

Федеральное государственное автономное образовательное учреждение
высшего образования «Национальный исследовательский университет
«Высшая школа экономики»

Факультет компьютерных наук
Основная образовательная программа
Прикладная математика и информатика

ГРУППОВАЯ КУРСОВАЯ РАБОТА
ИССЛЕДОВАТЕЛЬСКИЙ ПРОЕКТ НА ТЕМУ
"РОБОТ ИССЛЕДОВАТЕЛЬ"

Выполнили студенты:

Зыбин Михаил Александрович, БМТ161, 4 курс

Семавина Юлия Георгиевна, БПМИ171, 3 курс

Руководитель КР:

доцент Устюжанин Андрей Евгеньевич

Консультант:

Стажер-исследователь, аспирант Белавин Владислав Сергеевич

Москва 2020

Содержание

1	Введение	2
2	Обзор литературы	3
2.1	SciNet	3
2.2	Модель AI Physicist	4
2.2.1	Термины, используемые в данном разделе	5
2.2.2	Обзор решаемой задачи	5
2.2.3	Архитектура	6
2.2.4	Отличия предлагаемой нами модели RoboScientist от модели AI Physicist	7
3	RoboScientist	8
4	Исследование методов изучения теорий	9
4.1	Использование производных при приближении функции	9
4.1.1	Термины и обозначения	9
4.1.2	Метод, предложенный в статье	10
4.1.3	Обзор проведенных экспериментов и полученных результатов	11
4.2	Исследование метода обучения с использованием понятия глубины формулы	12
5	План дальнейшей работы	14

1 Введение

Машинное обучение отлично справляется с задачами выделения необходимой информации из данных. Такие задачи часто приходится решать в естественных науках, например, в физике часто приходится анализировать результаты наблюдений, находить в них закономерности и моделировать новые эксперименты для получения дополнительных данных и подтверждения теорий. Поэтому развитие в области машинного обучения и искусственного интеллекта в целом может поспособствовать развитию научного эксперимента. Машинное обучение уже успешно применяется для решения конкретных физических задач, однако, можно поставить и другой вопрос: может ли искусственный интеллект самостоятельно открывать ранее никому не известные законы по экспериментальным данным? Хотя этот вопрос остается открытым и многие полагают, что он таким и останется в ближайшее десятилетие [Uri Alon et al. \(2018\)](#), важные исследования были проведены в этом направлении. В [Alhousseini et al. \(2019\)](#) приводится обзор этих исследований.

В своей работе мы хотим развить идею применения искусственного интеллекта для решения реальных физических задач и открытия новых законов, а именно, мы хотим создать универсального агента, который мог бы исследовать различные среды с различными физическими законами на основании проводимых им экспериментов и предыдущих знаний. Мы основываемся на модели, предложенной в [Tailin Wu and Max Tegmark \(2019\)](#), которая по экспериментальным данным предсказывает физические законы в различных средах. В нашей работе мы хотим развить и доработать этот подход: мы хотим предложить новую более универсальную модель с модификациями в архитектуре и методах обучения. Предлагаемую нами модель мы в дальнейшем будем называть RoboScientist.

2 Обзор литературы

Как мы уже упомянули выше, центральный вопрос нашего исследования - может ли искусственный интеллект самостоятельно открывать ранее никому не известные законы по экспериментальным данным? (Для более подробного обзора исследований, связанных с этим вопросом, и знакомства с ключевыми используемыми методами можно обратиться к [Alhousseini et al. \(2019\)](#)).

Обычно искусственный интеллект для выявления законов используется следующим образом: собираются экспериментальные данные и передаются нейронной сети. На основании человеческого опыта и интуиции на все множество математических формул, которыми может быть представлен изучаемый закон, и на начальные состояния системы накладываются некоторые серьезные ограничения. После чего нейросеть может предлагать из заданного ей множества теории, которые лучше всего описывают данные. Хотя этот метод очень полезен, он не является универсальным: агент не изучает систему самостоятельно с чистого листа, как бы нам хотелось.

