

1. НАУЧНЫЕ СТАТЬИ

1.1. ПУТИ ПРЕОДОЛЕНИЯ НАДВИГАЮЩЕГОСЯ КРИЗИСА РАЗВИТИЯ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА

Малинецкий Г.Г.¹, Войцехович В.Э.², Смолин В.С.¹

¹ФИЦ ИПМ им. М.В. Келдыша РАН, Москва, Россия

²Тверской Государственный Университет, Тверь, Россия

Увеличение за последнее десятилетие числа настраиваемых параметров в современных «нейросетевых» вычислительных схемах на 8-10 десятичных порядков до сотен миллиардов и десятков триллионов позволило достичь выдающихся результатов при решении широкого класса «интеллектуальных» задач. Увеличение сложности управляющих устройств, согласно теореме Эшби о необходимом разнообразии систем управления, потенциально требуется для решения более трудных задач. Если бы стоимость, время обучения, объёмы исходных данных и даже количество потребляемой электроэнергии «нейросетей» росли пропорционально росту успехов «нейросетевых» вычислений, то уже давно бы были достигнуты пределы возможностей развития. Использование всё более широкого спектра математических идей, оптимизация алгоритмов, автоматизация сбора информации и совершенствование электроники снизили скорость роста затрат. По мере приближения к пределам разумных расходов на решение «интеллектуальных» задач, акцент развития всё больше переносится с аппаратных средств реализации алгоритма обратного распространения ошибки (BPE, backpropagation error) на другие математические подходы. Успех в реализации таких простых идей, как локализация памяти, декомпозиция сложных объектов и линеаризация преобразований может дать новый импульс развитию искусственного интеллекта (ИИ), сравнимый с внедрением идеи градиентного спуска при обучении «глубоких нейросетей», которая уже привела к революции в машинном обучении.

1. Введение

1.1. Растущее понимание проблемы

В июле 2021 года Йошуа Бенжио, Янн ЛеКун и Джеффри Хинтон (обладатели премии Тьюринга – ACM A.M. Turing Award 2018 – за прорывы, которые сделали глубокие нейронные сети критически важным компонентом вычислений) опубликовали статью «Глубокое обучение для искусственного интеллекта» [Bengio, 2021]. В статье описываются истоки и основные события, которые привели к революции нейросетей в машинном обучении, и указывается, что одной из причин успеха стал переход от символического к векторному описанию понятий. В частности, обладатели премии Тьюринга написали:

“В парадигме, основанной на логике, символ не имеет значимой внутренней структуры: его значение заключается в его отношениях с другими символами, которые могут быть представлены набором символических выражений или реляционным графом. Напротив, в парадигме, вдохновляемой мозгом, внешние символы, которые используются для коммуникации, преобразуются во внутренние векторы нейронной активности, и эти векторы имеют многомерную структуру сходства”.

В статье приводится описание ряда других причин, позволивших достичь успехов: внимание, конкурентное обучение [Chen, 2020] и вариационные подходы. При этом авторы отмечают, что это только выборочный обзор, причин намного больше. Все вместе эти достижения позволили достичь значительных успехов в решении задач восприятия, (системы 1 по Канеману) [Kahneman, 2011]. Успехи в решении задач системы 2 по Канеману (мышления) менее значительны, хотя тоже есть [Silver, 2017].

Глядя в будущее, Йошуа Бенжио, Янн ЛеКун и Джеффри Хинтон выделили ряд проблем, которые необходимо решить для достижения существенного ускорения прогресса в области AI. Они пишут: «Существуют фундаментальные недостатки текущего глубокого обучения, которые нельзя преодолеть одним лишь масштабированием», нужны новые идеи и подходы.

Также авторы статьи [Bengio, 2021] отмечают, что люди способны к более быстрому обучению и устойчивому поведению в изменяющихся условиях, чем современные алгоритмы глубокого обучения. При этом они указывают и пути приближения к людским возможностям: иерархическая обработка, совершенствование механизмов внимания, использование нескольких шкал времени, развитие способностей к решению незнакомых задач, выявление причинно-следственных связей и, главное, осуществление взаимодействия высокоуровневых концепций с низкоуровневым восприятием и действиями (grounding).

Выводом статьи [Bengio, 2021] является утверждение, что выделение понятий, которыми должен обладать искусственный интеллект (AI), важнее, чем проведение логических операций. Авторы считают, что для развития данного свойства необходимы новые идеи и подходы.

1.2. Приближение к пределам экстенсивного развития

Последние десятилетия происходит неуклонный рост количества вырабатываемой электроэнергии, производительности компьютеров, объёмов собираемой и обрабатываемой информации, величины капиталовложений и многих других количественных параметров развития цивилизации. Раздаются голоса, что мы недалеко от достижения, а где-то уже и превысили пределы возможностей нашей планеты в плане обеспечения всех этих благ без вредных последствий для грядущих поколений. Но нет особых сомнений, что экономический рост будет продолжен: будет развиваться наука, продолжится освоение космического пространства, будут эффективнее использоваться земли и моря и, наконец, появится сильный искусственный интеллект (СИИ), который во всём этом поможет. Можно спорить, замедлится или ускорится экономический рост, но общая тенденция к нему, скорее всего, сохранится. И, если нас и ожидают периоды регресса, то они, скорее, будут связаны с конфликтами и катастрофами, чем с ограниченностью ресурсов.

До последнего времени развитие ИИ не определялось ресурсными проблемами: хватало и денег, и энергии, и времени. А не хватало понимания сути решаемых задач, эффективных алгоритмов, высокопроизводительной вычислительной техники и больших объёмов информации для решения «интеллектуальных» задач. Сейчас можно констатировать рост понимания даже в высших эшелонах власти стратегической значимости возможностей построения СИИ.

Скорость развития возможностей нейросетей значительно обгоняла темпы прогресса цивилизации во многих отраслях. Деньги, энергия, объёмы данных, вычислительные мощности, конечно, были сдерживающими факторами развития, но организации, государственные и корпоративные, выделявшие эти ресурсы, были далеки от пределов своих возможностей. Уровень обеспечения выполняемых работ по ИИ определялся степенью понимания их значимости и регулярно повышался за последнее десятилетие.

Но, как бы ни была высока степень понимания проблемы, никто не сможет обеспечить выделение ресурсов выше имеющихся возможностей. При достижении их текущего предела развитие будет продолжено, но темпы будут определяться ростом цивилизационных ресурсов.

1.3. Новой «зимы» ИИ не будет!

Хотя на горизонте маячат серьёзные проблемы, само понимание их наличия открывает путь к их решению. Вкладываются большие силы, есть высокая вероятность успеха, но нет гарантий, что проблемы будут решены. Вместе с тем есть уверенность, что целиком преодолеть «проклятие размерности», «комбинаторный взрыв» и научиться решать N -полные задачи при $N \gg 1$ ($N > 1000$) не удастся. Надежда есть только на открытие путей смягчения данных проблем. Причём, ниже будет говориться о том, что смягчения можно достичь за счёт декомпозиции сложных задач на компоненты, для которых $N \sim 1$ ($N < 20$).

