Prompt engineering

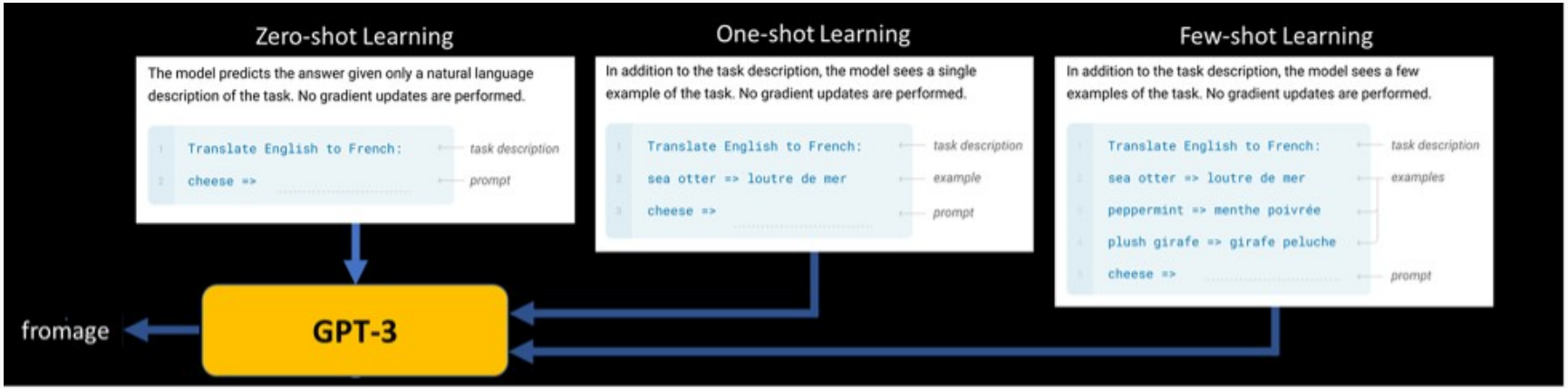
06.10.2024

Контекстное обучение

LLM обладают способностью обучаться новым задачам "на лету", не требуя явного обучения или обновления параметров. Такой способ использования LLM называется **контекстным обучением**. Он основан на предоставлении модели подходящего входного промпта, содержащего инструкции и/или примеры решения требуемой задачи.

Промпт направляет вывод модели, но модель **не меняет** свои **веса**.

