеченности.

> **Один из немногих экосистемных проектов, принимающих во внимание цель создания общего ИИ, — это проект по разработке платформы с открытым кодом SingularityNET.**

В нулевом приближении ее можно охарактеризовать как децентрализованный рынок сервисов ИИ. В отличие от платформ виртуальных ассистентов, таких как Alexa, в которых обычно ядром (нередко закрытым) является узкий ИИ, аддитивно расширяемый узкими навыками, SingularityNET стимулирует создание не специализированных изолированных предметных навыков типа заказа пиццы, а именно сервисов ИИ, которые могут вызывать друг друга, в том числе формируя и более универсальные сервисы, решающие все более широкий круг задач. Так, в качестве одного сервиса ИИ может выступать отделение в песни музыки от вокала, а в качестве другого — распознавание речи, и эти сервисы могут комбинироваться. Более того, сервис может предоставлять не готовую

модель, выполняющую конкретную функцию, а услуги по обучению модели на пользовательских данных. При этом другой сервис может выполнять генерацию или дополнение данных и комбинироваться с сервисом обучения модели. Третий может выбирать среди сервисов, предоставляющих обучение моделей, наиболее подходящий под текущую задачу и т.д.

> **Пример проекта, экосистемность которого имеет другое выражение, — Open Mind Common Sense (OMCS) — проект, основанный в Медиа-лаборатории Массачусетского технологического института (MIT), целью которого и его целью являлось создание и использование обширной базы знаний на основе открытого сбора высказываний пользователей через интернет.**

С момента своего основания в 1999 г. он собрал более миллиона фактов от более чем 15 000 участников. Основной результат проекта — семантическая сеть ConceptNet, распространяемая под открытой лицензией и широко используемая в исследованиях и технологических ИИ-проектах. Сегодня ConceptNet поддерживается компанией Luminoso Technologies.

Несмотря на большое число участников, их вклад в этом проекте весьма ограниченный. Можно ли получить вклад от столь большого числа участников не просто в форме фактов для базы знаний, а в форме именно алгоритмов и компонентов искусственного интеллекта?

> **Платформы для виртуальных помощников, таких как Amazon Alexa, Google Assistant, «Салют» и «Яндекс Алиса», являются возможным вариантом решения. Они позволяют собирать для пользователя в одной точке входа множество различных «навыков». При этом неважно, какое направление ИИ использовалось для создания отдельных навыков. По мере того как растет число навыков и повышается уровень владения отдельно взя-**

тым навыком, суммарный интеллект виртуального помощника становится все более и более широким.

Следующий этап развития виртуальных помощников — создание открытых платформ. Пример такого проекта — DeepPavlov, разрабатываемый МФТИ при поддержке Сбербанка. DeepPavlov предоставляет инструменты для разработки отдельных навыков, а также для их интеграции в целостного виртуального помощника. Другой проект по созданию открытой платформы — Open Virtual Assistant — ведется в университете Стэнфорда.

Сегодня неизвестно, можем ли мы прийти к общему ИИ через один универсальный метод, самостоятельно изобретающий эффективные специализированные алгоритмы и модели под каждую конкретную задачу, или нам придется создавать для этого гибридную систему. Самый прагматичный подход к ускорению прогресса с целью приближения к общему ИИ заключается в следующем:

1) создание условий для форсирования исследований гибридных архитектур, чтобы как можно быстрее получить системы с максимальным спектром решаемых задач;
2) поддержка исследований, максимально отличающихся от известных сегодня подходов;
3) поддержка публикации базовых инструментов и алгоритмов с открытым кодом.

Тогда движение в сторону общего ИИ будет выглядеть как последовательность все более и более универсальных гибридных систем. В каждый отдельный момент наиболее «продвинутая» ИИ-система должна комбинировать навыки, (не только и не столько предметные, сколько общекогнитивные), созданные при помощи разных подходов. В ближайшем

будущем стоит развивать экосистемы исследований и разработок вокруг платформенных решений. И тогда, вполне возможно, общий ИИ возникнет как распределенный интернет-разум на базе одной из платформ ИИ-сервисов или их объединения.

А может, просто загрузить мозг в компьютер?

Нейронауки и искусственный интеллект

Нейронауки изучают физические процессы головного мозга и их связь с психикой и поведением. Это одна из наиболее мультидисциплинарных научных областей нашего времени (пересекается с ИТ, медициной, бихевиористикой, социальными науками, молекулярной биологией, экономикой и др.) и одновременно одна из наиболее быстро и активно развивающихся в последние десятилетия. Нейронауки развивают наше понимание природы человека и того, что же именно делает человека человеком, предлагая свои гипотезы и концепции для объяснения того, какие механизмы лежат в основе мышления, эмоций, поведения и других феноменов человеческой психики.

> Но в связи с популярностью глубокого обучения основное внимание при разработке ИИ уделяется лишь одной области нейронауки — коннектомике.

Суть ее заключается в том, чтобы изучать мозг как систему, каждая часть которой как-то связана с другой, и пытаться воссоздать эти системы связей на разных уровнях. Направление

коннектомики под названием Whole Brain Emulation (WBE), то есть «полная имитация мозга», можно считать альтернативой AGI — вместо того чтобы создавать продвинутый ИИ с нуля, мы могли бы оцифровать человеческий мозг и воссоздать его работу на компьютере во всех деталях.

Два уровня WBE

Картирование коннектома на микроуровне (с разрешением в микрометры) означает постройку полной карты нейронной сети — всех синаптических связей от нейрона к нейрону. Одна только кора головного мозга содержит порядка 10 млрд нейронов, соединенных 10^{14} синаптическими связями. Для сравнения: число пар оснований в человеческом геноме — 3×10^9.

Но при попытке реализации возникают проблемы:

- описание всех взаимосвязей между нейронами необязательно приводит к пониманию психики. Полная модель взаимосвязей всех нейронов у нематоды (круглого червя) Caenorhabditis elegans существует уже больше 20 лет, но мы до сих пор не можем до конца описать даже его поведение;
- на сбор данных о человеческом мозге требуются годы при нынешних технологиях;
- инструменты машинного зрения для этой предметной области на сегодняшний день находятся в зачаточном состоянии;
- данные большинства исследований на сегодня имеют разную структуру (в том числе в силу кросс-дисциплинарности), и их сложно объединить в общую базу.