Но, даже если развитие новых подходов к ИИ будет идти медленно, «зимы» ИИ уже не случится никогда. Да, новые идеи с трудом находят понимание у старшего поколения учёных, которые сделали себе карьеру на других идеях и заинтересованы именно в их развитии. Корпорации, вложившие деньги в технологии и производство систем ИИ, будут стремиться сдержать развитие новых идей, чтобы успеть окупить свои расходы. Но производство и совершенствование «интеллектуальных» устройств и технологий не прекратится.

Так происходило со всеми базовыми технологиями. Внедрение тепловых машин, электричества после стадии бурного роста переходило в фазу плавного развития, но не отказа от технологий или снижения интереса к ним. Нынешняя «весна» ИИ отличается от предыдущих тем, что тогда были только большие надежды, а теперь их удалось реализовать в виде отдельной промышленности ИИ со своими массовыми производствами, технологиями, научными центрами, учебными программами и пр. [Кай-Фу Ли, 2019]. Никто не будет резать курицу, несущую золотые яйца.

Но так же, как переход от паровых машин к двигателям внутреннего сгорания дал новый толчок развитию и расширению применения тепловых машин, новые идеи могут решить ряд проблем в разработке современных подходов к реализации ИИ и привлечь значительно большие ресурсы за счёт заметного расширения круга решаемых задач и повышения эффективности их анализа. Это позволит не снижать, а наоборот, увеличивать темпы создания «интеллектуальных» устройств и подойти к созданию СИИ.

1.4. Кризис понимания

Кризис роста сложности решаемых ИИ «интеллектуальных» задач является мягким, но при этом он не может быть преодолен в полном объёме – возможности систем ИИ, как и любых других устройств, всегда будут ограниченными, какими бы большими они не стали.

Кризис понимания носит противоположный характер: машины давно решают более сложные задачи, чем человек, но понимания решаемых задач они не достигли ни в какой мере. Триллионы обуча-

емых параметров не дают «эмерджентного» рождения понимания. И ряд философов активно обсуждают эту тему, доказывая, что «разрыв между активностью нейронов и сознанием» не просто существует, а не может быть преодолен.

Но, как и с апориями Зенона (и не только), всякие парадоксы основаны на отсутствии адекватных моделей процессов. Есть надежда, что предлагаемые ниже математические подходы позволят полностью устранить философский «разрыв» путём создания более адекватных моделей процессов познания окружающего мира и построения поведения в нём.

1.5. Что может дать математика для преодоления кризисов ИИ?

Кажется, что в развитии «нейросетевых» вычислений используется почти весь спектр возможностей современной математики. Но это так лишь в том смысле, что все средства направлены на совершенствование использования идеи градиентного спуска, которая является центральной для работы «глубоких» нейросетей (DNN, deep neural nets).

Идея градиентного спуска лежит в основе глубокого обучения и даёт прекрасные результаты. Не менее очевидной является идея локализации результатов обучения, которая может быть реализована механизмом конкурентного обучения при картировании. Эта идея позволяет обеспечить лучшую сохранность ранее полученных знаний при обучении новым.

Декомпозиция сложных сцен на простые объекты – не менее важный подход, который используется человеком при решении практически всех задач. Есть мнение, что именно фазовые портреты подпространств простых составляющих входных сигналов являются алгоритмическим обоснованием для выделения понятий в непрерывном мире. Дальнейшее продвижение в сторону более компактного описания сложного мира может быть осуществлено не только за счёт декомпозиции, но и за счёт линеаризации представлений путём введения в карты подпространств нелинейных осей. Таким образом

- локализация
- декомпозиция
- линеаризация

являются теми математическими идеями, которые, как и градиентный спуск, могут стать ключевыми направлениями развития ИИ.

Конечно, локальное представление данных и осуществление декомпозиции не сразу решит все проблемы. Как и с градиентным спуском, процесс перехода потребует использования и развития ряда других математических подходов.

Так, например, декомпозиция сложных задач на более простые части вызовет необходимость формирования иерархической структуры описания данных, поскольку работу выделенных отдельных частей задачи необходимо согласовывать. Возникнет ряд проблем сходимости, устойчивости и тому подобных.

И чем раньше мы начнём интенсивно решать эти проблемы, тем больше будет шансов занять передовые позиции на фронте создания СИИ.

2. Локализация

Степень локальности хранения данных

Прежде, чем выяснять степень локальности хранения данных при различных представлениях, дадим короткое определение, поясняющее, чем данные отличаются от информации:

- данные – любое описание объектов или явлений, которое всегда может быть представлено в цифровой форме;
- информация – данные, позволяющие устранить неопределённость при выборе действий.

Устранение одной неопределённости типа «да-нет» – это 1 бит информации. Данные, позволяющие сделать выбор из N вариантов, несут $\log_2 N$ бит информации. Есть более тонкие определения для вероятностных оценок, но здесь важно подчеркнуть, что наличие в данных информации определяется задачей, для решения которой эти данные используются.

Если данные применяются для построения модели объекта или процесса, который они описывают, то почти каждый бит этих данных для такой задачи несёт информацию. Для других задач информации в этих данных меньше, а для некоторых проблем информации может не содержаться совсем. Вне зависимости от того, как и где хранятся данные, они могут быть использованы для получения информации, позволяющей определиться с выбором. Но извлечение информации из данных – это отдельная задача и далеко не всегда простая. Иногда для выделения одного бита требуется обработать огромные массивы данных.

Некоторые способы извлечения информации из данных мы обсудим ниже, а рассмотрим степень локальности хранения именно данных.

Считается, что в современных компьютерах используется локальное представление данных, как при хранении в памяти, так и при обработке. Многие считают, что основное отличие нейросетевых вычислений состоит в распределённом представлении данных. Но как это утверждение соотносится с тем, что все нейросетевые вычисления производятся на стандартной вычислительной технике? Правильнее говорить про степень локальности представления данных.

Даже одно число в компьютере хранится на нескольких вентилях, причём состояние каждого вентиля описывает соответствующий разряд двоичного представления числа. Но и само это число может нести как данные, например, про яркость цветовой компоненты одного пикселя, так и пространственно-частотную характеристику участка изображения, если картинка хранится в стандарте .jpeg. Во втором случае информация о каждом пикселе изображения хранится распределённо, в виде набора пространственно-частотных характеристик участка изображения, отдельно хранящихся в нескольких ячейках памяти.

Аналогичное «распределение» информации происходит в «нейросетевых» вычислениях. Каждый формальный нейрон производит свёртку входного сигнала \vec{X} со своим вектором входных связей $\vec{W}_{k,l}$ в свою линейную активность $A^{k,l}$, которую затем нелинейно преобразует в выходную активность $O^{k,l}$ по формулам:

$$A^{k,l} = \sum_{i=0}^{N_{k-1}} w_{k,i,l} * O^{k-1,i}; O^{k,l} = \sigma(A^{k,l}); O^{k-1,0} \equiv 1, k = 1 \dots M, l = 1 \dots N_k, \quad (1)$$

где $\sigma(A^{k,l})$ – монотонное нелинейное преобразование, k – номер слоя, i, l – номера элементов в слоях, а N_{k-1} – количество элементов в слое $k - 1$ (рис. 1).

