

# Повторное исследование способности графов к рассуждению больших языковых моделей: случаи изучения в переводе, связности и кратчайшем пути

Дата: 2025-01-07 00:00:00

Ссылка на исследование: <https://arxiv.org/pdf/2408.09529>

Рейтинг: 68

Адаптивность: 75

## Ключевые выводы:

Исследование направлено на анализ способности больших языковых моделей (LLM) к рассуждениям на графах. Основная цель - понять разрыв между теоретическими возможностями LLM (которые теоретически должны справляться с задачами на графах) и их практическими неудачами. Главные результаты показывают, что на производительность LLM в задачах на графах влияют типы связности узлов, размеры графов, способы описания графов и методы именования узлов.

## Объяснение метода:

Исследование предоставляет практические рекомендации по оптимальному представлению графов в запросах к LLM: использование списков соседей вместо списков рёбер, последовательное именование узлов, включение алгоритмических подсказок. Выявленные факторы влияния могут применяться для повышения точности ответов в графовых задачах. Ограничением является узкий фокус на графовых задачах и необходимость некоторых технических знаний.

## Ключевые аспекты исследования: 1. **Комплексная оценка способностей LLM к рассуждениям на графах:** Исследование систематически анализирует, как LLM справляются с графовыми задачами (определение связности, поиск кратчайшего пути), выявляя расхождение между теоретическими возможностями и практическими результатами.

**Выявление ключевых факторов влияния:** Авторы идентифицировали факторы, влияющие на эффективность LLM в графовых задачах: тип связности узлов (K-hop, изолированные компоненты, асимметричные связи), размер графа, метод описания графа и способ именования узлов.

**Анализ различных методов представления графов:** Исследование сравнивает три способа описания графов (матрица смежности, списки соседей, списки рёбер) и

их влияние на способность LLM понимать структуру графа.

**Влияние обучения и размера модели:** Авторы демонстрируют, что увеличение размера модели и количества обучающих данных значительно улучшает способность LLM решать графовые задачи.

**Различия в процессах рассуждения:** Обнаружено, что LLM используют разные стратегии рассуждения в зависимости от способа представления графа (списки соседей vs списки рёбер).

## Дополнение:

### Применимость методов в стандартном чате

Исследование не требует дообучения или API для применения основных выводов. Большинство методов и подходов можно непосредственно применить в стандартном чате с LLM.

#### Ключевые применимые концепции:

**Оптимальное представление графов:** Использование списков соседей вместо списков рёбер для более точных результатов Последовательное именование узлов (1, 2, 3...) вместо случайных идентификаторов Использование осмысленных имён узлов вместо абстрактных идентификаторов

**Структурирование запросов:**

Учёт сложности связности при формулировании задач (разбиение сложных задач на более простые) Адаптация запросов к известным ограничениям LLM (проблемы с  $k\text{-hop} > 3$ , изолированными компонентами)

**Алгоритмические подсказки:**

Включение в промпт описания алгоритма (BFS для связности, Дейкстра для кратчайшего пути) Использование Chain-of-Thought промптинга для пошагового решения ##### Ожидаемые результаты:

- Повышение точности ответов в задачах определения связности на 20-30%
- Значительное улучшение результатов в задачах поиска кратчайшего пути
- Более последовательные и логичные рассуждения модели
- Снижение количества "галлюцинаций" при работе со структурированными данными

Важно отметить, что хотя авторы использовали специализированные методы для своих экспериментов, основные выводы исследования о влиянии формата представления, именования и алгоритмических подсказок полностью применимы в

стандартном чате без какого-либо дообучения.

**## Анализ практической применимости: 1. Выбор оптимального способа описания графа** - Прямая применимость: Пользователи могут значительно улучшить результаты, представляя графы в виде списка соседей вместо списка рёбер, особенно для сложных графов. - Концептуальная ценность: Понимание того, что LLM по-разному обрабатывают различные представления графов, помогает выбирать оптимальный формат для конкретной задачи. - Потенциал для адаптации: Пользователи могут адаптировать свои запросы, выбирая наиболее подходящий формат представления графов в зависимости от типа задачи.