Что с этим можно сделать? Для решения проблем со сбором данных несколько групп исследователей создают

серийные электронные микроскопы с высокой пропускной способностью, а статистическая теория графов развивает способы распознавания образов и инструменты для обработки изображений. Если мы преуспеем в развитии этих технологий, мы сможем быстрее разобраться в том, как работают разные нейронные цепочки и системы и в каких психологических реакциях эти процессы отражаются. Это может оказаться полезным как для развития психологии, психиатрии и нейронаук, так и для понимания того, как устроен человеческий интеллект. Возможно, это понимание приблизит нас и к созданию «человекоподобного» общего искусственного интеллекта, если мы решим пойти к AGI таким путем.

«Мезо»-коннектом соответствует масштабу в сотни микрометров. Вместо того чтобы пытаться отобразить каждый отдельный нейрон, в этом случае мы захватываем анатомически и функционально различные популяции нейронов, соединенных в локальные цепи (например, кортикальные колонки — функциональные единицы коры головного мозга, формирующие и принимающие сигналы).

> **В чем проблема: для того чтобы построить такую модель, неинвазивные методы нам не помогут — придется залезть в черепную коробку живого человека, и здесь этика нас серьезно ограничивает.**
>
> **Что с этим можно сделать? Возможно, в будущем появятся миниатюрные нейроимплантаты, которые помогут преодолеть это ограничение.**

Основные проекты

Blue Brain Project (IBM, Швейцарский федеральный технический институт Лозанны).

Цель этого проекта, который стартовал еще в 2000-х гг., — создание полноценной компьютерной симуляции головного

мозга млекопитающего, основанной на моделях, воспроизводящих работу нейронов на биологическом уровне. Для этого использовался метод обратного проектирования: вначале на основе биологических данных (в данном случае это были срезы мозга крысы, на которых проводили эксперименты с окраской и электрической стимуляцией нейронов, чтобы понять, как все работает) на компьютере создаются модели отдельных нейронов, а потом из них выстраиваются более сложные «модули» — кортикальные колонки (рис. 28).

В 2008 г. участникам удалось продемонстрировать первую работающую колонку, аналогичную колонке неокортекса

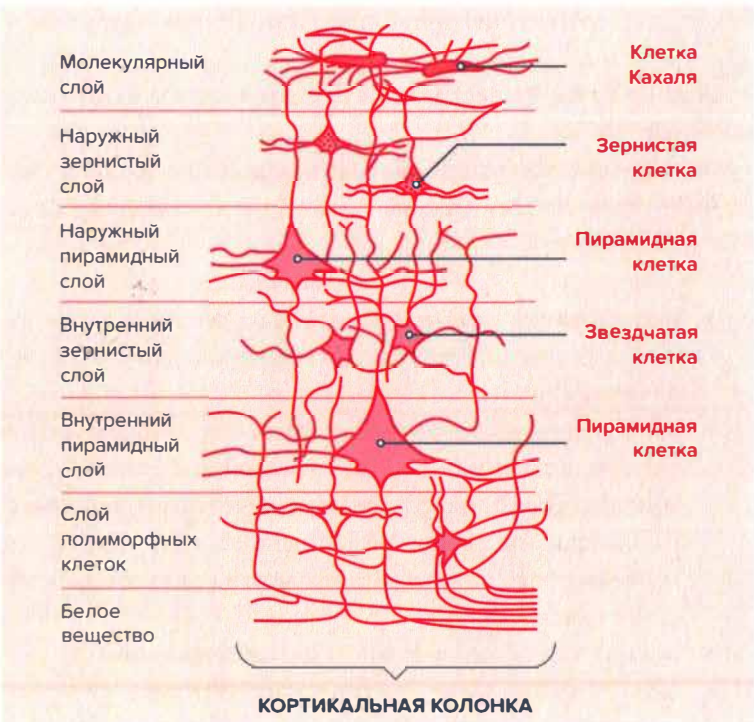


Рис. 28

Схема расположения коры головного мозга

крысы. Получая на входе те же самые сигналы, на выходе модель генерировала те же сигналы, что и настоящая нервная ткань животного, и с ее помощью исследователям удалось предсказать большую часть синаптических связей, возникающих между нейронами в участке крысиного мозга. Позже исследователи смогли построить 3D-симуляцию неокортекса мыши, но в процессе исследований обнаружили, что клеточная структура и соединения куда сложнее, чем предполагалось. Были обнаружены новые типы нейронных связей, которые пришлось интегрировать в модели. Судя по всему, проект продолжает сталкиваться с новыми и новыми сложностями и ограничениями, которые не позволяют исследователям приблизиться к пониманию принципов функционирования мозга.

Human Brain Project (ЕС, Федеральная политехническая школа Лозанны).

Этот проект окрестили «большим адронным коллайдером от нейрофизиологии». Главный акцент в нем ставится на следующих моментах:

- исследования разных уровней и принципов организации мозга мыши (для того, чтобы сравнить его с человеческим);
- заполнение пробелов между данными на микроуровне (полученными в результате трехмерной электронной микроскопии) и макроуровне (полученными, например, с помощью магнитной резонансной томографии);
- изучение того, какие области мозга и как взаимодействуют между собой;
- построение общей математической модели мозга;
- моделирование мозга на суперкомпьютерах.

В исследовании принимают участие сотни ученых из 135 научных центров в 26 странах Евросоюза, и в целом команда

испытывает большие сложности с объединением данных из различных исследований для формирования выводов и создания унифицированной базы знаний. Кроме того, наличие нескольких команд с разными задачами поднимает вопрос о ключевых приоритетах и вызывает споры из-за распределения финансирования.

China Brain Project (научные учреждения КНР).

По имеющимся данным, основная задача CBP — исследование когнитивных процессов, относящихся к высшим психическим функциям (сознание, внимание, память), чему способствует, в частности, использование в качестве экспериментальной модели крупных приматов. При этом исследуются как макро-, так и микроуровни психики.

Ключевые задачи:

- разработать эффективные подходы к ранней диагностике и лечению болезней мозга;
- создать новое поколение технологий ИИ, основанных на принципах работы мозга.

BRAIN Initiative (DARPA, IARPA, частные компании и университеты США, Австралии, Канады, Дании).

Приоритет программы — не столько технологии, сколько разработка и использование методов получения фундаментальных знаний о работе нервной системы. В первую очередь — анализ сетей взаимодействующих нейронов и понимание того, как динамические паттерны нейронной активности становятся мышлением, эмоциями, восприятием и принятием решений в здоровом и больном мозге. Для изучения мозга предполагается в том числе использовать наноимплантаты, передающие информацию об активности нейронов и синапсов с помощью беспроводной связи.