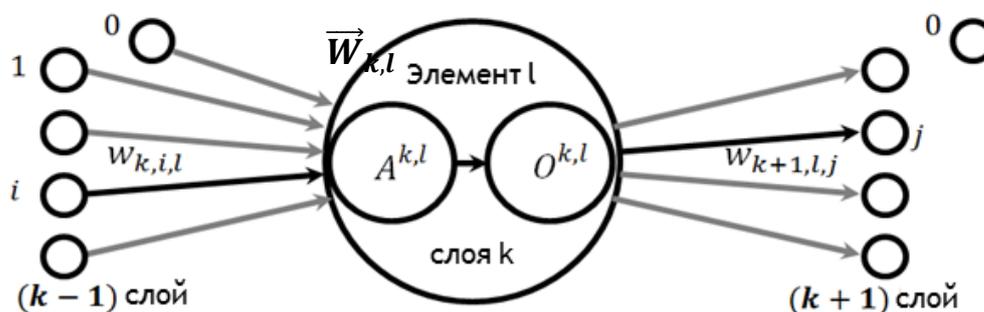


Рис. 1. Схема формального нейрона и структура индексации параметров

Свёртка входного сигнала \vec{X} с вектором входных связей $\vec{W}_{k,l}$ осуществляется в каждом формальном нейроне. Просто в глубоких скрытых слоях в качестве входного сигнала \vec{X} выступает вектор активности элементов предыдущего слоя. (Свёрточными принято называть только сети, в которых много разных элементов производят свёртку разных участков входного сигнала \vec{X} с одинаковыми векторами $\vec{W}_{k,l}$, но операция свёртки выполняется на каждом формальном нейроне в любых «нейросетевых» вычислениях).

Степень «локальности» хранения данных определяется тем, как часто выходная активность формального нейрона $O^{k,l}$ оказывается отличной от нуля. Если она заметно отличается от нуля при любых наблюдаемых значениях \vec{X} , то данный формальный нейрон хранит информацию о всём пространстве состояний \vec{X} в распределённой форме, имеющей отношение к любому \vec{X} .

Если же, как бывает в эффективно работающих нейросетях, выходная активность $O^{k,l}$ отлична от нуля только для небольшой области пространства состояний \vec{X} , то данный элемент представляет локальную информацию о данной области. И чем меньше размер области, тем выше степень локальности. Максимальная степень локальности соответствует случаю, когда некоторый формальный нейрон отвечает только на один (из пространства возможных) входной сигнал \vec{X} .

Распространено заблуждение, что распределённым образом можно хранить значительно больше данных, чем в случае локального хранения. Можно приводить много убедительных аргументов против такого мнения. Отметим только, что локальное представление данных позволяет использовать ячейки памяти с единичной эффективностью, а всякие надежды увеличить коэффициент их использования выше единицы эквивалентны попыткам создать вечный двигатель и могут основываться только на заблуждениях.

Невозможность превзойти эффективность локального представления данных не означает, конечно, бесполезности распределённых представлений и проведения операций свёртки. Это – один из наиболее эффективных путей извлечения информации из данных и, если не пытаться увеличить объём памяти (что невозможно), а использовать свёртку по назначению – выявлению различных свойств во входных данных – то её применение не только оправдано, но и необходимо.

Косвенным же доказательством необходимости повышать степень локализации представления данных может служить история развития нейросетей. Первые перцептроны имели полностью связанные слои, но не общую полностью связанность всех элементов. Некоторый шаг назад произошёл в 1982 г. с появлением сети Хопфилда, которая является целиком полностью связанной. Но, если в современных

нейросетевых моделях можно найти полносвязанные слои (и они составляют небольшую часть модели), то сети Хопфилда не используются совсем.

Все нововведения от AlexNet [Krizhevsky, 2017] до GPT-3 [Heaven, 2020] направлены на увеличение степени локальности представления данных.

2.1. Что даёт увеличение степени локализации хранения данных?

Почему прогресс нейросетей идёт путём увеличения степени локализации хранения данных? Что это даёт?

Во-первых, когда на работу каждого из элементов сети влияет только небольшая область (своя) пространства состояний входного сигнала \vec{X} , то не надо «утрясать» настройку параметров со всеми возможными входными сигналами при обучении.

Во-вторых, при воспроизведении изменения в параметрах элементов, которые не отвечают на входные сигналы из текущей области пространства состояний, не происходящие изменения портят преобразование, настроенное для данной области. Это позволяет дообучать сеть без необходимости компенсировать влияние новых данных на параметры. Если используется память с высокой степенью распределённости, то компенсацию осуществляют путём повторного обучения на старых данных вместе с новыми.

В целом это приводит к тому, что, чем выше степень локализации, тем быстрее происходит обучение. И в современных сетях глубокого обучения этот эффект хорошо заметен – именно поэтому большинство нововведений последних лет направлено на повышение степени локализации данных [Vabenko, 2015].

А как же генерализация?! Она никуда не пропадает, поскольку локализация связана с ограничением областей пространства состояний входного сигнала \vec{X} , на которые отвечают как различные структуры сети, так и единичные элементы, а генерализация – с операциями свёртки. Они от увеличения степени локализации никак не страдают и в современных моделях свёртки нисколько не уменьшили своей роли [Kuznetsov, 2018].

2.2. Картирование и аппроксимация.

Среди многочисленных методов увеличения степени локализации хранения данных можно выделить сети с конкурентной активностью. В предельном случае конкуренции активным остаётся один элемент (WTA, winner take all) или несколько (k) элементов (k WTA, k winners take all).

Наиболее популярной областью применения сетей с конкурентной активностью является нейросетевое картирование. Оно позволяет представить массив (тензор по современной нейросетевой терминологии) обучающих данных значительно меньшим набором векторов, описывающих пространство состояний входного сигнала \vec{X} . Причём каждый из сформированных векторов отвечает только сигналам из небольшой области пространства, что достигается равномерным распределением (в результате обучения) векторов по пространству состояний входного сигнала \vec{X} .

Способы построения нейросетевых карт берут начало от самоорганизующихся карт (SOM) Тиуву Кохонена [Kohonen, 1982]. В свою очередь, Кохонен немного развил идеи, заложенные в алгоритм k -средних:

$$\Delta \vec{M}_i = \eta (\vec{X} - \vec{M}_i), \eta \ll 1. \quad (2)$$

При предъявлении произвольного \vec{X} выбирается, ближайший к \vec{X} вектор \vec{M}_i (согласно рис. 2а) – разбиению Вороного-Дирихле,). Затем вычисляется разность $(\vec{X} - \vec{M}_i)$ и \vec{M}_i немного смещается в сторону \vec{X} . Даже такого простого алгоритма достаточно для равномерного распределения векторов \vec{M}_i (рис. 2б) по подпространству состояний вектора \vec{X} (из произвольных начальных положений внутри подпространства).