Zero/One/Few–shot



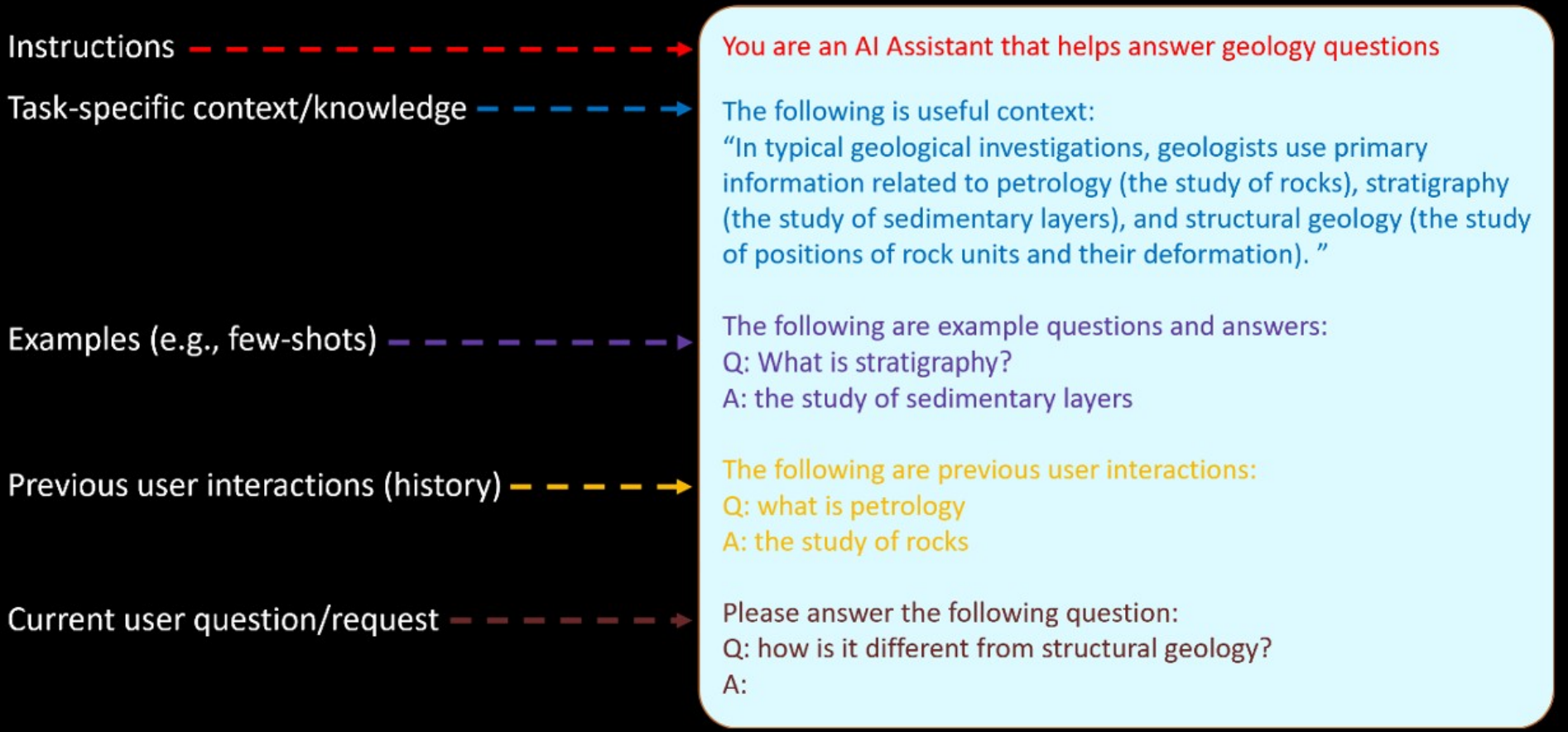
Prompt Engineering

Процесс разработки и настройки промптов на естественном языке для конкретных задач с целью повышения эффективности работы LLM называется **Prompt Engineering**.

Повышение производительности LLM

Тщательно продумывая промпты, исследователи могут **направить** внимание LLM на наиболее **релевантную** информацию для конкретной задачи, что приведет к более **точным** и **надежным** результатам.

Компоненты промпта



Используйте промпты для форматирования ответа LLM

Например, если LLM обучают генерировать инструкции к рецептам, входной текст может быть отформатирован как:

"Сначала [действие], затем [действие] и, наконец, [действие]".

Такое форматирование направляет LLM на **последовательное генерирование** инструкций.

Few–shot примеры

Few–shot подразумевают включение в LLM **нескольких примеров** ввода и вывода (пары "ввод-вывод"), чтобы ориентироваться на их содержание и формат.

Следующий пример представляет собой простую классификацию с несколькими примерами:

- *apple: fruit*
- *orange: fruit*
- *zucchini: vegetable*
- *tomato:*
- ***Complete this list***

Проблемы и ограничения

Несмотря на то, что Prompt engineering может быть полезен для повышения точности и эффективности результатов выводов LLM, он имеет существенные **проблемы** и **ограничения**.

Ограничение длины промпта

LLM имеют ограничение на количество токенов, которые могут быть использованы в качестве промпта для генерации ответа.

Сложные промпты увеличивают время ожидания и затраты ресурсов

LLM требуют времени и ресурсов для обработки и ответа на сложные запросы. Это также увеличивает **время ожидания**, которое может замедлить общий процесс разработки и развертывания модели.

Небольшие изменения промпта могут иметь большое влияние на ответ LLM

Трудно **предсказать**, как поведет себя модель при даже **небольших изменениях** промпта, что может привести к неожиданным результатам.

Это становится проблематичным в приложениях, где важны **точность** и **согласованность**, например, в автоматизированном обслуживании клиентов или медицинской диагностике.

Выделение ключевых компонентов промпта с помощью ключевых слов, символов и тегов

Вы также можете указать LLM искать контекст, представленный в определенном формате, что может помочь при разработке более сложных шаблонов промптов, например например:

Резюмируй текст, ограниченный тройными кавычками. Используйте менее 25 слов". '''<текст для обобщения>'''.

Zero-shot >>> Few-shot

Начните с Zero-shot обучения (**ZSL**), а затем с Few-shot обучения (**FSL**).

ZSL может быть полезно, когда у вас **нет** или **мало данных** для вашей задачи. **FSL** может быть полезно, когда у вас есть некоторые данные для вашей задачи или когда вы хотите проследить за поведением и качеством модели.

Перестановка элементов в промпте

Порядок расположения элементов в промпте может **повлиять** на внимание и влияние модели. Небольшие примеры или другая информация, расположенные в **нижней** части промпта, будут **сильнее влиять** на результаты выполнения, чем более ранние, расположенные в верхней части.

Спасибо за вашу работу!

Алексей

@bykov_aleksei

Кирилл

@kirilltobola