Brain/MINDS (Министерство образования, культуры, спорта, науки и технологий Японии).

Основные направления исследований — изучение психических функций приматов и использование полученных результатов в медицине, а также разработка технологий создания коннектома и сам процесс его создания.

Чем все это может помочь развитию AGI?

С точки зрения AGI нейронауки можно использовать как источники новых данных и принципов для программного обеспечения. Основные преимущества такого подхода заключаются в том, что благодаря исследованиям мозга возникают математические модели, которые описывают электрохимические процессы, происходящие в нейронных цепях, и ограниченные симуляции разных участков коры, а также помогают лучше понять происходящие в мозге процессы (и, соответственно, искусственно смоделировать их при необходимости). Также возникают различные прототипы инвазивных и неинвазивных нейроинтерфейсов, что двигает вперед как нейронауки, так и бионику с робототехникой. Кроме того, эволюционный отбор делает устройство человеческого мозга решением, близким к оптимальному, а значит, имеет смысл сравнивать альтернативные решения с изобретениями природы, даже если мы отказываемся от буквального копирования.

Но у коннектомики есть и слабые места.

> Мозг — это невероятно сложная система, и для того чтобы разобраться во всех типах нервных клеток и взаимосвязей между ними, нам нужно получать гораздо более точные данные.

При этом у нас пока нет ни высокоточных неинвазивных нейрогаджетов, которые позволяли бы изучать работу мозговых структур «крупным планом», ни технологий искусственного культивирования нервных тканей, чтобы экспериментировать

с ними в лаборатории. При этом биоэтика очень сильно ограничивает ученых во вмешательстве в мозг живых людей, а мозг мертвых неинформативен, потому что активность в нем останавливается. И наконец, очень сложно объединить данные из разных научных областей в единую систему, понятную для всех участников.

Так что воспроизвести человеческий мозг в цифровой среде в обозримой перспективе вряд ли возможно.

> **С другой стороны, это не такая уж и хорошая альтернатива AGI: мозг человека обладает множеством недостатков, имеет существенные ограничения, не способен решить задачи, которые уже сейчас решает узкий искусственный интеллект.**

Наконец, нельзя не учитывать этические аспекты проблемы, а также возможные социальные риски, связанные с воспроизводством человеческого мозга или эмуляцией «личности».

Зато можно использовать новые знания о принципах работы мозга для создания общего искусственного интеллекта, вне привязки к физиологическим и нейробиологическим характеристикам живой ткани. Один из примеров такого подхода — теория **Brain principles programming (BPP)**, разработанная в Лаборатории нейронаук и поведения человека ее научным руководителем Курпатовым А. В. Суть теории состоит в том, что деятельность мозга является, по существу, информационной моделью реальности, которая постоянно соотносится с полученными от фактической реальности откликами и корректируется в зависимости от целей (решаемых мозгом задач).

Под «принципами» в BPP понимаются универсальные механизмы работы с информацией, которые реализуются на всех уровнях организации нервной ткани (начиная с клеточного и заканчивая системным) и могут быть описаны соответствующими математическими моделями. Аналогов теории нет

в той степени, что целостно бы охватывала принципы работы мозга как единый системный процесс.

Формализация принципов работы головного мозга в Brain principles programming как шаг к техническому воплощению AGI

Не так давно рядом отечественных исследователей была предпринята успешная попытка формализовать в языке теории категорий[88] основные методологические концепты:

- *«интеллектуальный объект»*, под которым мы понимаем любую единичную целостность, выделяемую нами в этом пространстве — например, когда мы видим стол, сигналы от зрительного нерва обрабатываются мозгом, и сочетание отдельных линий опознается как стол;
- *«интеллектуальная функция»*, которая описывает все возможные операции в рассматриваемой системее — это все, что психика может сделать с интеллектуальным объектом. Когда мы опознаем стол как объект, мы можем оценить его размер или придумать, как его использовать;

---

[88] **Теория категорий** — раздел математики, изучающий свойства отношений между математическими объектами, не зависящие от внутренней структуры объектов. Она главным образом фокусируется на отношении между объектами — то есть важно не что именно соотносится между собой, а то, что получается в итоге таких отношений. Иными словами, теория категорий не интересуется внутренним устройством объекта, определяя его через совокупность всех отношений, в которые он может вступить с другими объектами категории.

- *«сущность»* — специфическое значение объекта для психики. То есть, знание о том, для чего можно использовать стол.

Такая формализация очень продуктивна как с точки зрения унификации способов думать о мышлении и работе мозга в частности, так и с точки зрения поиска единого языка научной мысли в целом.

По мнению исследователей, перевод «методологии мышления» на формальный язык *теории категорий* был осуществлен довольно «естественным» образом — в том смысле, что в ходе работы не пришлось ничего намеренно подгонять под выбранные заранее определения. То есть неформальные, но все же внятные и рациональные рассуждения могут быть переформулированы на языке теории категорий, который, будучи частью алгебры, уже является полностью формальным.

Другими словами, на этом основании есть возможность предположить, что существует некоторая универсальная систама символов, которая может быть эффективно использована в качестве некоего единого универсального языка любой науки. Ею может и не быть теория категорий в том виде, в котором она известна в настоящее время, но уже сейчас представляется вполне вероятным, что основу такого языка все же будут составлять объекты, стрелки и согласованные определенным образом правила действий с ними.

Благодаря этому языку принципы оперируют понятиями целостного Мира (среды), позиции наблюдателя в нем, вещей как элементов неупорядоченных множеств и их отношениями.

> **Основная идея заключается в том, что некий наблюдатель — субъект опыта, которому является Мир — присваивает значения вещам исключительно через призму взаимодействия с ними.**

В этом и проявляется специфика познавательных способностей каждого индивида — «вещи, которыми мне является Мир, существуют для меня и лишь в отношении со мной». Вышеперечисленные аспекты и формируют в теории ВРР понятие «интеллектуальный объект», а именно, это:

- некоторый набор данных (A);
- наблюдатель, как отражение мира после взаимодействия с ним (Ω);
- отношение мира с наблюдателем, являющееся функцией внутреннего состояния/ожидания (f) — интеллектуальная функция.

$$A \xrightarrow{f} \Omega$$

В более же узком, прикладном аспекте категорная формализация фактически совпадает с алгоритмизацией тех основополагающих принципов, на основании которых, согласно теории ВРР, мозг человека обрабатывает информацию и принимает решения. Что дает мощный действенный инструментарий специалистам, занимающихся проблемами общего искусственного интеллекта. Дальше мы перечислим основные пять признаков работы мозга.

