

Куда дует ветер: где ждать новых прорывов в AI?

Конспект лекции

9 июля 2025 г.

Содержание

1	Введение: Тайнство следующего шага	2
2	Смена парадигм: до и после ChatGPT	2
2.1	Парадигма до ChatGPT	2
2.2	Парадигма после ChatGPT	3
3	Проблема текущего подхода и следующий скачок	3
4	Возвращение к обучению с подкреплением (RL)	3
5	Масштабирование верифицируемых задач	4
6	Направления для будущей работы	4
6.1	Инженерия	4
6.2	Создание верифицируемых задач	4
6.3	Алгоритмы обучения	4
7	Заключение: Выбор философии	5

1 Введение: Таинство следующего шага

Вопрос о том, где ждать новых прорывов в искусственном интеллекте (AI), часто кажется странным, но он крайне важен для исследователей, чья задача — заглядывать на несколько лет вперед. Этот вопрос обострился после новостей о том, как новая модель OpenAI успешно справилась с бенчмарком ARC (Abstraction and Reasoning Corpus), который считался сложным для AI, но простым для человека.

Задачи в ARC требуют интуитивного понимания и логики. Например, раскрасить фигуры в зависимости от количества в них отверстий. Для человека это легко, но для AI — огромная трудность.

Успех OpenAI породил ощущение, что AGI (Artificial General Intelligence) почти достигнут, и в AI больше не осталось нерешенных проблем. Это чувство, будто мы достигли технологической сингулярности, возникает не впервые. Подобные моменты турбулентности и смены парадигм — ключевые. Компании вроде OpenAI и Anthropic оказались на передовой, потому что во время предыдущих сдвигов они инвестировали в методы, которые казались маргинальными. Понимание текущего ветра перемен помогает принимать верные карьерные и образовательные решения, чтобы оставаться актуальным в будущем.

2 Смена парадигм: до и после ChatGPT

Ощущение, что в AI всё достигнуто, ярко проявилось с выходом ChatGPT. Модель могла формулировать математические теоремы, писать стихи и суммировать диалоги. Это казалось магией. Чтобы понять, почему это был такой прорыв, нужно взглянуть на смену парадигм.

2.1 Парадигма до ChatGPT

До появления больших языковых моделей (LLM) доминировал подход, основанный на решении **конкретных задач**. Парадигма описывалась формулой условной вероятности:

$$p(y|x)$$

Мы брали размеченные данные (x, y) для конкретной задачи и обучали модель восстанавливать это распределение.

- **Машинный перевод:** Собирался параллельный корпус текстов на двух языках (например, переведенные книги), и на нем обучалась модель-трансформер. Задача была сложной, но реалистичной.
- **Суммаризация текста:** Формально, это та же задача, что и перевод — преобразование длинного текста (x) в короткий (y) . Однако ключевая проблема заключалась в данных. Где найти большой и разнообразный датасет из пар «длинный текст — его краткое содержание»? Заголовки новостей — лишь частный случай. Создание такого датасета для всех доменов было практически невозможным.

Из-за отсутствия данных качественные суммаризаторы до ChatGPT практически отсутствовали.

2.2 Парадигма после ChatGPT

Прорыв ChatGPT стал возможен благодаря кардинальной смене подхода. Вместо обучения на узких задачах, исследователи решили обучить одну гигантскую модель на **всех доступных данных** (условно, на всем интернете). Идея заключалась в том, чтобы выучить общее распределение данных:

$$p(x)$$

Предполагалось, что если модель поймет структуру всех текстов в интернете, она неявно научится и суммиаризации, и переводу, и многим другим задачам. Как говорил Илья Суцкевер, сооснователь OpenAI, если модель обучена предсказывать следующее слово в любом тексте, почему бы не ожидать, что она сможет сгенерировать текст, который бы написал самый умный человек на планете для решения какой-либо задачи?

Этот переход от качества «ноль» к качеству «хотя бы что-то» для задач, которые раньше были нерешаемы, и вызвал ощущение революции.

3 Проблема текущего подхода и следующий скачок

Успех LLM породил и их главную проблему. Если модель учится на существующих данных, она ограничена этими данными. В интернете есть примеры кода, переводы, схемы вязания крючком. Но там **нет решений задач тысячелетия**, например, доказательства гипотезы Римана.

Современные модели отлично справляются с автоматизацией того, что люди **уже умеют делать**. Но они не могут создавать фундаментально новое знание или решать проблемы, ответов на которые нет в обучающей выборке.