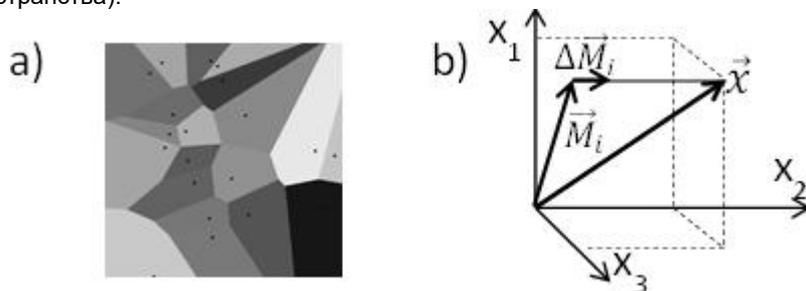


Рис. 2. а) Разбиение Вороного-Дирихле и б) схема изменения вектора \vec{M}_i согласно (2)

Топологически корректное картирование Кохонена только немного сложнее:

$$\begin{aligned} \Delta \vec{M}_j &= \eta_1 (\vec{X} - \vec{M}_j), \quad j: |\vec{X} - \vec{M}_j| = \min \\ \Delta \vec{M}_k &= \eta_2 (\vec{X} - \vec{M}_k) \Delta t, \quad \eta_2 < \eta_1 \text{ для соседей } j. \\ \eta_1 &\rightarrow 0, \quad \sum \eta_1 \rightarrow \infty \text{ при } t \rightarrow \infty, \quad \eta_2 < \eta_1 \end{aligned} \quad (3)$$

Алгоритм -средних можно изменить для нейросетей другим образом:

$$\begin{aligned} \Delta \vec{M}_j &= \eta_1 (\vec{X} - \vec{M}_j), \quad j: (|\vec{X} - \vec{M}_j| + b_j) = \min \\ \Delta b_j &= \eta_3 \\ \Delta b_k &= -\Delta b_j / (N - 1), \quad \forall k \neq j, \quad \eta_3 \rightarrow 0, \sum \eta_3 \rightarrow \infty \text{ при } t \rightarrow \infty, \end{aligned} \tag{4}$$

где η_1, η_2, η_3 – изменяющиеся в процессе обучения коэффициенты.

Картирование позволяет выбрать и запомнить вектора \vec{X} из подпространства, где они определены, и применять их для аппроксимации преобразования $\vec{X} \rightarrow \vec{Y}$. Обычно используют аппроксимацию нулевого порядка, но возможны аппроксимации первого и более высоких порядков. Рис. 3 создаёт впечатление, что переход к аппроксимации первого порядка (кусочно-линейной) не даёт больших преимуществ перед нулевым порядком. Поскольку увеличивая число опорных точек аппроксимации в N раз, мы во столько же раз уменьшаем ошибку. Но это только в одномерном случае. Если подпространство векторов \vec{X} имеет размерность l, то для уменьшения ошибки аппроксимации в N раз потребуется увеличения количества опорных точек аппроксимации в N^l раз, что даже при N = 10 и l = 7 ÷ 10 ведёт к кардинальному усложнению процесса аппроксимации. Не только к увеличению требуемого объёма памяти, но и к росту объёма обучающих данных и времени обучения.

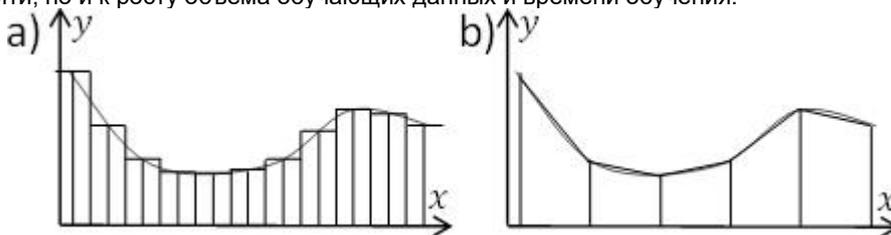


Рис. 3. Аппроксимации а) нулевого и б) первого порядка

Кусочно-линейная аппроксимация (первого порядка) тоже не лишена недостатков, но позволяет значительно сэкономять на объёме памяти и, главное – сократить время обучения. Показанные на Рис. 4 переключения от \vec{M}_2 к \vec{M}_3 и \vec{M}_3 к \vec{M}_4 прекрасно подходят для реализации кусочно-линейной аппроксимации. Не является простым вопросом о том, сколько элементов должно участвовать в аппроксимации. Если размерность подпространства $\{\vec{X}\} = R$, то следует взять R + 1 элемент. Но беда в том, что размерность R подпространства $\{\vec{X}\}$ заранее не известна.

Если размерность R = 1, то следует выбрать 2 элемента и аппроксимация будет выглядеть, как показано на рис. 4:

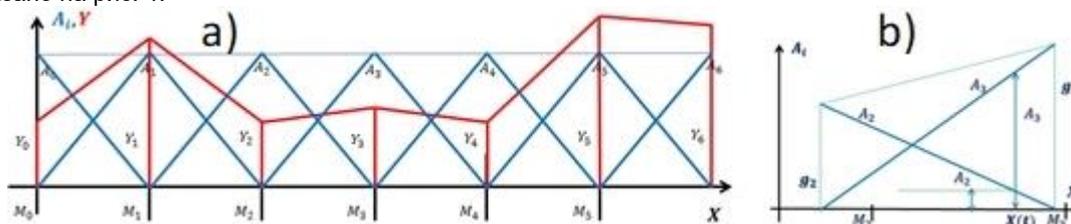


Рис. 4. Активности $A_i(\vec{X})$ для а) регулярной и б) нерегулярной решёток

Активность $A_i(\vec{X})$ каждого элемента достигает максимального значения g_i , можно показать, что в области активности одного набора элементов для любого $\vec{X}(t)$ выполняется соотношение:

$$\sum_i \frac{A_i}{g_i} = 1, \text{ для Рис. 4(b) у суммы два члена } \frac{A_2}{g_2} + \frac{A_3}{g_3} = 1 \tag{5}$$

Важность соотношения (5) трудно переоценить, поскольку оно позволяет определить, сколько членов должно входить в сумму в (5). То есть выяснить размерность R подпространства $\{\vec{X}\}$.

Для этого необходимо знать значения g_i , но они тоже доступны для вычисления, как методами аналитической геометрии, так и статистически. Не будем приводить эти формулы. Однако они были выведены, проведён анализ топологических особенностей аппроксимации в многомерных случаях, что подтвердило работоспособность данного подхода.

2.3. Запись и считывание

Распространено представление, что «нейросетевые» вычисления позволяют формировать очень сложные преобразования входного сигнала \vec{X} в выходной \vec{Y} . Это верно в том смысле, что формируются достаточно сложные для аналитического описания (представления стандартными алгебраическими функциями) отображения \vec{X} в \vec{Y} .

Если посмотреть на вид законов обучения при картировании (2)-(4) и изменения весов выходных связей при ВРЕ, то можно заметить, что вектора входных и выходных связей просто смещаются к зна-

чениям векторов \vec{X} и \vec{Y} . По существу, осуществляется запись значений векторов \vec{X} и \vec{Y} , подававшихся сети при обучении на веса связей формальных нейронов.