1. Принцип генерации сложности

Интеллектуальная функция в рассматриваемом нами контексте выступает единственным инструментом мышления, используя которую мы создаем новые отношения между интеллектуальными объектами и сами эти отношения тоже, по существу, есть новые интеллектуальные объекты. Мозг работает с весьма ограниченным объемом информации от окружающей его реальности, поступающим на его сенсоры в виде аналогового сигнала (химические, электромагнитные, механические раздражители), и которая затем преобразуется в сигнал внутреннего пространства психической деятельно-

сти. По мере использования этой, изначально скудной информации мозг, на всех уровнях своей организации многократно увеличивает ее объем, соотнося полученные вводные с уже существующими в нем данными (часть таких данных являются генетически предопределенными, другая часть — накопленным опытом).

Принцип генерации сложности позволяет мозгу, получив самый незначительный внешний сигнал, воспроизвести в сознании человека знание (интеллектуальный объект) несопоставимо большей мощности, обогатив модель этого объекта информацией, которая актуальна для мозга в рамках его задач (его целей). При этом, если естественный мозг работает только с теми внешними данными, к которым он адаптирован эволюционно (модальности восприятия и интенсивность сигнала, его сектор), искусственный интеллект, используя тот же принцип, может работать с самой разнообразной информацией.

2. Принцип выявления отношения

Все психические процессы, включая обработку информации, являются замкнутыми внутри нейронного комплекса мозга, а потому оценка возникающей в мозге информации (ее усложнение) осуществляется исключительно через акт соотнесения одной информации с другой, а сам мозг реагирует не на объект реальности как таковой, а на то, как он соотносится с другой информацией, находящейся в мозге.

В психологии этот принцип изначально получил название — *принцип гештальта* (или *отношение фигуры и фона*). Мозг, как мы знаем, реагирует не на конкретный стимул, а на то, каким становится этот стимул при соотнесении его с той информацией, которая в мозге уже содержится (часть соответствующих данных, как уже было сказано, являются генетически предопределенными, другая — накопленным опытом). Так, например, нейрон реагирует на импульсы от других нейронов в зависимости от того, в каком состоянии он

находится, что определено той информацией, которую он своим состоянием представляет. Ровно так же этот механизм работает и на самых высоких уровнях организации, когда реакция человека на один и тот же стимул зависит от того, в какой среде он сейчас находится.

Иными словами, для того, чтобы определить объект (что он будет значить), мозгу необходимо определить фон, на котором он этот объект рассматривает. Фон же, в свою очередь, определяется задачей, которую мозг решает (его целями). При этом, сам объект является, в свою очередь, фоном для своего фона. Например, предположим, что у нас есть объект «антибиотик» — для ребенка из страны третьего мира с неразвитой медициной и дефицитом лекарств его значение будет сильно отличаться от значения для ведущего врача западной клиники — это и есть принцип отношения: значение для мозга определяется через отношение объекта с другими объектами, которые в нем есть.

3. Принцип аппроксимации до сущности

Мозг человека не создает модель реальности всякий раз заново, он пользуется той моделью, которая формировалась и редактировалась на протяжении всей его жизни. То есть, по сути, при столкновении с новой информацией он всегда основывается на своем прежнем опыте, на уже сделанных им выводах. Собственно принцип аппроксимации до сущности заключается в том, что мозг как бы уже всегда знает, с чем он может встретиться (часть соответствующих знаний, как уже было сказано, являются генетически предопределенными, другая — накопленным опытом), и формирует новое знание только в том случае, если оно противоречит существующим моделям и не схватывается ими.

Однако, в реальности не существует абсолютно идентичных объектов, поэтому мозг осуществляет аппроксимацию, то есть игнорирует отличия, если ему удается по специфическим признакам присвоить объекту ту или иную «сущность».

При этом, под «сущностью» понимается функционал объекта — то, какое значение он имеет для мозга (какую роль он выполняет) в рамках решаемых им задач (его целей).

Наглядным примером в этом случае является использование какого-либо объекта в качестве другого, путем наделения первого функционалом второго под актуализированную потребность: когда человек устал и хочет отдохнуть — в лесу пень может служить стулом, так как на нем можно сидеть; в то же время человек хочет выжить — тогда услышанный рык и шуршание кустов могут быть сапроксимированы до дикого зверя, от которого нужно бежать, так как он может быть опасен для жизни.

4. Принцип локальности-распределенности

Вся информация, поступающая в мозг, может в нем многократно дублироваться, и ее копии обрабатываются параллельно разными структурами самостоятельно, и лишь затем эта информация интегрируется в целостный образ. Иными словами, мозг обрабатывает одну и ту же информацию разными способами (в разных отделах), чтобы получить несколько результатов и объединить их в рамках одного, целостного интеллектуального объекта, в соответствии с определенной им сущностью (последняя может определяться генетически, исходным кодом, или накопленным опытом). С другой стороны, наделяя соответствующий интеллектуальный объект сущностью (аппроксимация до сущности), из чего определяются его дополнительные признаки, соответствующей этой сущности, он способен объединять в один интеллектуальный объект (модель) информацию, которая поступает к нему по разным каналам.

Этот же принцип реализуется мозгом на клеточном уровне, на уровне кортикальных колонок, а также отдельных функциональных областей: информация обрабатывается параллельно множеством структур, производящих в конечном итоге

цельный единичный объект (или решение), соответствующий задачам мозга (его целям).

5. Принцип «тяжести»

Количество нейронных связей, включенных в создание модели объекта, количество отношений между элементами континуума интеллектуальных объектов, объем привносимой в объект информации (атрибуты сущности), количество способов расчета информации об объекте и объединение разноканальной (модальности) информации о нем в единое целое, соотнесенные с актуальностью задачи (цели) системы, определяют «тяжесть» интеллектуального объекта. «Тяжесть» интеллектуального объекта предопределяет решение системы.

Так, например, если человек голоден — он будет искать пищу, которая утолит голод, однако если ему начнет угрожать непосредственная опасность (например от хищника), то начнет главенствовать оборонительная стратегия, и он перестанет искать еду и начнет спасаться, так как без еды он проживет еще какое-то время, а если его настигнет хищник — он умрет сразу. То есть, приоритет отдается наиболее актуальной и выраженной в каждой конкретной ситуации стратегии.

