Лекция 11.

Малые языковые модели (Small Language Models, SLM) как будущее агентного ИИ

SLM как будущее агентного ИИ

- Агентный ИИ
- SLM
- Преимущества SLM
- Альтернативный подход

Основано на статье

Small Language Models are the Future of Agentic AI

Peter Belcak, Greg Heinrich, Yonggan Fu, Xin Dong, Saurav Muralidharan, Yingyan Celine Lin, Pavlo Molchanov
NVIDIA Research

Агентный ИИ

Агентный ИИ (Agentic AI) — это искусственный интеллект, который действует как автономный агент: он не просто отвечает на запросы, а сам ставит цели, планирует, принимает решения, выполняет действия и учится на результатах, чтобы достичь заданной задачи с минимальным вмешательством человека.

Ключевые характеристики агентного ИИ:

- Автономность
- Целеполагание
- Планирование и рассуждение
- Действие
- Память и контекст
- Адаптивность

Агентный ИИ

Примеры задач для агентного ИИ:

Полный цикл исследований: «Проанализируй последние статьи о сверхпроводимости, найди противоречия, напиши сводный отчет и презентацию».

Управление бизнес-процессами: «Следи за моей почтой, выделяй срочные запросы от клиентов, согласовывай время встреч в календаре и готовь первые варианты ответов».

Сложный анализ данных: «Загрузи эти датасеты, почисти их, проведи сравнительный анализ, построй прогнозную модель и визуализируй ключевые инсайты».

Создание таких агентов на основе **гигантских моделей (LLM вроде GPT-4)** очень дорого и медленно для массового применения. **Малые модели (SLM)**, будучи меньше, быстрее и дешевле, идеально подходят для создания множества **специализированных агентов**, каждый из которых отлично выполняет свою конкретную задачу.

Агентный ИИ. Развитие

- Более половины крупных ИТ-компаний активно используют агентов ИИ, причём 21% внедрили их только в течение последнего года
- Помимо пользователей, рынки также видят значительную экономическую ценность в агентах ИИ: по состоянию на конец 2024 года сектор агентного ИИ получил более 2 млрд долларов США в виде стартапного финансирования, был оценен в 5,2 млрд долларов США и, как ожидается, вырастет почти до 200 млрд долларов США к 2034 году

Малая языковая модель (SLM)

Малая языковая модель (Small Language Model, SLM) — это языковая модель, которая может быть размещена на обычном потребительском электронном устройстве и выполнять вывод (инференс) с достаточно низкой задержкой, чтобы быть практичной при обработке агентных запросов одного пользователя.

Модели размером менее 10 млрд параметров будем считать SLM:

- *достаточно мощны*, чтобы справляться с задачами языкового моделирования в агентных приложениях;
- *более операционно пригодны* для использования в агентных системах, чем крупные модели (LLM);
- *более экономичны* для подавляющего большинства случаев использования языковых моделей в агентных системах, чем их универсальные крупные аналоги (LLM)

Малая языковая модель (SLM)

GPT-4 (большая модель):

- 1,7 трлн параметров;
- стоимость обучения — >$100 млн;
- время ответа — 2–5 сек.;
- стоимость запроса — $0,03–0,06.

Llama 3.2 3B (малая модель):

- 3 млрд параметров;
- стоимость обучения — $50–100 тыс.;
- время ответа — 0,1–10,5 сек.;
- стоимость запроса — $0,001–0,005.

Примеры преимуществ SLM

- **SLM достаточно мощны, чтобы занять место LLM в агентных системах**
- ***Серия Microsoft Phi.*** Phi-2 (2.7 млрд параметров) демонстрирует результаты в тестах на здравый смысл и генерацию кода, сопоставимые с моделями на 30 млрд параметров, при этом работая примерно в **15 раз быстрее**. Phi-3 small (7 млрд параметров) демонстрирует понимание языка и здравый смысл на уровне, сопоставимом с моделями того же поколения на 70 млрд параметров.
- ***Семейство NVIDIA Nemotron-H.*** Гибридные модели Mamba-Transformer на 2/4/8/9 млрд параметров демонстрируют точность в следовании инструкциям и генерации кода, сравнимую с LLM того же поколения на 30 млрд параметров, при **на порядок меньших** затратах FLOP.
- ***Серия Hugging Face SmolLM2.*** Каждая демонстрирует производительность в понимании языка, вызове инструментов и следовании инструкциям на уровне современных моделей на 14 млрд параметров, достигая уровня моделей на 70 млрд параметров двухлетней давности.