Если в сети с конкурентной активностью область kWTA перемещается по элементам слоя, то на выход будут последовательно воспроизводиться (с линейной аппроксимацией) вектора весов выходных связей \vec{W}_j этих элементов (рис. 5):

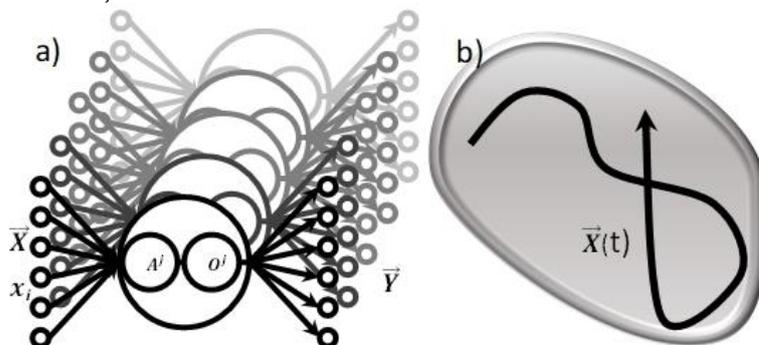


Рис. 5. Смещение центра активности по а) структуре формальных нейронов и б) пространству состояний входного сигнала

Выходной сигнал \vec{Y} будет формироваться как сумма векторов выходных связей \vec{W}_j , каждый из которых умножен на выходную активность своего элемента.

$$\vec{Y}(t) = \sum_{j=1}^N O^j(\vec{X}, t) \vec{W}_j. \quad (6)$$

Схема записи-воспроизведения выглядит очень простой, но это действительно так только в том случае, если размерность пространства состояний входных сигналов \vec{X} невелика. Лучше всего если пространство одномерно. Тогда наблюдается полная аналогия считыванию с магнитофонной плёнки.

В многомерных случаях всё становится сложнее: даже в случае, когда входной сигнал \vec{X} непрерывно меняется и оставляет за собой след в виде одномерной траектории, вероятность попадания новых сигналов \vec{X} , отличных от использовавшихся при обучении на ту же траекторию, стремится к нулю. Необходимо осуществлять аппроксимацию по пространствам \vec{X} и \vec{Y} , а, как это отмечалось выше, с ростом размерности это требует экспоненциально возрастающих затрат.

Из этого следует требование осуществлять декомпозицию сложных сигналов на более простые компоненты, чтобы пространства состояний описывающих эти компоненты векторов \vec{X}_c имели невысокую размерность ($R < 10 \div 15$).

3. Декомпозиция

3.4. Стандартный приём при решении сложных задач

Выделение человека из животного мира принято связывать с членораздельной речью, которая позволила сформировать общества с глубоким разделением труда и многоуровневой иерархической структурой. По сути, разделение труда – это декомпозиция задач, стоящих перед сообществом, на более простые компоненты. Но и отдельный человек, когда сталкивается со сложной задачей, стремится разделить её на более простые подзадачи.

Из анализа проблем аппроксимации понятно, что сложность аппроксимации преобразования \vec{X} в \vec{Y} , в основном, определяется размерностью подпространства состояний вектора \vec{X} в сенсорном пространстве. Размерность подпространства, в свою очередь, связана с числом независимых переменных, описывающих процесс выполнения задачи. Осуществляя декомпозицию задачи на простые компоненты, мы разделяем переменные по компонентам, что приводит к тому, что подпространства состояний этих компонент, формируемые векторами \vec{X}_c , имеют более низкую размерность. Есть все основания считать, что в ежедневной практике разделение труда и декомпозиция задач на простые компоненты оказывается высокоэффективной именно по этой причине.

3.2. Декомпозиция в современных DNN

Современные сети глубокого обучения (DNN) решают довольно сложные задачи, иногда на уровне, превосходящем человеческие возможности. Как им удаётся обходиться без проведения декомпозиции сложных сигналов (с которыми они работают) на более простые компоненты?!

С одной стороны, гигантские и всё время возрастающие ресурсы, затрачиваемые на формирование аппроксимаций преобразований \vec{X} в \vec{Y} , позволяют немного продвигаться в область высоких размерностей подпространств состояний \vec{X} и решать некоторые (но не все) сложные задачи.

С другой стороны, можно утверждать, что декомпозиция всё-таки уже осуществляется. В свёрточных сетях поле сигнала явно разделяется на части, в многослойных структурах на разных слоях выделяются различные признаки, различные функции внимания и переключатели (gates) тоже направлены на выделение частей сигнала из его сложных векторов.

Более того, можно предположить, что при обучении по алгоритму BPE происходит неявная декомпозиция сигналов. Поскольку BPE направлено на оптимизацию преобразования, а декомпозиция – один из путей такой оптимизации, то случайное формирование декомпозиции сигнала должно быть усилено алгоритмом BPE. Другое дело, что вероятность спонтанного появления декомпозиции в слоях DNN невелика... Но, если увеличивать число слоёв DNN, то небольшая вероятность, умноженная на сотни слоёв, может стать почти достоверным событием, которое из раза в раз реализуется с вероятностью, близкой к единице. Это может служить объяснением явного преимущества «глубоких» сетей над «мелким» (имеющими много меньше слоёв при одинаковом с «глубокими» суммарном числе элементов).

3.3. Пути осуществления декомпозиции

В настоящее время основным путём выявления существенных переменных, описывающих подпространства состояний входных сигналов \vec{X} , является использование автокодировщиков с «узким горлом» [Liou, 2013]. Процесс оптимизации преобразования с использованием BPE вынуждает структуру сети выделять небольшое число параметров, которые позволяют по ним восстановить исходный сигнал. Зачастую эти параметры соответствуют существенным переменным объектов и явлений, формирующих сложный входной сигнал. Есть ряд подходов с использованием мультимодальных GANs [Li, 2019], которые позволяют сделать этот процесс более направленным. А уже знание существенных переменных даёт возможность целенаправленно изменять свойства генерируемых сигналов. Это, по существу, эквивалентно декомпозиции.

Кроме того, как говорилось выше, декомпозиция осуществляется за счёт структурного построения сети и вероятностных возможностей, которые имеются в многослойных сетях при выполнении любых преобразований.

Но все эти способы относятся к сетям DNN без явно выраженной конкурентной активности и не осуществляющим картирование. Как можно строить карты простых компонент сложного сигнала, когда они смешаны в единый поток?

Основная идея состоит в том, что каждая карта простых компонент воспроизводит не только входной сигнал \vec{Y} , но и входной сигнал \vec{X} . Нейросеть состоит из значительно числа подсетей (на одном слое), способных формировать карты простых объектов и явлений. Конкуренция активности осуществляется не только между элементами (с номерами m) в каждой подсети, но и между подсетями (с номерами i, j). Каждая подсеть получает свою часть входного сигнала (рис. 6 и формулы (7)-(10)):

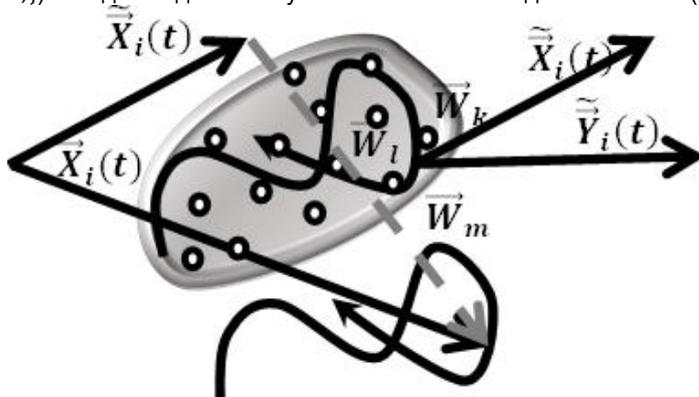


Рис. 6. Одна из подсетей, конкурирующая за «свою» часть входного сигнала $\vec{X}_i(t)$

Изменение значений векторов $\vec{W}_{x_i}^m$ и $\vec{\tilde{W}}_{x_i}^m$ осуществляется в направлении разности $\vec{X}_i(t) - \vec{\tilde{X}}_i(t)$ (серая пунктирная стрелка)