Вывод

Математизация BPP в языке теории категорий, а также тесная связь с вероятностными моделями, глубоким обучением и когнитивными архитектурами, позволяет реализовать систему принципов работы головного мозга в технических средах различными, активно развиваемыми AI-сообществом способами. Таким образом, системный анализ и проработка общих принципов работы мозга, выделенных в ходе нейронаучных исследований, могут привести к более глубокому пониманию интеллекта, которое будет способствовать созданию AGI..

Заключение

Возникнет ли общий ИИ как разум в теле конкретного робота? Такой сценарий нельзя исключить. Но приверженцы экосистемного подхода скажут, что опыт одного робота слишком ограничен, и процесс накопления опыта можно сделать более эффективным и реализовать AGI как платформу, интегрирующую информацию от многих роботов в единой базе знаний или картине мира. Да и зачем ограничиваться лишь роботами? Есть огромный интернет вещей, умные дома и города. А почему бы тогда не пойти еще дальше и не решить, что общий ИИ вероятнее всего реализуется в форме платформы, предназначенной для решения произвольных задач?

Возможно ли это? Не исключено. Но как же воплощенность? Сможет ли такая платформа сформировать какую-то интегрированную картину мира, привязанную через сенсорику и действия к реальности и используемую и пополняемую при решении разных задач? Вовсе не во всех сценариях ее развития. И может оказаться, что без этого полноценный общий ИИ не создать.

Возникнет ли общий ИИ как агент, обучающийся с подкреплением, в результате искусственной эволюции или через загрузку человеческого сознания? Пока нельзя исключить ни один из вариантов, хотя каждый из них по отдельности вызывает некоторые сомнения. Возможно, общий ИИ воплотится в какой-то гибридной или вообще пока непредставимой форме. При этом, хотя обсуждение вопроса о том, какой внешний облик может принять общий ИИ, не дает непосредственных подсказок о его возможном внутреннем устройстве (например, о том, как на уровне алгоритмов должно реализовываться трансферное обучение), каждый из рассмотренных вариантов воплощения подчеркивает те или иные важные проблемы, которые необходимо решать для продвижения к общему ИИ.

ПОСЛЕСЛОВИЕ. **БУДУЩЕЕ AGI**

Как изменится мир с появлением AGI?

В настоящее время область искусственного интеллекта привлекает немалое внимание со стороны общества, бизнеса и властей. При этом область общего искусственного интеллекта зачастую остается на периферии внимания, хотя именно она отражает научно-популярный и медийный образ ИИ как системы, способной к достижению и преодолению человеческого уровня познания и решения интеллектуальных задач. Но на практике этот образ применяется как рекламный плакат для привлечения финансирования с целью форсированного внедрения систем ИИ частного назначения, зачастую эффективных, но все еще не способных решать широкий круг задач. И можно понять, почему это происходит. Во-первых, предполагается (скорее всего, ошибочно), что процесс внедрения систем узкого ИИ двигает нас в направлении общего интеллекта и в перспективе приведет к появлению AGI эволюционным путем без специальных усилий. А во-вторых, области приме-

нения общего искусственного интеллекта довольно трудно очертить в силу его сложности и универсальности: придумать и построить специализированные системы намного легче.

В результате на практике оказывается проще проанализировать политические и социальные эффекты создания и внедрения AGI, чем экономические возможности, которые он способен принести. Чтобы выйти из этого тупика, важно разделить области ИИ, AGI и Narrow AGI.

Искусственный интеллект

ИИ — экстенсивный экосистемный подход к узкому искусственному интеллекту «снизу», ориентированный на прикладную науку, инженерные разработки и максимальное практическое внедрение сегодня и завтра. Чтобы развивать эту область, стоит создавать консорциумы, поощрять сотрудничество научно-исследовательских групп с индустриальными партнерами, обеспечивать поддержку со стороны профильных министерств и ведомств и широкое финансирование через институты развития. Также важно внедрять активную грантовую политику, поддерживать стартапы, стимулировать научные публикации и создавать хабы больших данных.

Польза для государства: построение экосистемы, вовлечение в перспективную область талантливой молодежи, поддержание технологических компетенций, превентивное импортозамещение, потенциал экспорта, повышение производительности труда.

Польза для бизнеса: лидерские позиции в отрасли, повышение эффективности, кадры, новые рыночные ниши.

AGI

AGI — универсалистский подрывной подход к искусственному интеллекту «сверху», ориентированный на решение задачи рекурсивного самообучения, моделирования мира и обобщения знаний и навыков из разных сфер и ситуаций.

Для его создания и развития важно создавать фреймворки и комплексы баз данных и алгоритмов, пригодных для работы с универсальным представлением знаний, проведения операций с цепочками причинно-следственных связей и совершения логического вывода над представлениями. Этот подход должен быть нацелен на достижение стратегического результата — получение функциональной системы, способной производить рассуждения над экосистемой узких ИИ. Тут хороший эффект может оказать точечное проектное финансирование исследований и разработок с отсутствием требований к обязательному практическому применению в ближайшем будущем. При этом стоит избегать концентрации на узких предметных областях и частных приложениях и стремиться к универсализации.

Польза для государства: научный прорыв, парирование информационных угроз, возможность проведения экспансивной политики на рынке высоких технологий в гражданском секторе экономики, возвращение к практике экспорта образовательных и управленческих систем под ключ.

Польза для бизнеса: лидерские позиции в отрасли, способность к энергичному рыночному маневру, интересные возможности для сотрудничества с государством.

Польза для науки и общества:

- эволюция гуманитарного знания под воздействием машинного критического мышления;
- прорыв в технических науках, освоение инструментов автоматического доказательства теорем, автоматического программирования, автоматизации научных исследований в целом.

Narrow AGI

Narrow AGI (NAGI) — интенсивный проектный подход к искусственному интеллекту «по горизонтали», ориентированный

на отработку ключевых технологий AGI на определенной широкой предметной области с достижением значительной степени обобщения знания в ней. Конечно, это компромиссный подход, но на практике он позволит снизить остроту конфликта между стратегическим подходом к технологии и полезной применимостью системы на тактическом горизонте. При этом действительно значимые практические результаты можно получить уже в обозримом будущем. Однако построение системы NAGI в пределах определенной предметной области существенно более затратно, чем внедрение узких инструментальных и, как правило, слабых моделей. С одной стороны, оно требует глубоких исследований и разработок, а с другой — создает условия для ограничения универсализации. Поэтому следует тщательно выбирать области применения, несущие в себе наибольший потенциал для реализации систем NAGI.