2.1 SciNet

Модель SciNet, обладающая большей самостоятельностью, и не требующая никаких ограничений или знаний, была предложена в [Raban Iten et al. \(2020\)](#). Эта модель работает следующим образом: Ей передаются экспериментальные данные, на основе которых она выучивает их представление, с помощью которого в дальнейшем может отвечать на задаваемые ей вопросы, не используя информацию о изначальных экспериментальных данных. Конечно, изначально неизвестно, сколько нейронов потребуется для выучивания представления по экспериментальным данным, поэтому, может так оказаться, что нейронов будет не хватать, и в этом случае модель будет давать неточные ответы на вопросы.

В качестве примера можно рассмотреть простой закон Ома. В качестве экспериментальных данных сеть будет получать силу тока I и результирующую

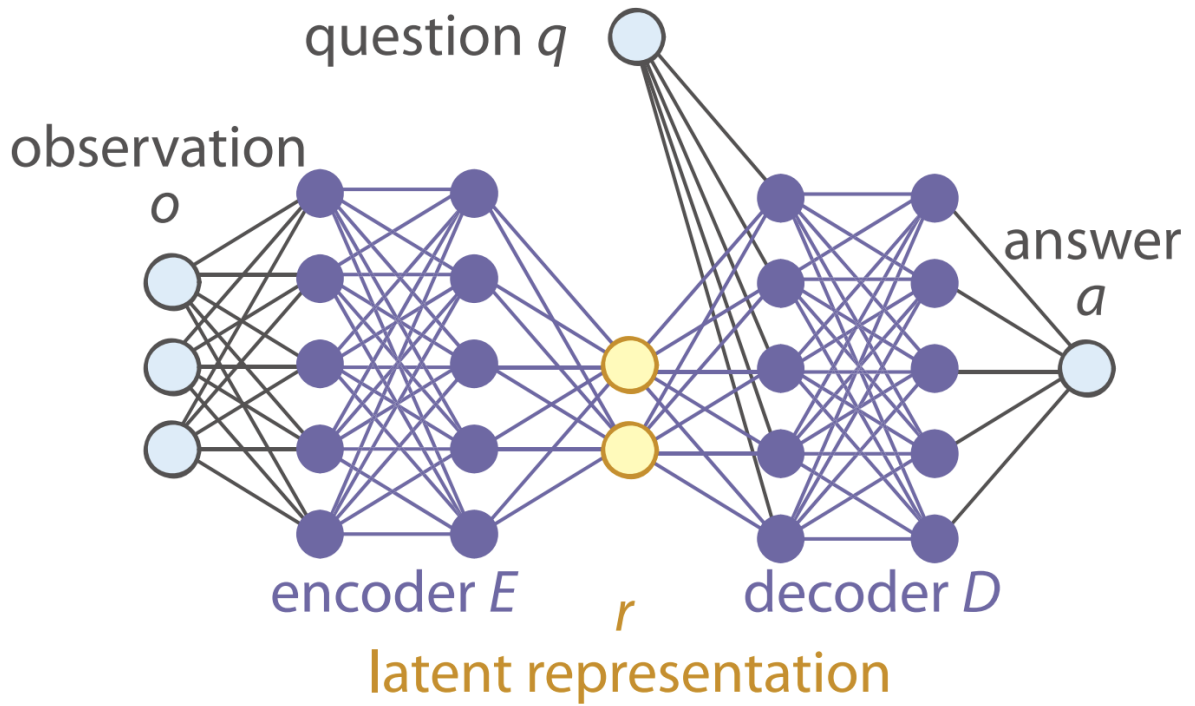


Рис. 1: Архитектура SciNet

щее напряжение U . На основе экспериментальных данных сеть выучит представление, представляющее собой сопротивление R , на основе которого сможет ответить на вопрос о том, чему будет равняться напряжение при данной силе тока.