Подсети, которые в процессе конкуренции с другими подсетями сохранили свою активность, выделяют элементы m с ненулевой активностью O_i^m и вносят вклад $\vec{\tilde{X}}_i^m(t)$ в выход $\vec{\tilde{X}}_i(t)$ своей подсети i вектором $\vec{\tilde{W}}_{x_i}^m$:

$$\vec{\tilde{X}}_i^m(t) = O_i^m(\vec{X}_i(t)) * \vec{\tilde{W}}_{x_i}^m, \quad (7)$$

где i – номер подсети, а m – номер элемента. Целиком выход подсети $\vec{\tilde{X}}_i(t)$ складывается как сумма выходов всех элементов с ненулевой активностью $\vec{\tilde{X}}_i^m(t)$:

$$\vec{\tilde{X}}_i(t) = \sum_{m=1}^N \vec{\tilde{X}}_i^m(t). \quad (8)$$

Этот суммарный выход подсети $\tilde{X}_i(t)$ сравнивается с предназначенной этой подсети части входа $\bar{X}_i(t)$, которая формируется как разность между сложным входным сигналом $\bar{X}(t)$ и суммой выходов всех подсетей слоя $\tilde{X}_j(t)$, кроме этой:

$$\tilde{X}_i(t) = \bar{X}(t) - \sum_{j \neq i}^N \tilde{X}_j(t). \quad (9)$$

Если сгенерированный вектор $\tilde{X}_i(t)$ отличается от поданного на вход i -той подсети $\bar{X}_i(t)$, то вектора весов связей \bar{W}_i^m и \tilde{W}_{xi}^m (Рис. 1) изменяются по следующему закону:

$$\Delta \bar{W}_i^m = \Delta \tilde{W}_{xi}^m = \Delta_m = \alpha O_i^m * (\bar{X}_i(t) - \tilde{X}_i(t)). \quad (10)$$

На «молчашие» подсети слоя также поступает их часть сигнала, вычисляемая по формуле (9) и, если подсеть способна сформировать $\tilde{X}_i(t)$ близкий к своему $\bar{X}_i(t)$, то такая подсеть начинает побеждать в конкуренции между подсетями и становится активной. Наоборот, те подсети, у которых разность $\tilde{X}_i(t) - \bar{X}_i(t)$ становится большой, начинают проигрывать в конкурентной борьбе между подсетями за получение активности.

Формулы (7)-(10) описывают только основную идею формирования карт простых компонент сложного сигнала. Для того, чтобы процесс сходился, необходимо обеспечить выполнение ряда свойств как структуры взаимодействующих подсетей, так и сложного входного сигнала. Понятно, что если на вход подавать белый шум, то никаких компонент из него выделить не удастся.

3.4. Иерархия как необходимая часть декомпозиции

Если сигнал от сложного объекта и процесса можно разложить на компоненты, то для создания модели всего текущего состояния необходимо осуществлять согласование работы моделей его компонент. Магистральным путём такого согласования представляется формирование иерархической структуры карт подпространств состояний простых объектов и явлений. Если для нижних уровней такой структуры входным сигналом $\bar{X}(t)$ является сенсорное возбуждение, прошедшее предобработку, то для более высоких уровней входом служит активность нижних слоёв иерархии, отражённая на межуровневые экраны.

Так же, как нижние уровни выделяют во входном сигнале простейшие свойства (если они есть), так и более высокие уровни пытаются выделить более генерализированные свойства активности нижних слоёв. Это тоже возможно только при условии, что такие свойства в активности элементов нижних слоёв существуют.

Не следует думать, что с ростом уровней число конкурирующих карт в слоях должно падать. Наоборот, теоретически оно могло бы расти вместе с увеличивающимся числом возможных комбинаций всё большего числа объединяемых компонент. Но «комбинаторного взрыва» не происходит, поскольку мы не в состоянии (даже мысленно) перебрать все возможные комбинации и попробовать найти в них зависимости для выявления более высокими уровнями. Фактически, возможности по разнообразию формируемых карт определяют структурой сети, созданной до начала обучения в которой задано количество конкурирующих подсетей, формирующих карты.

Можно сопоставлять себе такие подсети в структуре слоя с колонками в коре головного мозга, хотя система конкурирующих карт в искусственных нейрослоях была предложена в результате анализа возможных путей преодоления кризиса описания сложных сигналов, а не как результат исследования морфологических данных. Биологические данные можно трактовать так, что выделяемые в коре структуры позволяют организовать не два уровня конкуренции за активность (подсети и элементы в подсетях), а три (колонки, микроколонки и элементы микроколонок)

В отличие от стабильной структуры живых нейросетей, если и подверженной изменениям, то медленным и не большим по величине, реализованные на компьютере модели нейросетей легко могут менять свою структуру и перераспределять память и число подсетей среди слоёв иерархической структуры. Это даёт потенциальные возможности для более адекватного описания сложных сигналов как суммы простых компонент без явной зависимости от соответствия заранее созданной структуры сети свойствам наблюдаемых сложных сигналов. Структуру компьютерной модели сети возможно изменять под свойства сигнала в процессе обучения.

3.5. Понимание

Иерархическая структура линеаризованных конкурирующих карт с высокой локализацией хранения данных позволяет не только отслеживать изменяющиеся состояния входного сигнала и быстро подстраиваться под него. Она позволяет оценивать степень соответствия построенных моделей наблюдаемому новому сложному процессу, который раньше никогда не формировал входной сигнал. Комбинаций возможных состояний простых объектов и явлений слишком много, чтобы их можно было заранее все просмотреть или даже промоделировать. Но состояния простых объектов и явлений могут быть изучены (то есть откартированы) в полном объёме. Именно этим простые объекты и явления отличаются от сложных.

Наличие (почти) полных карт простых объектов и явлений, организованных в иерархическую структуру, позволяет оценивать «понимание» впервые встретившейся ситуации: если произведённая по формулам (7)-(9) декомпозиция сигнала позволяет описать ситуацию с высокой точностью, то «понимание» есть. А норма величины $\bar{X}_i(t)$, поступающей на «молчащие» картирующие подсети характеризует степень понимания: чем эта норма выше, тем степень понимания меньше.

«Понимание», выраженное количественной оценкой, позволяет не только оценивать, насколько полезно будет выполнение наработанных последовательностей действий с простыми объектами (решать продолжать выполнение или прекратить), но и непосредственно осуществлять считывание этих последовательностей при смещении положений компонентов векторов входных сигналов $\bar{X}_i(t)$ на картах. При изменениях состояний наблюдаемых объектов и задаваемых целей происходят переходы в активные состояния элементов (кТА), что можно рассматривать как смещение «центров активности» (ЦА) по структурам карт. Смещение происходит в соответствии с происходящими изменениями, которые могут быть вызваны как внешними, так и внутренними причинами. В процессе смещения ЦА по структуре карт происходит не только сравнение воспроизводимых $\bar{X}_i(t)$ с поступающими на карту $\bar{X}_i(t)$, но и воспроизведение $\bar{Y}_i(t)$. Этим же обеспечивается соответствие выполняемых действий изменяющимся состояниям наблюдаемых объектов. Отметим, что $\bar{Y}_i(t)$ – это не обязательно активные действия, отсутствия действий – тоже вид поведения, который соответствует большому числу ситуаций.