Ниже приведены сферы, где Narrow AGI может принести максимальную пользу.

Медицина и здравоохранение. Медицина — одна из самых древних и хорошо структурированных областей знаний, что открывает перспективные возможности для внедрения системы Narrow AGI. Кроме того, инвестиции в эту сферу оправданны как с экономической, так и с альтруистической точки зрения.

Машинное зрение и анализ временных рядов[89] требуются для качественной диагностики; обработка естественного языка необходима для работы с историей болезни и корпусом медицинской литературы; речевые интерфейсы могут применяться для ускорения ежедневной рутинной работы врачей; системы точного управления манипуляциями в 3D-пространстве требуются для развития робототехнической хирургии; работа с молекулярной структурой вещества способна приводить

---

[89] Временной ряд — это последовательность значений некоторой переменной (или переменных), регистрируемых через определенные промежутки времени. Например, показания кардиограммы.

к открытию новых полезных препаратов; комплексное моделирование химических процессов может ускорить клинические исследования; анализ кода ДНК поможет создать совершенно новые возможности активного лечения и диагностики.

Здравоохранение — одна из главных базовых потребностей человека, так что предоставление медицинского страхования и медицинского AGI как услуги может оказаться одним из ключевых факторов удержания потребителя в своей экосистеме, поддержать и развить позитивный образ компании.

Умные города. Технологии управления большими системами в реальном времени — еще одно перспективное направление для внедрения систем AGI. В качестве самого очевидного применения здесь выступают системы контроля и вычисления «удивительных» паттернов (аномалий) — странных событий, которые требуют внимания и, возможно, вмешательства (например, преступлений, аварий и т.д.). Но практических эффектов гораздо больше. Так, можно говорить не только об умных городах, но и умной инфраструктуре в целом, о внедрении полезных практик в градостроительство, логистику ресурсов, управление финансами. Перспективы для государства огромны и способны привести к значительному упорядочиванию хозяйственной инфраструктуры и к эволюции управленческих практик. При этом степень контроля над гражданской и частной жизнью может быть предметом активного и осознанного регулирования, но важно понимать, что такие системы не обязательно сводятся к тотальному контролю — вопрос в том, какие цели ставит перед собой общество, применяющее такие системы.

Наука и общество. Эта область включает три крупных домена: наука и образование; законотворчество и судебная практика; общественный дискурс. Активное развитие социального Narrow AGI в таком триедином формате позволит государству заметно поднять уровень информационной безопасности в обществе параллельно с получением пользы в юридической

и научной областях. Так, область юриспруденции — оптимальный полигон для упорядочивания, поиска и устранения логически конфликтующих друг с другом положений. Важную роль Narrow AGI может сыграть и в сфере образования — он может сделать процесс обучения более эффективным и вовлекающим, разрабатывать более адаптивные и персонализированные образовательные программы, более точно оценивать результаты учебы. Научная же сфера по праву может считаться областью применения Narrow AGI с самой высокой потенциальной комплексной отдачей — особенно с прикладным применением к исследованиям и разработкам в области искусственного интеллекта и инженерного программирования.

Но не только это. Очевидно, что многовековой научный багаж, накопленный человечеством, требует интеграции и переосмысления, для которого собственно человеческих ресурсов уже недостаточно.

> **Разумное внедрение приложений искусственного интеллекта в область образования приведет к формированию новых практик обучения человека.**

И поскольку в том или ином виде оно уже неизбежно, очень важно разрабатывать подобные системы в опережающем режиме.

Автоматизированная торговля. Сейчас активно обсуждается повсеместное внедрение ИИ в системы индустриального производства. Эта область давно освоена искусственным интеллектом. Так, старейшей из действующих индустриальных систем является ИИ-система компании Siemens, управляющая циклами работы и обслуживания энергетических турбин этого производителя. ИИ-подобные алгоритмы активно применяются и для автоматизации биржевой торговли.

При этом наибольший комплексный экономический эффект, скорее всего, может быть достигнут применением системы

Narrow AGI в области трансграничной торговли товарами и услугами. Широчайшая номенклатура товарных рынков и сложность предсказательного прогнозирования спроса делают эту задачу отличным испытательным полигоном для отработки технологий AGI, а ее успешное решение может послужить мощным драйвером развития для государства и обеспечить дополнительный экономический рост. Для корпораций же открывается возможность занять лидерские позиции в экосистеме автоматизированной торговли, в том числе в секторе кредитования сделок.

Робототехника. Робототехника все еще остается молодой индустрией, но создает все больше перспективных рынков с огромным спросом. Пока большая часть общей стоимости промышленных робототехнических решений приходится на точную механику и электротехнику. Но развитие систем ИИ и когнитивных архитектур предсказуемо приведет к появлению массового рынка сервисных роботов, роботов-игрушек, домашних систем, управляющих всеми устройствами. Так что робототехнику стоит признать одним из наиболее удачных полигонов для отработки технологий AGI.

Не станет ли это потрясением для человечества?

В полной мере оценить эффект от внедрения систем на основе AGI в образование, экономику, управление и многие сферы общественной жизни непросто даже на уровне практического освоения приложений Narrow AGI. Тем более сложно оценить комплексные и синергетические эффекты от появления полноценных систем общего искусственного интеллекта. Эти эффекты могут сильно изменить привычные для нас экономические, цивилизационные и общественные ландшафты.

По наиболее драматичному сценарию, AGI будет размывать сложившиеся порядки и институты быстрее, чем мы сможем к этому адаптироваться. А согласно самым консервативным сценариям, нас ждут фундаментальные перемены, новые вызовы и возможности.