Следующий большой скачок в AI — это научить модели решать задачи, которые **мы сами решать не умеем**.

4 Возвращение к обучению с подкреплением (RL)

Искусственный интеллект уже демонстрировал способность превосходить человека в задачах, где нет готовых «правильных» данных.

- **AlphaGo (DeepMind):** Модель научилась играть в го лучше любого человека. Данных о том, как играть «сверхчеловечески», не существовало. Решение было найдено через **обучение с подкреплением (Reinforcement Learning, RL)**: две модели играли друг с другом миллионы партий. Модель, чьи действия приводили к победе, поощрялась.
- **AlphaTensor (DeepMind):** Разработал новые, более эффективные алгоритмы для перемножения матриц.

Это подводит нас к новой парадигме. Вместо гигантских датасетов нам нужны три компонента:

1. **Задачи для решения** (проблемы, которые нужно решить).
2. **Возможность оценить качество решения** (верификатор, который определит, правильно ли решена задача).
3. **Время и вычислительные ресурсы** (видеокарты и терпение).

В этой парадигме мы можем решать задачи, даже не зная правильного ответа, — нам достаточно уметь его **проверить**.

5 Масштабирование верифицируемых задач

Будущее — за созданием огромного «облака» разнообразных **верифицируемых задач**.

- **Математика:** Мы можем не знать доказательство теоремы, но мы можем проверить, является ли предложенное доказательство корректным.
- **Программирование:** Мы можем проверить, решает ли сгенерированный код поставленную задачу (проходит ли тесты).
- **Логические игры:** Шахматы, sudoku, крестики-нолики на поле 100x100. Сложность можно варьировать.

Мы наблюдаем историческую рифму. Раньше мы обучали отдельные модели на каждую задачу. Затем появилась одна большая модель (LLM), которую мы дообучаем (fine-tune) на конкретные задачи. Сейчас мы находимся на этапе, когда мы дообучаем LLM при помощи RL на отдельные верифицируемые задачи (математика, код).

Следующий логический шаг — **обучать одну модель одновременно на тысячах разнообразных верифицируемых задач**, чтобы добиться генерализации и способности решать принципиально новые проблемы.

6 Направления для будущей работы

Что нужно делать, чтобы оказаться на фронтире этого нового подхода?

6.1 Инженерия

Создание инфраструктуры для RL в таком масштабе — сложнейшая инженерная задача. Необходимо построить высоконагруженные распределенные системы, которые смогут одновременно управлять взаимодействием LLM с тысячами различных «сред»-верификаторов. Специалисты в этой области будут чрезвычайно востребованы.

6.2 Создание верифицируемых задач

Это огромное поле для исследований. Нужно создавать новые среды и задачи:

- **Доказательство теорем:** Использование языков программирования вроде **Lean** или **Coq**, где верификация доказательства сводится к компиляции программы.
- **Логические головоломки:** Разработка генераторов задач разной сложности для обучения логике.
- **Анализ сложности задач:** Понимание того, как разные типы задач (например, из разных классов сложности) влияют на обучение модели.

6.3 Алгоритмы обучения

Обучение на тысячах разных задач одновременно — это вызов. Задачи имеют разную скорость сходимости и динамику. Нужны новые **алгоритмы обучения**, которые смогут:

- Балансировать обучение на разных задачах.
- Динамически добавлять или убирать задачи из процесса обучения.
- Начинать с простых задач и постепенно переходить к сложным (curriculum learning).

7 Заключение: Выбор философии

Подход, основанный на масштабировании простых идей, часто встречает скепсис. На заре глубокого обучения его называли «грубой силой» (brute force), противопоставляя «интеллектуальным» методам. Однако именно brute force победил. Дарио Амодей, CEO Anthropic, вспоминал, что его первый успех был основан на простом наблюдении: **«просто взять больше данных и сделать модель больше — работает»**.

Сегодня мы слышим похожие голоса. Франсуа Шолле, создатель Keras и бенчмарка ARC, утверждает, что увлечение языковыми моделями отсрочило приход AGI на 10 лет.

Это рифмуется с прошлым. И перед каждым исследователем стоит философский выбор:

- Верить в сложные, «изящные» методы.
- Или верить в то, что следующий прорыв будет достигнут путем масштабирования простых и понятных подходов, таких как обучение на огромном количестве верифицируемых задач.

Именно второй путь кажется сегодня наиболее перспективным и является фокусом передовых исследований.