

Самокорректирующее планирование задач с помощью обратного под prompting с использованием больших языковых моделей

Дата: 2025-03-10 00:00:00

Ссылка на исследование: <https://arxiv.org/pdf/2503.07317>

Рейтинг: 78

Адаптивность: 85

Ключевые выводы:

Исследование представляет новый подход к планированию задач с использованием больших языковых моделей (LLM) под названием Inverse Prompt. Основная цель - улучшить способность LLM к самокоррекции при планировании задач для роботов путем внедрения обратного запроса (inverse prompting). Главный результат: метод повысил успешность выполнения задач в среднем на 16.3% по сравнению с существующими методами планирования на основе LLM.

Объяснение метода:

Исследование предлагает высокоадаптивный метод InversePrompt, который позволяет улучшить планирование с помощью LLM через проверку логической согласованности планов. Метод использует интуитивно понятную концепцию обратных действий, не требует специальных знаний или ресурсов и может быть легко адаптирован для повседневных задач планирования.

Ключевые аспекты исследования: 1. **Метод InversePrompt** - новый подход к самокоррекции планов действий для LLM, основанный на использовании обратных действий для проверки логической согласованности планов.

Трехступенчатый процесс рассуждения - модель сначала генерирует обратное действие, затем применяет его к текущему состоянию, а затем сравнивает результат с исходным состоянием для проверки логической целостности плана.

Явное обоснование обратной связи - метод позволяет LLM давать более детальную и точную обратную связь с четкими обоснованиями при обнаружении ошибок в плане благодаря сравнению инвертированного и исходного состояний.

Самокоррекция без внешних ресурсов - подход не требует внешних валидаторов, predetermined наборов ошибок или дополнительных моделей, позволяя LLM

самостоятельно обнаруживать и исправлять ошибки.

Эффективность в реальных условиях - метод показал способность успешно обнаруживать и корректировать ошибки при выполнении задач роботом в реальном мире.

Дополнение: Исследование не требует дообучения или специального API для применения метода InversePrompt. Хотя авторы использовали API моделей GPT-4o-mini и Gemini 1.5 Flash для своих экспериментов, сам метод может быть полностью реализован в стандартном чате с любой современной LLM.

Основные концепции и подходы, которые можно применить в стандартном чате:

Структура трехступенчатого рассуждения: Можно составить промпт, который просит LLM: Сгенерировать план действий Предложить обратные действия для каждого шага плана Проверить, возвращают ли обратные действия систему в исходное состояние Выявить несоответствия и предложить исправления

Проверка через обратные действия: Этот принцип универсален и может применяться для проверки логичности любого плана. Например, если LLM предлагает последовательность действий для решения задачи, пользователь может попросить модель проверить, можно ли вернуться в исходное состояние, выполнив обратные действия.

Итеративная самокоррекция: Пользователь может организовать диалог с LLM таким образом, чтобы модель сначала предложила план, затем проверила его с помощью метода обратных действий, а затем исправила выявленные ошибки.

Подробное обоснование обратной связи: Метод позволяет получать более детальные и обоснованные объяснения ошибок в планах, что может быть полезно для понимания ограничений и возможностей LLM.

Результаты от применения этих концепций в стандартном чате: - Более точное и логически согласованное планирование - Лучшее понимание ошибок и ограничений в предложенных планах - Меньше итераций для достижения работоспособного плана - Более детальные и обоснованные объяснения от LLM

Важно отметить, что, хотя исследование сосредоточено на робототехнических задачах с использованием формализма PDDL, сам принцип проверки через обратные действия является универсальным и может быть применен к широкому спектру задач планирования в повседневной жизни, от составления расписания до планирования проектов.

Анализ практической применимости: 1. **Метод InversePrompt** - Прямая применимость: Высокая. Пользователи могут адаптировать этот метод для проверки логической согласованности планов в повседневных задачах (например, планирование маршрута, рабочего процесса, домашних дел). - Концептуальная ценность: Очень высокая. Идея использования обратных операций для проверки

правильности плана интуитивно понятна и может быть применена широким кругом пользователей. - Потенциал для адаптации: Высокий. Метод может быть упрощен для использования в обычных чатах без необходимости в специальном API или дообучении.