> **«Спокойные» варианты развития событий предполагают, что человечество сможет объединиться, чтобы осознанно построить новое общество и выработать общую, межгосударственную стратегию регулирования искусственного интеллекта.**

К сожалению, в нынешних условиях такой путь кажется просто недостижимым. Куда вероятнее сценарий обострения конкурентной борьбы за ресурсы, зоны контроля и умы людей. При этом общества, претендующие на политический суверенитет и долю мирового рынка, должны будут выигрывать в гонке за адаптацию, разрабатывать жизнеспособную управленческую модель, транслировать позитивный образ будущего. А для этого им неизбежно понадобятся AGI-технологии. Соответственно, технологическое и научное лидерство в области AGI или как минимум технологический паритет с другими государствами/сообществами стоит рассматривать как насущную необходимость, ключевой компонент борьбы за выживание, безопасность и процветание. При этом важно соблюдать этические принципы при разработке общего искусственного интеллекта, чтобы избежать возможных катастроф и сделать технологии ИИ максимально безопасными, полезными и доступными для всех людей на планете. Мы предлагаем разработчикам взять за основу следующие положения:

- **ИИ как общее благо.** Стоит стремиться к тому, чтобы технологии ИИ использовались во благо как можно большего числа людей и приносили пользу для развития человечества.

- **Безопасность ИИ.** Важно внимательно относиться к опасениям и рискам, связанным с безопасностью технологий ИИ, чтобы разработка и применение технологий ИИ были безопасными и контролируемыми в максимальной возможной степени.
- **Ответственное отношение к ИИ.** Надо ответственно относиться ко всем рискам, которые возникают при применении технологий ИИ, а также к социальным последствиям их широкого применения.
- **Применение ИИ в интересах граждан.** Технологии ИИ должны применяться прежде всего для пользы и благополучия людей, внедрение ИИ никогда не является самоцелью.
- **Справедливость.** Технологии ИИ должны использоваться справедливо, на равных для всех условиях и без предвзятости к каким-либо группам населения
- **Обеспечение конфиденциальности.** Технологии ИИ должны применяться с соблюдением требований конфиденциальности и с уважением к частной жизни человека, а также к коммерческой тайне. Они не используются для незаконного сбора, хранения, обработки или использования персональной информации.
- **Уважительное отношение к мнению пользователей и общества.** При внедрении ИИ стоит опираться на пожелания пользователей и прислушиваться к их мнению, а также поощрять и организовывать широкое общественное обсуждение этических вопросов применения ИИ.
- **Соблюдение закона.** Использование технологий ИИ должно всегда осуществляться в полном соответствии с применимым законодательством и гарантировать обеспечение всех прав и свобод человека при применении ИИ.

Важно учитывать, что все эти принципы могут обновляться при необходимости, с учетом высокой скорости развития технологий.

Стратегии разработки AGI

В прошлых главах мы уже обсудили разные технические подходы к достижению общего искусственного интеллекта. Но успех во многом будет зависеть от правильно выстроенной инфраструктуры, менеджмента, модели финансирования и налаженной коммуникации между специалистами разных направлений. В экспертных кругах часто можно встретить два во многом противоборствующих подхода к проблематике разработки AGI и управлению ресурсами в гонке за лидерство в данной области.

Принцип вертикальной интеграции

Первый подход предполагает решающую ключевую роль фундаментальной науки и междисциплинарных исследований и, соответственно, широкое финансирование исследований с минимальной привязкой к метрикам достижения измеримого результата. Сильная сторона такого подхода — возможность вовлечения государственных научных институций (зачастую находящихся в стагнации) в активную работу вокруг актуальной и амбициозной задачи, соизмеримой с самыми масштабными проектами прошлого — космическим и атомным.

> **Достижение общего искусственного интеллекта как миссия — хорошая мотивация для выделения финансирования и успешного перезапуска экосистемы фундаментальной науки как таковой.**

Правда, существует риск того, что вложение усилий и ресурсов в разнонаправленные поиски может привести

к недофинансированию тех точек роста, от работы с которыми может быть достигнут максимальный эффект.

Но, во-первых, никто точно не знает, где именно находятся такие точки роста, а во-вторых, ключевую роль в разработке систем AGI в России призвана сыграть исторически сильная математическая школа, которой удалось выжить в первую очередь благодаря подготовке массово востребованных программистов и разработчиков информационных систем. Разумеется, мы можем лучше использовать эффективную научную традицию как наше конкурентное преимущество.

Кроме того, распределение ресурсов на междисциплинарные исследования (например, связанные с проектированием когнитивных архитектур и психологией, методологией машинного мышления и философией, машинным моделированием мира и педагогикой) может привести к конструктивным идеям, неожиданным открытиям и большей готовности социальных наук к ожидающему нас технологическому прогрессу.

Принцип горизонтальной интеграции

Второй подход в большей степени предполагает акцент на синтезе актуальных достижений прикладной науки и создании широких платформенных экосистем внедрения и разработки ИИ с опорой на лидирующие компании. Компании в этой системе станут источниками финансирования и софинансирования, поставщиками и хабами больших данных, пилотными площадками внедрения систем и апробации моделей. В рамках этого подхода предполагается проектное, грантовое и венчурное финансирование исследований и разработок с привлечением различных предприятий-подрядчиков. Особое внимание лучше уделить поддержке технологических стартапов и малых исследовательских команд для включения их в общую платформенную экосистему с компаниями-лидерами, а также средними предприятиями.

В целом горизонтальный подход более реалистичен и практичен с точки зрения достижения конкретного результата — создания действующих систем ИИ, кооперации разрабатывающих и производящих их предприятий, формирования внутреннего потребительского спроса на ИИ.

Но непонятно, насколько он будет работоспособен без специальных дополнительных мер для создания AGI. Самая слабая сторона этого подхода — ставка на генерацию идей малыми предприятиями и стартапами, которые при этом будут вынуждены работать в основном в условиях ограниченного локального рынка. Кроме того, чисто рыночный подход к измерению успешности, когда работа отдельных предприятий оценивается через их вклад в производство, будет размывать линию движения к AGI, потому что тогда становится проще продемонстрировать свою успешность через «узкие» решения.

При этом горизонтальный подход опирается на признанную перспективной общую идею децентрализации разработок, которая особенно хорошо применима к изобретательской инициативе. В случае AGI вполне может оказаться, что изобретательская инициатива приведет к выработке действительно амбициозных и многообещающих и при этом вполне практических методов, подходов и техник, которые станут ключевыми элементами сквозной технологии, пронизывающей многие рынки и области применения. Также в защиту подхода стоит отметить, что заявляемая на уровне платформенной экосистемы децентрализация все же позволяет точечно сфокусировать ресурсы и усилия на небольшом пуле ключевых проектов. Такие ключевые проекты, став точками концентрации специфических технологий и компетенций, в дальнейшем смогут эффективно транслировать их на всю платформенную экосистему, стимулируя развитие прикладных технологий и внедрение продуктовых решений.