Данная модель не опирается на внешние знания и не требует ограничений, и умеет отвечать на задаваемые ей вопросы о новых данных, что конечно, является большим продвижением в разрешении центрального вопроса, но она не выучивает точные физические законы в явном виде, что является целью нашего исследования.

2.2 Модель AI Physicist

В этом разделе мы подробно обратим внимание на модель AI Physicist, предложенную в [Tailin Wu and Max Tegmark \(2019\)](#). Это самая близкая к нашему исследованию работа. Цель модели AI Physicist не в возможности делать примерные предсказания, обучившись на экспериментальных данных, а в явном выводе абсолютно точных математических формул, описывающих физическую среду.

2.2.1 Термины, используемые в данном разделе

Теория (theory) - кортеж $\tau = (t, c)$ из двух отображений: f - функция, вычисляющая предсказание модели на точке из области определения данной теории, c - классификатор, определяющий, попадает ли вход в область определения данной теории.

2.2.2 Обзор решаемой задачи

AI Physicist - обучающийся агент. Его задача состоит в изучении теорий для различных сложных сред, в которых действуют различные физические законы. Понятие сложной среды и работу агента проще всего объяснить на примере эксперимента, проведенного в работе [Tailin Wu and Max Tegmark \(2019\)](#). Представим, что среда представляет из себя квадратную огороженную область на плоскости, разделенную на 4 одинаковых квадрата. В разделенной области под действием различных сил передвигается шарик, упруго отражаясь от стенок. В левом верхнем квадрате шарик движется через гармонический потенциал, в левом нижнем квадрате действует гравитационное поле, и в правом нижнем - электромагнитное поле. В этой системе задачей агента является определение границ огороженной области и выявление всех законов движения в ней. На вход агенту подается набор координат - последовательных положений шарика внутри области с течением времени. Агент учится предсказывать следующее положение шарика по двум предыдущим, получая отсюда уравнение движения.

Движение шарика под действием различных сил - лишь одна из задач, которые должен решать агент. Примеров таких задач существует очень много, и AI Physicist должен уметь с ними справляться. Так как целью AI Physicist является симуляция работы физика, агент, исследуя среду, старается вывести точный закон, а не приблизительную формулу. Помимо этого, как и настоящий исследователь, агент делает выводы на основе ранее обретенного опыта.

2.2.3 Архитектура

Основным компонентом архитектуры обучающегося агента является концентратор теорий (Theory Hub), в котором хранятся все выученные теории.