Примеры преимуществ SLM

- **NVIDIA Hymba-1.5B.** Эта гибридная SLM с архитектурой Mamba и механизмом внимания (attention) демонстрирует наилучшую точность следования инструкциям и **в 3.5 раза** большую пропускную способность по токенам, чем трансформерные модели сопоставимого размера.
- **Серия DeepSeek-R1-Distill.** Это семейство рассуждающих моделей размером от 1.5 до 8 млрд параметров, обученных на сэмплах, сгенерированных моделью DeepSeek-R1. Они демонстрируют выдающиеся способности к логическому и здравому рассуждению.
- **DeepMini RETRO-7.5B.** Это модель на 7.5 млрд параметров, расширенная за счет обширной внешней текстовой базы данных. Она достигает производительности, сопоставимой с GPT-3 (175 млрд параметров), в языковом моделировании, используя при этом **в 25 раз меньше параметров.**

Примеры преимуществ SLM

- **SLM более экономичны в агентных системах**
- **Эффективность инференса.** Обслуживание SLM на 7 млрд параметров обходится **в 10–30 раз дешевле**, чем обслуживание LLM на 70–175 млрд параметров.
- **Адаптивность дообучения (Fine-tuning agility).** Дообучение SLM требуют всего **несколько GPU-часов**, что позволяет добавлять, исправлять или специализировать поведение моделей за одну ночь, а не за недели.
- **Периферийное развертывание (Edge deployment).** Локальное выполнение SLM на потребительских GPU, обеспечивая работу агентов в реальном времени и офлайн с более низкой задержкой и более строгим контролем над данными.
- **Эффективность использования пространства параметров и эмбеддингов.** SLM могут быть принципиально более эффективными, поскольку меньшая доля их параметров вносит вклад в стоимость инференса без заметного влияния на результат.

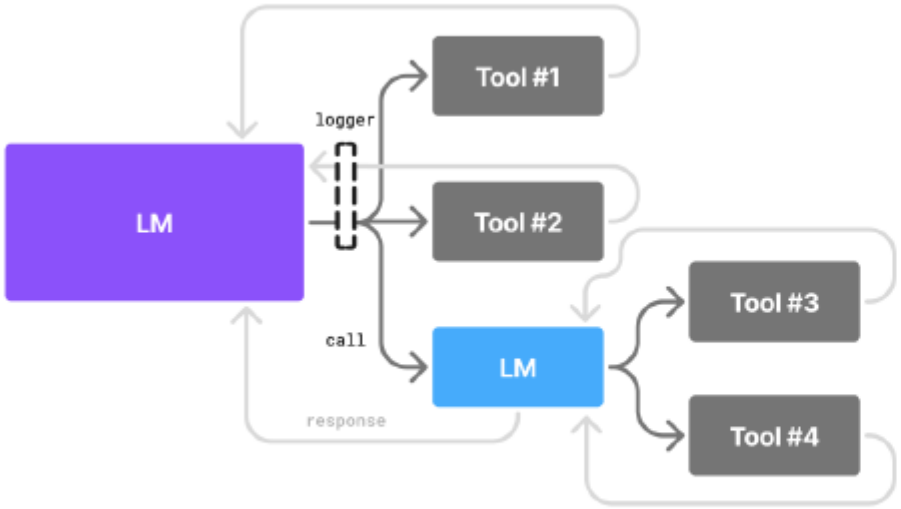
Примеры преимуществ SLM

- **SLM обладают большей операционной гибкостью по сравнению с LLM.**
- SLM по своей природе более гибки, чем их крупные аналоги, при использовании в агентных системах.
- **Демократизация создания агентов**.
- **Агенты раскрывают лишь очень узкий функционал языковой модели**
- **Агентное взаимодействие требует точного соответствия поведения**

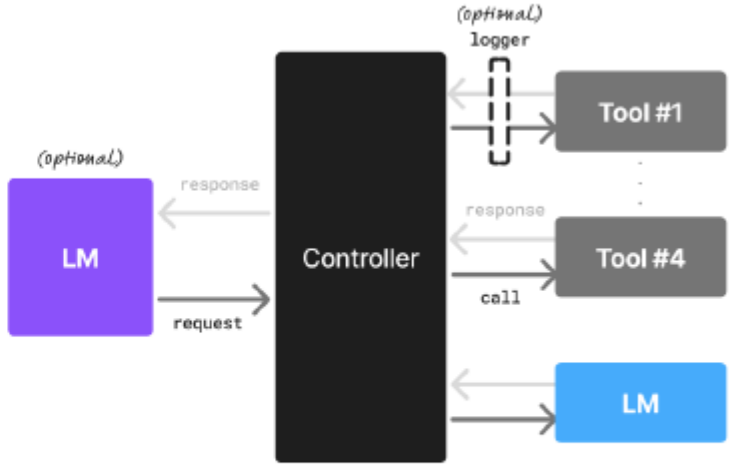
Важно, чтобы сгенерированный вызов инструмента или выходные данные строго соответствовали формату, который диктуется порядком, типизацией и природой параметров инструмента или ожиданиями кода, вызывающего модель, соответственно.