Принцип комплексной интеграции

Но есть и третье решение, представляющее собой практический вариант синтеза вышеописанных подходов, который учитывает специфику проблематики исследований, разработок и внедрения AGI в контексте глобальной гонки за лидерство. Прямое противопоставление подходов, ориентированных на фундаментальную науку и рыночную экосистему, в случае AGI неконструктивно, поскольку и «непрактичные» с точки зрения быстрого продуктового решения академические исследования, и инженерные разработки одинаково важны.

Кроме того, академический подход ассоциируется с условной «советской» моделью, а ориентация на построение рыночной экосистемы — с условной «западной». Но на практике принять и использовать любую из этих моделей в чистом виде для России сейчас невозможно. Советская модель требует гораздо большей способности к выстраиванию иерархических структур, концентрации ресурсов, отлаженных механизмов финансирования и практического воплощения научных достижений. А западная модель требует наличия компаний-локомотивов, работающих на глобальных рынках и в совокупности способных фактически содержать как академическую науку, так и большую часть венчурной экосистемы, уделяя при этом достаточно внимания собственным подразделениям исследования и разработки. Примечательно, что западная модель функционирует подобным же образом в целом пуле незападных экономик, которые могут составить серьезную конкуренцию в гонке за достижение AGI, например, в Китае, Южной Корее, Японии. В этих экономиках также либо имеются свои отраслевые лидеры и вместительный собственный рынок (как в Китае), либо они напрямую замкнуты на западные рынки.

Комплексный метод призван объединить для работы над общей задачей в единой экосистеме исследования, разработки и внедрения разные коллективы и организации:

1) академические группы, ведущие проработку перспективных направлений исследования, результаты работ по которым должны выражаться в пригодных к практическому инженерному применению математических аппаратах и академическом программном коде;
2) инженерные команды, которые ведут разработку основанных на результатах работы первой группы новых опорных технологических платформ и фреймворков, аппаратных вычислительных комплексов, а также производят взаимную адаптацию и интеграцию их критических компонентов, проводят масштабные эксперименты;
3) различные предприятия (от компаний-лидеров до стартапов), которые разрабатывают рыночные продукты, а также домен-ориентированные платформы и продукт-ориентированные технологии, и внедряют прикладные методы и техники.

Выстраивание кооперационных связей между участниками экосистемы будет производиться с учетом различия этих трех групп и с сохранением возможности прямого программного (типы 1, 2), грантового (типы 1, 2, 3) или венчурного (тип 2, 3) финансирования любой из них для форсированного развития AGI. При этом каждая из структурированных таким образом групп может (а в идеале должна) иметь свой базовый постоянный поток и источник финансирования, отдельный от целевых фондов форсированного развития AGI. Это нужно для того, чтобы эти группы могли выступать с инициативными предложениями от себя и привлекать гранты в форме софинансирования, что может быть наиболее эффективно в ряде случаев.

Стоит подчеркнуть, что принцип софинансирования не стоит делать обязательным для всех случаев. Так, для академической среды в качестве основного источника может действовать программа финансирования и постановки

задач, регулируемая Министерством науки и высшего образования; для инженерных интеграторов — программа адресного финансирования, создание частно-государственных партнерств, крупные якорные вложения в совместные предприятия от компаний-лидеров; для рыночных предприятий и стартапов — венчурное финансирование со стороны специализированных государственных институций и частных инвесторов, больших и малых.

> **Любая стратегия развития AGI и самые интенсивные и хорошо организованные управленческие усилия неизбежно столкнутся с ключевым вопросом о критериях отнесения того или иного проекта или исследования к области общего искусственного интеллекта.**

Для этого нужно понимать структуру данной области, составить единую и непротиворечивую теоретическую базу и выявить наиболее перспективные подходы к разработке AGI. Этому и посвящена данная книга. Это первый шаг на пути выработки общего бэкграунда на русском языке для всех заинтересованных в AGI. Хочется надеяться, что она подтолкнет исследователей, инженеров, представителей бизнеса и государства к эффективному сотрудничеству. Сейчас над общим ИИ работают разрозненные группы специалистов, и не хватает инициатив, которые помогли бы этим группам объединять свои знания и навыки и координировать работу над совместными проектами. Кроме того, в этой области не хватает прямого финансирования. Разумные инвестиции в подобные разработки и создание общего пространства для эффективного сотрудничества стали бы важным шагом для России на пути к технологическому лидерству. Ведь искусственный интеллект, как говорил наш президент, — это будущее не только России, но и всего человечества.

На подступах к сверхразуму

СИЛЬНЫЙ ИСКУССТВЕННЫЙ ИНТЕЛЛЕКТ

Руководитель проекта *А. Марченкова*
Дизайнер *А. Маркович*
Дизайн обложки: *А. Бурсаков*
Иллюстрации: *А. Смирнов*
Корректоры *Ю. Семенова, А. Смышляева*
Компьютерная верстка *Б. Руссо*

Подписано в печать 26.11.2020. Формат 60 × 90 $^1/_{16}$.
Бумага офсетная № 1. Печать офсетная.
Объем 15 печ. л. Тираж 4000 Заказ А-3390.

Отпечатано в типографии филиала АО «ТАТМЕДИА» «ПИК «Идел-Пресс».
420066, Россия, г. Казань, ул. Декабристов, 2.
e-mail: idelpress@mail.ru

ООО «Интеллектуальная Литература»
123007, г. Москва, ул. 4-я Магистральная, д. 5, стр. 1,
Тел. +7 (495) 980-53-54
e-mail: info@intlit.ru

Знак информационной продукции
(Федеральный закон №436-ФЗ от 29.12.2010 г.)

12+

Для заметок

Для заметок

Эта книга — первый кросс-дисциплинарный гид по искусственному интеллекту на русском языке. **Сильный искусственный интеллект** — это следующая ступень в развитии ИИ, не обязательно наделенного самосознанием, но, в отличие от современных нейросетей, способного справляться с широким кругом задач в разных условиях. Авторы книги рассказывают о том, что должен уметь сильный ИИ, какие научные подходы помогут его создать и как изменится мир с его появлением.

#СильныйИИ
9 785907 394186

Знания, которые меняют жизнь

альпина паблишер

заказ книг +7 (495) 120-07-04
и на сайте www.alpina.ru
www.facebook.com/alpinabook

приложение Альпина.Книги в App Store и Google Play

ideabooks alpinabook alpinabook alpinaru