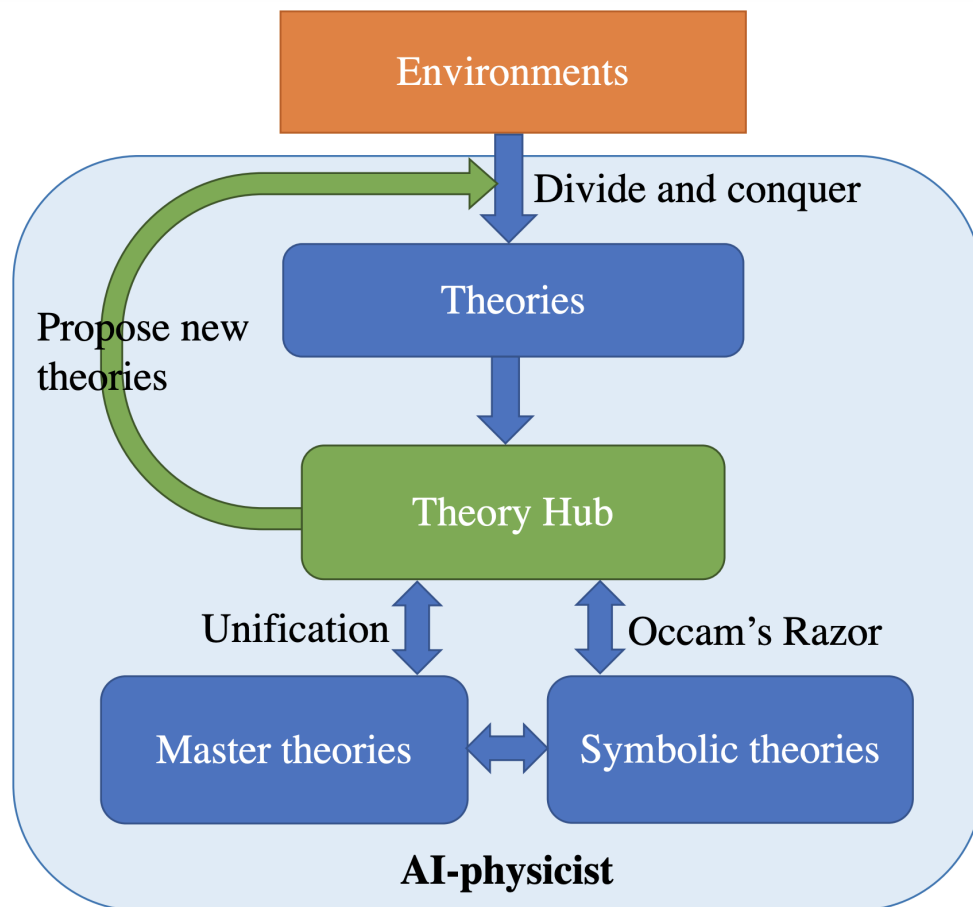


Рис. 2: Архитектура AI Physicist

Процесс исследования новой среды состоит из следующих шагов:

- 1 Агент предлагает теории для описания элементов неизученной среды. Среди этих теорий присутствуют как и совершенно новые, так и старые, выбранные агентом из концентратора, теорий.
- 2 Выдвинутые теории обучаются. После обучения из них выбираются подходящие теории, которые добавляются в концентратор вместе с данными, на которых они показывают хорошие результаты.
- 3 Внутри концентратора теории упрощаются и обобщаются.

2.2.4 Отличия предлагаемой нами модели RoboScientist от модели AI Physicist

Модель AI Physicist использует абстрактное понятие среды и в реальности оперирует с экспериментальными данными. В модели AI Physicist изучение среды строится на обработке входных файлов с фиксированным числом экспериментальных данных. На основании этих данных происходит выучивание теорий. Мы же предлагаем модель, в которой агент бы взаимодействовал со средой. Преимуществами такой модели являются:

- агент сам может решать, какие данные от среды ему необходимы, и на основании протокола взаимодействия получать их
- агент сможет получать столько данных для обучения, сколько требует данная среда
- добавлять новые среды будет очень просто
- агент становится более самостоятельным

3 RoboScientist

RoboScientist - обучающийся агент, задача которого состоит в изучении различных сред и выведении точных математических законов, которым подчиняются эти среды.

Перед тем, как дать более формальную постановку задачи, решаемой нашим агентом, мы дадим определения основных понятий, которыми мы оперируем.

Начнем с понятия среды. Мы называем средой объект, который, получив запрос с набором входных данных, проводит над ними некоторое фиксированное преобразование, получая при этом выходные данные, которые возвращает в качестве ответа на запрос. Например, среда “Закон Ома” способна обрабатывать запросы, в которых входными данными является набор сил тока I , преобразовывать набор данных по правилу $R \times I$ для некоторого фиксированного параметра R и возвращать получившиеся значения. Помимо этого, среда обладает критерием успеха - объектом, который по некоторому набору параметров способен ответить на вопрос, является ли данная среда изученной. Например, критерий успеха может сравнивать переданную ему формулу с реальной формулой среды.

Другим важным понятием является понятие теории. Важно не путать его с понятием теории в обзоре модели AI Physicist. Мы называем теорией объект, хранящий в себе обучаемую модель, способный продолжить обучение модели по экспериментальным данным (входам и выходам среды), и способный протестировать качество обучаемой модели по экспериментальным данным. Теория способна представлять обучаемую модель в символическом виде.