Трехступенчатый процесс рассуждения Прямая применимость: Средняя. Обычные пользователи могут использовать этот подход для структурирования запросов к LLM при планировании сложных задач. Концептуальная ценность: Высокая. Метод демонстрирует, как многошаговое рассуждение улучшает качество ответов LLM, что полезно для понимания принципов взаимодействия с моделями. Потенциал для адаптации: Высокий. Структуру рассуждения можно адаптировать для различных задач планирования и проверки логической согласованности.

Явное обоснование обратной связи

Прямая применимость: Высокая. Пользователи могут применять этот подход для получения более детальных и обоснованных объяснений от LLM. Концептуальная ценность: Очень высокая. Метод демонстрирует, как можно получать более качественную обратную связь от LLM через структурированные запросы. Потенциал для адаптации: Высокий. Принцип сравнения состояний для выявления ошибок может быть применен в различных контекстах.

Самокоррекция без внешних ресурсов

Прямая применимость: Высокая. Пользователи могут использовать этот подход для улучшения качества ответов LLM без необходимости в дополнительных инструментах. Концептуальная ценность: Очень высокая. Демонстрирует, что LLM могут эффективно самокорректироваться при правильной структуре запроса. Потенциал для адаптации: Средний. Требуется понимания принципов работы LLM и структуры запросов.

Эффективность в реальных условиях

Прямая применимость: Средняя. Хотя исследование фокусируется на робототехнике, принципы применимы к широкому спектру задач планирования. Концептуальная ценность: Высокая. Показывает, что методы самокоррекции могут быть эффективны в реальных сценариях с непредвиденными ситуациями. Потенциал для адаптации: Высокий. Методы обнаружения и исправления ошибок могут быть адаптированы для различных приложений.

Prompt:

Применение метода Inverse Prompt в промптах для GPT ## Основная идея исследования

Исследование представляет метод **Inverse Prompt** для улучшения планирования задач с помощью LLM. Суть подхода: генерация обратных действий для проверки логической согласованности планов и последующая самокоррекция.

Пример промпта с использованием Inverse Prompt

[=====] # Задача планирования с самопроверкой

Контекст Я работаю над планированием последовательности действий для робота на кухне.

Инструкции 1. Создай план для задачи: "Приготовить омлет из трех яиц с помидорами и сыром". 2. Для каждого шага плана создай обратное действие, которое вернет систему в состояние до этого шага. 3. Проверь, могут ли эти обратные действия в обратном порядке логически вернуть кухню в исходное состояние. 4. Если обнаружишь несоответствия, переосмысли и исправь первоначальный план. 5. Объясни свои рассуждения при обнаружении и исправлении ошибок.

Формат вывода - Первоначальный план: [список шагов] - Обратные действия: [для каждого шага] - Проверка согласованности: [анализ] - Исправленный план (если необходимо): [список шагов] - Объяснение исправлений: [рассуждение] [=====]

Как работает метод в промпте

Многоэтапное рассуждение: Промпт структурирует процесс мышления модели, заставляя её проходить через несколько этапов анализа.

Генерация обратных действий: Модель создает обратные действия для каждого шага плана, что заставляет её глубже анализировать причинно-следственные связи.

Самопроверка логической согласованности: Проверка возможности возврата в исходное состояние помогает выявить скрытые ошибки и несоответствия в плане.

Самокоррекция: При обнаружении ошибок модель переосмысливает план и вносит необходимые исправления.

Обоснование решений: Требование объяснить исправления улучшает качество рассуждений и прозрачность процесса.

Этот подход повышает точность планирования и снижает вероятность логических ошибок в сгенерированных планах, что особенно ценно для сложных многоэтапных задач.