Естественно, что «субъективные» действия определяются не только состояниями наблюдаемых объектов, но и целями «субъекта». Для каждого нижележащего уровня цели формирует уровень, расположенный в иерархии на одну ступень выше. В отличие от них самый верхний уровень просто выдаёт нижнему уровню цели (из числа имеющихся), наилучшим образом соответствующие сложившейся ситуации.

4. Линеаризация

Процесс записи входных и выходных сигналов на структуру весов связей сетей формальных нейронов путём формирования карт подпространств состояний простых объектов и сигналов прост только концептуально. Для его технической реализации необходимо преодолеть ряд проблем описания и регулирования процессов конкуренции за активность, сходимости декомпозиций и других. Но формируемые через активности картирующих подсетей отображения \bar{X} в \bar{Y} могут описывать очень широкий класс преобразований. Среди которых особое значение имеют нелинейные преобразования, позволяющие отображать сложные зависимости в линейные.

Подпространства состояний простых объектов и процессов в сенсорном пространстве, как правило, будут иметь нелинейную форму. Но, после построения карт этих подпространств в них можно ввести нелинейные оси, в которых представление подпространств будет линейным.

Линейность представления подпространств простых объектов, описываемых векторами \bar{X}_i , не даёт сама по себе линейной зависимости $\bar{Y} = \sum_{i=1}^N \vec{F}(\bar{X}_i)$. Но в ряде случаев с использованием нелинейных преобразований такие преобразования тоже возможны.

Выделение существенных для задач формирования поведения компонент сигнала в принципе может быть осуществлено и при нелинейных представлениях отображений \bar{X} в \bar{Y} . Но проведение линеаризации отображений делает задачу значительно более простой.

В целом, линеаризация представления преобразований открывает дополнительные возможности к разделению решаемых задач, повышению компактности их описания и эффективности процессов аппроксимации оптимизации и прогнозирования.

5. Другие возможности

Построение иерархической структуры линеаризованных конкурирующих карт с высокой локализацией – это путь к решению ряда современных проблем, стоящих на пути развития ИИ, как направления науки.

Иерархическая декомпозиция сложных сигналов на простые компоненты делает работу таких нейросетей объяснимой, поскольку модули, в которых формируются карты простых составляющих, структурно определены заранее, а простота компонент позволяет понять содержание сформированных отображений.

Высокая степень локальности хранения данных позволит заметно ускорить процесс обучения и сделать переобучение отдельных компонент практически независимым от полного обучения.

Но самое главное в иерархической декомпозиции в том, что она формирует систему описания наблюдаемого мира, отражающую как выделенные в нём зависимости, так и независимость ряда компонент друг от друга. Это открывает очень широкие возможности по использованию уже полученных знаний для ускорения решения ранее не встречавшихся задач.

5.1. Использование имеющихся знаний

Если сигнал формируется в результате наблюдения сложной сцены, то возможности по его декомпозиции напрямую связаны с тем, какая часть простых объектов и явлений, составляющих сцену,

уже имеет настроенные под них карты. В случае, когда незнакомых объектов нет совсем, декомпозиция по формулам (7)-(9) осуществляется легко и если и требует дообучения, то незначительного. В случае, когда в сцене появляется 1-2 незнакомых объекта, то для них достаточно быстро можно сформировать карты, поскольку новые объекты просты и малочисленны. Если же неизвестных (даже простых) объектов много, то процесс формирования их карт или невозможен совсем, или требует очень значительного времени.

5.2. Мышление и система 2 по Канеману

Иерархическая структура записи информации об окружающем мире, выполняемых действиях и взаимосвязи между ними должна быть настроена на считывание лучших вариантов действий, выполненных в аналогичной ситуации. Как правило, повторяющиеся действия осуществляются с простыми объектами и явлениями, а сложные ситуации практически никогда не повторяются.

Тем не менее, необходимо выбирать, какие действия и с какими простыми объектами лучше выполнить во впервые встретившейся ситуации. На выбор могут повлиять оценка степени «понимания» ситуации и результат аппроксимации действий в чём-то похожих ситуаций. Но для сложных, впервые встретившихся ситуаций, аппроксимация вряд ли будет хорошей (если не было похожих ситуаций или предыдущие действия в них были не удачными). Но если выбор необходимо осуществить срочно, то лучше всё-таки последовать имеющимся навыкам действий, пусть и в не очень похожих ситуациях.

Если же время есть (а, как правило, цели, поставленные нижним уровням, требуют времени для их достижения), то можно иерархическую систему карт верхних уровней использовать для моделирования последствий разных вариантов действий. Не всегда на это есть время, но такой медленный (относительно, он всё-таки быстрее и менее затратный, чем непосредственное выполнение моделируемых вариантов) перебор последовательностей действий позволяет заметно улучшить выбор ближайших целей. И, что важно, это не требует организации отдельной системы 2 по Канеману [Kahneman, 2011], а использует ту же самую иерархическую структуру карт, которая регулярно применяется для формирования текущего поведения и отслеживается на соответствие реальным объектам и явлениям.

Когда ситуация не требует уточнения целей текущих действий, моделирование на верхних уровнях иерархической структуры может быть осуществлено для ситуаций, не имеющих прямого отношения к текущему состоянию дел. Это аналогично обучению на аугментации входных сигналов (созданию из одного входного сигнала множества похожих путём применения к исходному сигналу заданных классов трансформаций с разными значениями параметров трансформации и использовании их для обучения). Но для карт более высокого уровня, работающих на генерализованных сигналах и взаимодействующих с картами более низкого уровня возможностей для обучения на сгенерированных (внутри самой сети!) сигналах значительно больше.

5.3. Формирование целей

Поскольку верхние уровни иерархической системы карт участвуют в формировании последовательностей действий постановкой текущих целей для нижних уровней, обучение в виде формирования карт, как за счёт опыта взаимодействия с реальным миром, так и путём моделирования сложных ситуаций, по существу является формированием целей. Но для выбора целей необходимы сравнительные оценки разных последовательностей действий для возможности выбора лучшей.

У животных и человека такие оценки формируются в «центрах удовольствия», которые собирают не только положительные, но и отрицательные эмоции. Хотя это сложные и тонко настроенные в процессе эволюции системы, нет причин считать, что их воспроизведение техническими средствами невозможно. Важно не пытаться слепо копировать биологические прототипы, а понимать смысл их работы. А он не такой сложный, как кажется. Для выполнения любых задач животным, человеку и «интеллектуальным» устройствам требуется сохранять свою целостность, работоспособность, обеспечивать энергоснабжение. Для этого необходимо получать новые знания, позволяющие лучше прогнозировать последствия своих действий и находить баланс дружелюбных и враждебных отношений с объектами и субъектами во внешнем мире.

Список требований можно расширить и уточнить, и он не будет очень простым. Но, даже не обладая никакими представлениями ни о мышлении, ни о генетике, человечество смогло вывести полезные и дружелюбные породы собак. Сейчас мы находимся в значительно лучшей позиции, что даёт основания надеяться, что задачу создания полезных и дружелюбных агентов ИИ нам тоже удастся решить. После чего весь круг задач создания СИИ будет замкнут.

Социальные аспекты создания СИИ были рассмотрены авторами в [Малинецкий, 2021].