Теперь можно сформулировать задачу RoboScientist. Она заключается в том, чтобы заданной среде построить такую теорию, с помощью которой RoboScientist смог бы добиться от критерия успеха данной среды положительного ответа на вопрос, изучена ли среда.

4 Исследование методов изучения теорий

В этом разделе мы приведем подробное описание проведенных нами исследований и предъявим экспериментальные результаты.

Основным критерием качества нашей модели является точность выученных ей законов. Одной из подзадач, решаемых моделью, является определение символической формулы, описывающей среду, опираясь на данные, полученные агентом. В качестве простого примера такой подзадачи можно рассмотреть нахождение закона Ома $U = IR$, взаимодействуя со средой, которая по силе тока I возвращала бы напряжение U .

4.1 Использование производных при приближении функции

Над этой частью работает Юлия Семавина

В [Gleb Ryzhakov and Ivan Oseledets \(2018\)](#) было замечено, что информация о производных оказывается очень полезной в задаче приближения функции полиномами. Мы предположили, что этот подход, возможно, будет полезен при обучении агента, поэтому решили повторить приведенные в статье результаты и попробовать провести ряд дополнительных экспериментов.

4.1.1 Термины и обозначения

- Полиномы Чебышева - $T_0(x) = 1, T_1(x) = x, T_{n+1}(x) = 2xT_n(x) - T_{n-1}(x)$.
- Полиномы Лежандра - $T_0(x) = 1, T_1(x) = x, T_{n+1}(x) = \frac{2n+1}{n+1}xT_n(x) - \frac{n}{n+1}T_{n-1}(x)$.
- Стандартные полиномы - $T_n(x) = x^n$
- Полиномы Фурье - $T_0 = 1, T_{2n-1}(x) = \cos(nx), T_{2n}(x) = \sin(nx)$

4.1.2 Метод, предложенный в статье

Рассмотрим задачу аппроксимации функции на $[-2, 2]^2$

$$f(x) = \exp(-x_1^2 - 0.5(x_2 - 1)x_2) \quad (1)$$

полиномами Чебышева.

Полиномы разложения P_i строятся по полиномам Чебышева следующим образом:

Для всех степеней d от 0 до N добавляются все полиномы, имеющие вида

$C(x_1)^a C(x_2)^b$, $a + b = d$, где $C(x)$ - полином Чебышева. Полиномы разложения P_i отсортированы по степени, потом по $(\min(a, b))$, потом по a .

В классическом подходе:

$$f(x) \approx \hat{f}(x) = \sum_{i=1}^M \alpha_i P_i(x),$$

где α_i могут быть найдены с помощью метода наименьших квадратов, который будет уменьшать норму остатков в выбранных точках множества $[-2, 2]^2$:

$$\sum (f(x_i) - \hat{f}(x_i))^2 \rightarrow \min$$

.

Рассмотрим другой подход, использующий информацию о градиентах функции f .

Будем решать следующую задачу: $A\alpha = F$, где $\alpha = (\alpha_1, \alpha_2, \dots, \alpha_M)^T$ - вектор весов, упомянутых выше. Назовем множество выбранных точек из $[-2, 2]^2$ множеством

$$G = \{(g_{1,1}, g_{1,2}), (g_{2,1}, g_{2,2}), \dots, (g_{k,1}, g_{k,2})\}.$$

Тогда F - вектор размерности $3k$, состоящий из значений функции F в

точках множества G и градиентов в этих же точках сначала по x_1 , потом по x_2 :

$$F = (f(g_{1,1}, g_{1,2}), f(g_{2,1}, g_{2,2}), \dots, f(g_{k,1}, g_{k,2}), d_{x_1}f(g_{1,1}, g_{1,2}), d_{x_1}f(g_{2,1}, g_{2,2}), \dots, d_{x_1}f(g_{k,1}, g_{k,2}), d_{x_2}f(g_{1,1}, g_{1,2}), \dots, d_{x_2}f(g_{k,1}, g_{k,2})). \quad (2)$$

