

Процедурные знания в предварительном обучении обеспечивают мышление в больших языковых моделях

Дата: 2025-03-06 00:00:00

Ссылка на исследование: <https://arxiv.org/pdf/2411.12580>

Рейтинг: 72

Адаптивность: 85

Ключевые выводы:

Исследование направлено на понимание того, как большие языковые модели (LLM) обучаются рассуждать на основе предобучающих данных. Основной вывод: модели используют процедурные знания из предобучающих данных для решения задач рассуждения, а не просто извлекают готовые ответы.

Объяснение метода:

Исследование показывает, что LLM используют процедурные знания для рассуждений, а не просто извлекают ответы. Это имеет высокую практическую ценность: пользователи могут получать более точные ответы через пошаговые запросы, использовать код для структурирования сложных задач и лучше понимать, с какими типами рассуждений модель справится успешно.

Ключевые аспекты исследования: 1. **Процедурные знания в предобучении:** Исследование показывает, что большие языковые модели (LLM) при выполнении задач рассуждения опираются не на конкретные ответы из предобучения, а на процедурные знания - документы, демонстрирующие методы решения похожих задач.

Механизм обобщения: Авторы выявили, что при математических рассуждениях модели синтезируют стратегии решения из документов, содержащих похожие процедуры, а не просто извлекают готовые ответы, как при фактических вопросах.

Роль кода в рассуждениях: Обнаружено, что код непропорционально сильно влияет на способность моделей выполнять математические рассуждения, представляя собой ценный источник процедурных знаний.

Корреляция влияния документов: Документы, влияющие на решение одной задачи определенного типа, часто влияют и на другие задачи того же типа, даже с разными числами, что подтверждает использование процедурных знаний.

Меньшая зависимость от отдельных документов: При рассуждениях модели меньше полагаются на отдельные документы, чем при ответах на фактические вопросы, что говорит о более обобщенной стратегии.

Дополнение: Исследование не требует дообучения или доступа к API для применения его основных выводов. Методы, которые использовали ученые (функции влияния EK-FAC), действительно требуют специальных технических возможностей, но это было необходимо только для того, чтобы проанализировать, как работают модели внутри. Сами же выводы и концепции полностью применимы в стандартном чате.

Вот основные концепции и подходы, которые можно использовать в обычном чате:

Chain-of-thought (пошаговые рассуждения) - исследование подтверждает эффективность этого метода. Пользователи могут добавлять "думай шаг за шагом" в конце запроса для повышения точности математических рассуждений.

Структурирование задач через код - исследование показывает, что код является важным источником процедурных знаний. Пользователи могут формулировать математические задачи в виде псевдокода или алгоритмических шагов.

Разбиение сложных задач на простые шаги - исследование демонстрирует, что модели полагаются на процедурные знания. Следовательно, разбиение сложных задач на последовательность простых шагов может улучшить результаты.

Запрос на объяснение методологии, а не только ответа - исследование показывает, что модели усваивают процедурные знания. Пользователи могут просить не только ответ, но и объяснение метода решения.

Понимание ограничений моделей - исследование показывает, что модели лучше справляются с задачами, для которых они видели методы решения. Это помогает пользователям формировать более реалистичные ожидания.

Применяя эти концепции, пользователи могут получить более точные ответы на математические вопросы, лучшие объяснения решений и более глубокое понимание рассуждений модели без необходимости в каком-либо специальном API или дообучении.

Анализ практической применимости: **Ключевой аспект 1: Процедурные знания в предобучении** - Прямая применимость: Пользователи могут формулировать запросы, чтобы модель продемонстрировала процедуру решения, а не просто ответ, что повышает обучающую ценность взаимодействия. - Концептуальная ценность: Понимание, что модели используют процедурные знания, помогает пользователям осознать, что модель может "объяснить" свои рассуждения через пошаговые решения. - Потенциал адаптации: Можно использовать промпты, запрашивающие пошаговые решения (chain-of-thought), для повышения точности ответов.

Ключевой аспект 2: Механизм обобщения - Прямая применимость: Пользователи могут полагаться на модель для решения новых задач определенного типа, если она видела подобные процедуры. - Концептуальная ценность: Понимание, что модель не просто "запоминает" ответы, а обобщает методы решения, говорит о её более глубоких возможностях. - Потенциал адаптации: Можно формулировать сложные задачи в виде последовательности более простых, для которых модель, вероятно, усвоила процедурные знания.

Ключевой аспект 3: Роль кода в рассуждениях - Прямая применимость: Пользователи могут использовать код как способ структурирования математических задач для получения более точных ответов. - Концептуальная ценность: Понимание, что код является важным источником процедурных знаний, объясняет, почему модели хорошо справляются с задачами, которые можно алгоритмизировать. - Потенциал адаптации: Можно формулировать математические задачи в виде псевдокода, что может улучшить понимание задачи моделью.

Ключевой аспект 4: Корреляция влияния документов - Прямая применимость: Ограниченная прямая применимость для конечных пользователей, но может помочь в понимании, почему модель дает похожие ответы на задачи одного типа. - Концептуальная ценность: Показывает, что модель действительно "понимает" структуру задачи, а не просто повторяет видимые паттерны. - Потенциал адаптации: Можно использовать для предсказания, с какими типами задач модель справится хорошо, если она уже решала похожие.

Ключевой аспект 5: Меньшая зависимость от отдельных документов - Прямая применимость: Указывает на то, что модель более надежна в задачах рассуждения, чем в фактологических вопросах, где она больше полагается на конкретные документы. - Концептуальная ценность: Помогает пользователям понять разницу между тем, как модель отвечает на фактические вопросы и как решает задачи. - Потенциал адаптации: Можно ожидать, что модель будет способна решать даже те математические задачи, которые она никогда не видела в точности такими, как они сформулированы.

Prompt:

Использование процедурных знаний в промптах для GPT ## Ключевые выводы из исследования

Исследование показало, что большие языковые модели (LLM) полагаются на **процедурные знания** при решении задач рассуждения, а не просто извлекают готовые ответы. Это означает, что модели используют обобщенные процедуры решения задач, которые они усвоили из предобучающих данных.

Как применить в промптах

Основываясь на этом исследовании, мы можем создавать более эффективные промпты, которые задействуют процедурные знания моделей:

Включать пошаговые инструкции вместо просьбы о конечном результате
Демонстрировать процесс решения на примерах **Использовать элементы кода** для структурирования рассуждений **Фокусироваться на общих процедурах**, а не конкретных примерах ## Пример промпта для решения математической задачи

[=====] Я хочу, чтобы ты решил следующую задачу на оптимизацию. Пожалуйста:

Сформулируй задачу математически Опиши общую процедуру решения таких задач
Примени эту процедуру шаг за шагом Проверь результат Используй математические обозначения, где это уместно, и объясняй каждый шаг своего рассуждения.

Задача: Найти максимальную площадь прямоугольника с периметром 100 метров.
[=====]

Почему этот промпт работает

Этот промпт эффективен, потому что:

Активирует процедурные знания — просит модель описать общую процедуру решения **Структурирует мышление** — разбивает задачу на четкие шаги **Стимулирует пошаговое рассуждение** — требует последовательного применения процедуры **Использует метакогнитивные элементы** — включает проверку результата Согласно исследованию, модели GPT будут использовать обобщенные процедурные знания из предобучения, а не искать готовый ответ для конкретной задачи, что приведет к более надежному и объяснимому решению.