- **Гетерогенность агентных систем**
- **Агентные взаимодействия —** естественный путь для сбора данных для будущего улучшения

Модульный агентный ИИ



Example Control Flow:



Example Control Flow:

Недостатки SLM

- **Ограниченная универсальность.** SLM хорошо справляются с задачами в своих доменах, но за их пределами становятся менее эффективными.
- **Зависимость от качества данных.** Если обучающие данные плохие, модель начнет ошибаться. А в случае SLM это особенно чувствительно: даже немного «шумные» примеры могут сильно ухудшить работу.
- **Узкая база знаний.** SLM не обладают широким пониманием языка и мира вокруг нас. Это плохо в задачах, требующих более глубокого понимания различных тем и доменов.
- **Потенциальная предвзятость в конкретных доменах.** Даже при хорошей выборке SLM могут «унаследовать» предвзятости, если они присутствуют в исходных данных.

Преимущества LLM

- **Универсальность.** LLM могут справляться с задачами самого разного типа без специальной донастройки, что делает их адаптируемыми к различным приложениям.
- **Глубокое понимание языка.** Из-за широты и разнообразия обучающих данных такие модели «чувствуют язык», структуру текста и общий контекст. Это помогает им решать сложные языковые задачи.
- **Генеративные возможности.** LLM превосходно справляются с созданием креативного контента, такого как рассказы, стихи или компьютерный код.
- **Возможности дообучения.** LLM могут быть дообучены для выполнения конкретных задач или работы в определенных доменах, предлагая адаптированные ответы, которые могут быть более точными или специфичными для домена, что полезно для специализированных приложений.

Недостатки LLM

- **Ресурсоемкость.** Их нужно обучать и запускать на дорогом оборудовании с мощными GPU и большим объемом памяти. В большинстве случаев их нельзя развернуть локально — только использовать через API.
- **Проблемы предвзятости и справедливости.** LLM учатся на «всем интернете», где много предвзятых или устаревших данных. Из-за этого они могут непреднамеренно воспроизводить стереотипы.
- **Чувствительность к вводу.** LLM очень чувствительны к получаемому вводу, так называемым промптам. Небольшое изменение во входной фразе — и результат может быть совсем другим, что может повлиять на согласованность и предсказуемость их ответов.
- **Отсутствие глубокого понимания.** Несмотря на обширные знания и языковые возможности, LLM не обладают истинным пониманием мира, особенно в специфических, профессиональных темах.

LLM	SLM
Огромные сети с миллиардами параметров	Простая сеть с меньшим количеством параметров
Для обучения требуются обширные и разнообразные наборы данных	Использует меньшие по размеру и более релевантные наборы данных
На обучение модели уходит несколько месяцев	Обучение занимает несколько недель
Нужны самые передовые вычислительные мощности и ресурсы	Хватает базовых ресурсов и мощностей
Подходят для широкого круга задач, в том числе НЛП и создания творческого контента	Предназначены, в первую очередь, для простых задач в конкретной области
Низкая адаптивность: настройка требует дополнительных ресурсов	Легко адаптируются под новые потребности
Требуют специализированного оборудования или облачных сервисов	Работают на обычных ПК и даже смартфонах

Альтернативный подход

LLM будут иметь преимущество в более общем понимании языка

- Существует немало эмпирических свидетельств превосходства крупных языковых моделей в общем понимании языка над малыми моделями того же поколения. LLM приобретают свои способности к пониманию языка в соответствии с законами масштабирования. Утверждать обратное — значит противоречить законам масштабирования языковых моделей.

- Недавние исследования утверждают, что LLM обладают механизмом «семантического хаба» (semantic hub), который позволяет им обобщённым образом интегрировать и абстрагировать семантическую информацию из различных модальностей и языков.

Из этого можно сделать вывод, что LLM-генералисты всегда будут сохранять преимущество **универсально лучшей производительности** на языковых задачах, как бы узко они ни были определены, по сравнению с малыми языковыми моделями того же поколения. Это должно быть их преимуществом над SLM при развертывании в агентных приложениях.

Почему всё же SLM

- Популярные исследования законов масштабирования предполагают, что архитектура модели остается постоянной в пределах одного поколения, тогда как последние работы по обучению малых языковых моделей демонстрируют явные преимущества в производительности при использовании разных архитектур для моделей разных размеров.
- Гибкость малых языковых моделей.
- Способность к сложным рассуждениям (что традиционно было сильной стороной огромных LLM) теперь может быть достигнута и малыми моделями (SLM), если во время их работы (инференса) применить специальные техники, требующие дополнительных вычислительных ресурсов.
- Полезность предполагаемого «семантического хаба» проявляется, когда задачи или входные данные, которые должна обработать модель, являются сложными.

Препятствия для внедрения

- Значительные первоначальные инвестиции в централизованную инфраструктуру для инференса LLM.
- Использование генералистских бенчмарков при обучении, проектировании и оценке SLM.
- Недостаток популярной осведомленности.

Алгоритм конвертации LLM-агента в SLM-агента

- Организация сбора данных об использовании.
- Курирование и фильтрация данных.
- Кластеризация задач.
- Выбор SLM.
- Специализированное дообучение (Fine-tuning) SLM.
- Итерация и уточнение.