А матрица A имеет вид:

$$A = \begin{bmatrix} P_1(g_{1,1}, g_{1,2}) & P_2(g_{1,1}, g_{1,2}) & \dots & P_M(g_{1,1}, g_{1,2}) \\ P_1(g_{2,1}, g_{2,2}) & P_2(g_{2,1}, g_{2,2}) & \dots & P_M(g_{2,1}, g_{2,2}) \\ \vdots & \vdots & \ddots & \vdots \\ P_1(g_{k,1}, g_{k,2}) & P_2(g_{k,1}, g_{k,2}) & \dots & P_M(g_{k,1}, g_{k,2}) \\ d_{x_1}P_1(g_{1,1}, g_{1,2}) & d_{x_1}P_2(g_{1,1}, g_{1,2}) & \dots & d_{x_1}P_M(g_{1,1}, g_{1,2}) \\ d_{x_1}P_1(g_{2,1}, g_{2,2}) & d_{x_1}P_2(g_{2,1}, g_{2,2}) & \dots & d_{x_1}P_M(g_{2,1}, g_{2,2}) \\ \vdots & \vdots & \ddots & \vdots \\ d_{x_1}P_1(g_{k,1}, g_{k,2}) & d_{x_1}P_2(g_{k,1}, g_{k,2}) & \dots & d_{x_1}P_M(g_{k,1}, g_{k,2}) \\ d_{x_2}P_1(g_{1,1}, g_{1,2}) & d_{x_2}P_2(g_{1,1}, g_{1,2}) & \dots & d_{x_2}P_M(g_{1,1}, g_{1,2}) \\ d_{x_2}P_1(g_{2,1}, g_{2,2}) & d_{x_2}P_2(g_{2,1}, g_{2,2}) & \dots & d_{x_2}P_M(g_{2,1}, g_{2,2}) \\ \vdots & \vdots & \ddots & \vdots \\ d_{x_2}P_1(g_{k,1}, g_{k,2}) & d_{x_2}P_2(g_{k,1}, g_{k,2}) & \dots & d_{x_2}P_M(g_{k,1}, g_{k,2}) \end{bmatrix}$$

4.1.3 Обзор проведенных экспериментов и полученных результатов

1 Сравнение качества приближения функции [1](#) на $[-2, 2]^2$ двумя методами.

В качестве полиномов будем использовать стандартные полиномы и полиномы Чебышева, Лагранжа и Фурье.

В обоих подходах будет обучать модель линейной регрессии на 200 слу-

чайных точек, после чего измерять качество моделей на тестовой выборке.

Результирующим качеством будет среднее качество по 50 экспериментам.

Далее приведен полученный нами график зависимости качества аппроксимации от количество многочленов в разложении.