**Влияние именования узлов** Прямая применимость: Использование последовательной нумерации (1, 2, 3...) вместо случайных имён улучшает понимание графов моделью. Концептуальная ценность: Осознание влияния именования на производительность LLM помогает пользователям лучше структурировать свои запросы. Потенциал для адаптации: Этот принцип можно применить к широкому спектру задач, где требуется идентификация объектов.

### **Учёт сложности связности и размера графа**

Прямая применимость: Пользователи могут предсказать снижение точности при увеличении длины пути между узлами (K-hop) и соответственно корректировать свои ожидания или запросы. Концептуальная ценность: Понимание ограничений LLM в работе с графами разной сложности помогает избегать ошибок. Потенциал для адаптации: Возможно разбиение сложных графовых задач на подзадачи с меньшим количеством шагов.

### **Применение специализированных алгоритмов**

Прямая применимость: Включение алгоритмических подсказок (BFS, Дейкстра) в промпт значительно улучшает результаты в задачах связности и поиска кратчайшего пути. Концептуальная ценность: Понимание, что явное указание алгоритма помогает LLM структурировать процесс рассуждения. Потенциал для адаптации: Этот подход можно применить к широкому спектру алгоритмических задач, не только связанных с графами.

### **Анализ процессов рассуждения LLM**

Прямая применимость: Ограниченная, требует технических знаний для применения. Концептуальная ценность: Высокая - понимание того, что LLM используют разные стратегии для разных представлений графов, помогает выбирать оптимальный формат запроса. Потенциал для адаптации: Средний - принципы могут быть применены к другим структурированным данным.

### **Prompt:**

Применение знаний о графовом рассуждении LLM в промптах **##** Ключевые выводы из исследования

Исследование показывает, что эффективность LLM при работе с графами зависит от: - Способа представления графа (список соседей работает лучше, чем список рёбер) - Длины пути между узлами (точность падает с увеличением длины) - Именования узлов (семантически значимые имена повышают точность) - Явного включения алгоритмов (например, BFS) в промпт

## Пример эффективного промпта для задачи поиска кратчайшего пути

[=====] Я опишу граф в виде списка соседей для каждого узла. Мне нужно найти кратчайший путь между двумя узлами.

Граф: - Alice: Bob, Carol, Dave - Bob: Alice, Eve - Carol: Alice, Frank - Dave: Alice, Grace - Eve: Bob, Frank - Frank: Carol, Eve, Grace - Grace: Dave, Frank

Задача: Найди кратчайший путь от Alice до Grace.

Используй алгоритм поиска в ширину (BFS): 1. Начни с узла Alice 2. Исследуй всех соседей Alice 3. Для каждого непосещенного соседа, добавь его в очередь 4. Продолжай, пока не найдешь Grace или не исчерпаешь все возможные пути 5. Запиши каждый шаг твоего рассуждения 6. В конце укажи найденный путь и его длину [=====]

## Почему это работает

**Представление графа:** Используется список соседей вместо списка рёбер, что согласно исследованию даёт лучшую производительность ( $O(|N|)$  против  $O(|E|)$ ).

**Семантические имена:** Вместо абстрактных идентификаторов (Node1, Node2) используются осмысленные имена (Alice, Bob), что улучшает понимание графа моделью.

**Явный алгоритм:** В промпт включен алгоритм BFS, что, согласно исследованию, повышает точность результатов на ~8%.

**Пошаговое рассуждение:** Запрос явно просит модель показать шаги рассуждения, что помогает отслеживать правильность пути и соответствует метрикам Fidelity (Facc) и Path Consistency Ratio (PCR) из исследования.

**Ограниченная сложность:** Граф небольшой, что соответствует выводу о том, что LLM лучше справляются с графами меньшего размера.

## Дополнительные рекомендации

- Для сложных графов разбивайте задачу на подзадачи с меньшим количеством шагов

- При необходимости работы с большими графами используйте наиболее мощные доступные модели (например, GPT-4 вместо GPT-3)
- Для критически важных задач рассмотрите возможность использования специализированных графовых алгоритмов вместо полагания только на LLM