6. Выводы

Надвигающийся «мягкий» кризис развития не приведёт к новой «зиме» ИИ. Речь будет идти только о том, останутся ли темпы развития возможностей «интеллектуальных устройств» столь же высокими или даже ускорятся или будет наблюдаться некоторое замедление темпов роста. Возможности экстенсивного использования имеющихся ресурсов скоро могут подойти к пределу, если рассчитывать в основном просто на увеличение мощности систем и объёмов данных. На повышение интереса к альтер-

нативным подходам к развитию ИИ направлены учебники (например, [Николенко, 2017]) и государственные указы [Указ, 2019].

Сейчас особенно важно обратить пристальное внимание на использование математических идей, позволяющих уменьшить зависимость от объемов обрабатываемых данных. И, хотя теорему Эшби [Ashby, 1956] о необходимой сложности управляющих систем никто не отменял и простыми «интеллектуальные» системы быть не могут, следует рассматривать варианты, повышающие эффективность использования имеющихся ресурсов. И, как обсуждалось выше, такие возможности есть!.

Литература

1. Кай-Фу Ли. Сверхдержавы искусственного интеллекта. Китай, Кремниевая долина и новый мировой порядок. // Манн, Иванов и Фербер; Москва; 2019. – 238 с. ISBN 978-5-00146-163-0.
2. Малинецкий Г.Г., Смолин В.С. О развитии прикладной математики, искусственного интеллекта и компьютерных вычислений // Препринты ИПМ им. М.В.Келдыша. 2021. № 69. 49 с. <https://doi.org/10.20948/prepr-2021-69>.
3. Николенко С., Кадури А., Архангельская Е. Глубокое обучение. Погружение в мир нейронных сетей // Питер; 2017.
4. Указ Президента Российской Федерации от 10.10.2019 № 490 "О развитии искусственного интеллекта в Российской Федерации". // <http://publication.pravo.gov.ru/Document/View/0001201910110003>. ноябрь 2019.
5. Bengio Y., LeCun Y., and Hinton G. Deep Learning for AI. Communications of the ACM, , Vol. 64 No. 7, Pages 58-65. July 2021.
6. Chen, X., Fan, H., Girshick, R., and He, K. Improved baselines with momentum contrastive learning, 2020; arXiv:2003.04297.
7. Kahneman, D. Thinking, Fast and Slow. Macmillan, 2011.
8. Silver D.; Hubert T., ...; Hassabis D. Mastering Chess and Shogi by Self-Play with a General Reinforcement Learning Algorithm. 2017 arXiv:1712.01815.
9. Krizhevsky, Alex; Sutskever, Ilya; Hinton, Geoffrey E. (2017-05-24). "ImageNet classification with deep convolutional neural networks" Communications of the ACM. 60 (6). 84–90. doi:10.1145/3065386. ISSN 0001-0782. S2CID 195908774.
10. Heaven Will Douglas. A GPT-3 bot posted comments on Reddit for a week and no one noticed (англ.). MIT Technology Review (8 October 2020).
11. Babenko A., Lempitsky V. Aggregating local deep features for image retrieval Proceedings of the IEEE international conference on computer vision, 1269-1277, 2015.
12. Kuznetsov M., Oseledets I. Tensor Train Spectral Method for Learning of Hidden Markov Models (HMM) Computational Methods in Applied Mathematics, V 19, № 1, с. 93-99 DOI/10.1515/cmam-2018-0027.
13. Kohonen, T. "Self-Organized Formation of Topologically Correct Feature Maps". Biological Cybernetics. 43 (1): 59–69. 1982. doi:10.1007/bf00337288. S2CID 206775459.
14. Liou, C.-Y., Cheng, C.-W., Liou, J.-W., and Liou, D.-R., Autoencoder for Words. Neurocomputing, Volume 139, 84-96 (2014), doi:10.1016/j.neucom.2013.09.055.
15. Li, J., Ma, H., Tomizuka, M.: Conditional generative neural system for probabilistic trajectory prediction. In: International Conference on Intelligent Robots and Systems. (2019).
16. Ashby W.R. Introduction to Cybernetics, Chapman & Hall, 1956, ISBN 0-416-68300-2.

References in Cyrillics

1. Kai-fu Lee. The superpowers of artificial intelligence. China, Silicon Valley and the New World Order. // Mann, Ivanov and Ferber; Moscow; 2019. – 238 p. ISBN 978-5-00146-163-0.
2. Malinetsky G.G., Smolin V.S. On the Development of Applied Mathematics, Artificial Intelligence and Computer Computing // IPM Preprints im. M.V. Keldysh. 2021. No. 69. 49 p. <https://doi.org/10.20948/prepr-2021-69>.
3. Nikolenko S., Kadurin A., Arkhangelskaya E. Deep learning. Dive into the world of neural networks // Peter; 2017.
4. Decree of the President of the Russian Federation of October 10, 2019 No. 490 "On the development of artificial intelligence in the Russian Federation". // <http://publication.pravo.gov.ru/Document/View/0001201910110003>. November 2019.

Ключевые слова

искусственный интеллект, нейросети, локализация, декомпозиция, линеаризация, кризис искусственного интеллекта, иерархия, прикладная математика, самоорганизация, векторное описание понятий, понимание

*Малинецкий Георгий Геннадьевич – д.ф.-м.н., профессор, заведующий отделом,
Институт прикладной математики им. М.С. Келдыша, РАН,
ORCID: [0000-0001-6041-1926](https://orcid.org/0000-0001-6041-1926), E-mail: GMalin@keldysh.ru*

*Войцехович Вячеслав Эмеринович – д.ф.н.,
профессор, ФГБОУ ВО Тверской государственный университет,
ORCID: 0000-0002-8093-7121, E-mail: synerman@gmail.com*

*Смолин Владимир Сергеевич –
Институт прикладной математики им. М.С. Келдыша, РАН, ORCID:0000-0001-9030-6545,
E-mail: smolin@keldysh.ru*

George Malinetsky, Vyacheslav Voitsekhovich, Vladimir Smolin, Ways to overcome the impending crisis of artificial intelligence development

Keywords

artificial intelligence, neural networks, localization, decomposition, linearization, artificial intelligence crisis, hierarchy, applied mathematics, self-organization, vector description of concepts, understanding

DOI: 10.34706/DE-2022-05-01

JEL classification C02 – Математические методы; M15 Управление информационными технологиями.

Abstract

Over the past decade, the increase in the number of adjustable parameters in modern "neural network" computing schemes by 8-10 decimal orders up to hundreds of billions and tens of trillions has made it possible to achieve outstanding results in solving a wide class of "intelligent" problems. Increasing the complexity of control devices, according to Ashby's theorem on the necessary variety of control systems, is potentially required to solve more difficult problems. If the cost, training time, initial data volumes and even the electricity amount consumed by "neural networks" grew in proportion to the growth of the success of "neural network" calculations, then the limits of development possibilities would have been reached long ago. The use of an ever-wider range of mathematical ideas, algorithms optimization, information collection automation and improvement in electronics have reduced the rate of cost growth. As we approach the limits of reasonable costs for solving "intellectual" problems, the focus of development is increasingly transferred from the backpropagation error algorithm (BPE) hardware implementation to other mathematical approaches. Success in the implementation of such simple ideas as memory localization, decomposition of complex objects and linearization of transformations can give a new impetus to the artificial intelligence (AI) development, comparable to the introduction of the idea of gradient descent in training "deep neural networks", which has already led to a revolution in machine learning.