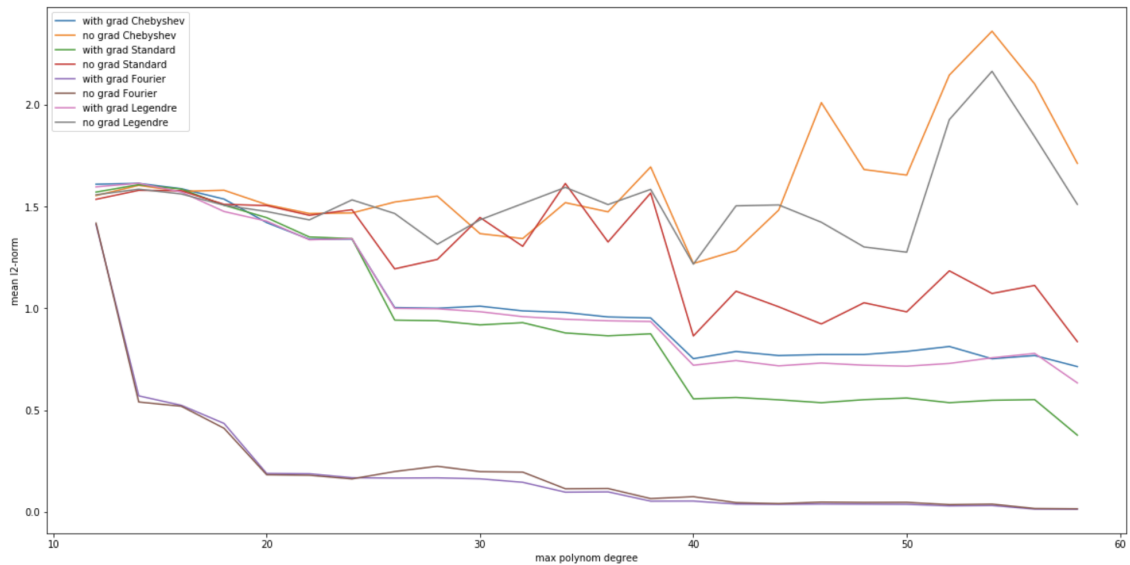


Рис. 3: Качество аппроксимации функции 1 методами с и без использования производных для различных полиномиальных разложений.

По Графику видно, что полиномы Фурье приближают функцию лучше остальных. Разница в качестве методов для полиномов Фурье практически не заметна. Для всех остальных разложений видно, что метод с производными стабильнее и показывает более хорошее качество, чем метод, не использующий информацию о производных.

4.2 Исследование метода обучения с использованием понятия глубины формулы

Над этой частью работает Михаил Зыбин

Приведенный выше пример с законом Ома является простым примером задачи, так как в нем нужно распознать пропорциональную зависимость. Конечно, мы хотим выучивать и более сложные формулы. В формулах могут

встречаться элементарные функции, например, \sin , \cos и константы, среди которых могут быть рациональные, которые можно определить точно, и иррациональные (например, π), для которых нужно использовать дополнительные идеи.

Предлагаемый подход обучения агента основывается на идее глубины формулы.

Определим формулу глубины 0 как число:

$$\phi_0(x_1, x_2, \dots, x_n) = \lambda_0$$

Далее определяем по индукции:

$$\begin{aligned} \phi_{k+1}(x_1, x_2, \dots, x_n) = & \phi_k(x_1, x_2, \dots, x_n) + \\ & + \sum_{k=1}^n [\lambda_{k+1,i,0} x_i^{\alpha_{k+1,i}} \phi_{k,i,0}(x_1, x_2, \dots, x_n) + \lambda_{k+1,i,1} \sin(\phi_{k,i,1}(x_1, x_2, \dots, x_n) + \\ & + \lambda_{k+1,i,2} \cos(\phi_{k,i,2}(x_1, x_2, \dots, x_n))]. \quad (3) \end{aligned}$$

Заметим, что при фиксированной выборке ϕ_k представляет собой дифференцируемую функцию от своих параметров $\lambda_{i,j}$, $\alpha_{i,j}$, что процесс вычисления значения ϕ_k можно представить в виде графа, аналогичного нейронной сети, в котором на m -ом слое будут вычисляться значения формул глубины m . Поэтому к этой функции можно применять back-propagation.

Цель - чтобы формула на данных давала значение, достаточно близкое к нулю. Для этого мы будем перебирать глубину и для каждой глубины будем минимизировать MSE-loss с нулевым таргетом методом back-propagation до тех пор, пока не найдем глубину, которая даст близость к нулю не больше наперед заданного threshold'a.

5 План дальнейшей работы

На данный момент мы уже реализовали базового агента, способного изучать простые законы, однако, точность изучения даже простых законов оставляет желать лучшего. Базовый агент реализован Юлией Семавиной на основе метода, предложенного Михаилом Зыбиным, и изложенным в разделе 4.2. Наши дальнейшие планы заключаются в следующем:

- 1 Настраивать данные для запросов к среде на основании процесса обучения. (*Юлия Семавина*)

Сейчас входные данные для среды всегда выбираются равномерно из фиксированного промежутка. Нужно придумать, как использовать информацию о процессе обучения для того, чтобы выбирать эти данные лучшим способом и тем самым ускорить и улучшить процесс обучения теории.

- 2 Добавление шума в среды (*Михаил Зыбин*)

В реальных физических экспериментах редко удастся получить точные данные. Поэтому для применимости наших идей на практике имеет смысл адаптировать агента для работы с шумными данными.

- 3 Провести эксперимент с добавлением информации о производных в предлагаемую модель (*Юлия Семавина*)

В разделе 4.1.2 было показано, что использование информации о производных может оказать положительное влияние на качество модели. Предлагается проверить, можно ли применить это наблюдение к задаче RoboScientist. Это стоит делать следующим образом: добавить отдельную нейросеть, задача которой не выявление закона, а, наоборот, базовая задача приближения данных - предсказания выхода среды по входу. С помощью autograd найти производные и использовать их как target для значения производной формулы текущей модели на входных

данных. Сравнить качество модели с и без производных и выбрать лучшую.

4 Добавить в модель RoboScientist память *(Юлия Семавина)*

Модель AI Physicist доказывает полезность сохранения выученных теорий и применения их для решения новых задач. Аналогичный механизм можно попробовать добавить и в RoboScientist.

5 Попробовать улучшить качество модели, предложенной в 4.2. *(Михаил Зыбин)*

Настоящая реализация нейросети не показывает хороших результатов. Нужно либо доработать модель, либо совершенно ее изменить. Варианты улучшения модели: Байесовская оптимизация, временная заморозка параметров, избавление от ненужных параметров.

6 Реализовать алгоритм Occam's Razor, предложенный в [Tailin Wu and Max Tegmark \(2019\)](#). *(Юлия Семавина)*

Алгоритм основывается на предположении о том, что простые объяснения всегда лучше сложных. Именно этот алгоритм позволяет AI Physicist изучать точные, а не приближительные теории. Условно, если нейросеть выучила, что некий параметр равен 1.9998, то очень вероятно, что в реальности этот параметр равен 2. Идея алгоритма основывается на минимизации так называемого описания формулы, идея которого состоит в том, что на описание простых выражений тратится меньшее количество ресурсов, чем на описание более сложных выражений. Конкретные определения и формулировки приводятся в [Tailin Wu and Max Tegmark \(2019\)](#).

7 Предложить способ обобщения выученных теорий на основе метода, предложенного в [Tailin Wu and Max Tegmark \(2019\)](#). *(Михаил Зыбин)*

В процессе изучения различных сред агент может встречать очень по-

хожие среды, например, закон Ома с сопротивлением 1 и закон Ома с сопротивлением 2. Хорошо бы научиться объединять эти теории в одну.

Список литературы

1. Toward an AI Physicist for unsupervised learning, Tailin Wu and Max Tegmark In Phys. Rev. (2019)
2. Function approximation using gradient information with application to parametric and stochastic differential equations, Gleb Ryzhakov and Ivan Oseledets In ArXiv (2018)
3. Journal of Artificial Intelligence Research 62, 729, K. Grace, J. Salvatier, A. Dafoe, B. Zhang, and O. Evans (2018)
4. Physicist's Journeys Through the AI World - A Topical Review There is no royal road to unsupervised learning, I. Alhousseini, W. Chemissany, F. Kleit, and A. Nasrallah In ArXiv (2019)
5. Discovering physical concepts with neural networks, Raban Iten, Tony Metger, Henrik Wilming, Lidia del Rio, and Renato Renner In Phys. Rev. (2020)