**Использование модели llama2 для разработки бота**

1. **Информация о модели.**

В ходе проверки модели ollama с подключённой моделью llama2 использовалась версия на 7 миллионов параметров (см. Рисунок 1). При тестировании запросов с таким количеством параметров не возникало проблем с пониманием контекста и распознаванием запросов на русском языке.

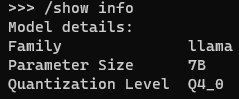


Рисунок 1 – Информация о модели.

Уровень квантования Q4\_0 означает, что каждый промпт в модели был сжат до размера 4 бит (при 16 битах без сжатия).

Во время процесса дообучения новые промпты изначально будут весить также 16 бит. Но при необходимости их можно будет также сжать до уровня Q4\_0.

1. **Шаблон входных данных, используемый в модели.**

Шаблон, по которому llama2 обрабатывает контекстное окно выглядит следующим образом:

“[INST] <<SYS>>{{ .System }}<</SYS>>

{{ .Prompt }} [/INST]”

Вместо {{ .System }} записывается инструкция для модели, чтобы она понимала, как следует реагировать на {{ .Promt }}, т. е. на сообщение пользователя. Пример, как может выглядеть такой шаблон:

“<s>[INST] <<SYS>> You are a helpful, respectful and honest assistant. Always answer as helpfully as possible, while being safe. Your answers should not include any harmful, unethical, racist, sexist, toxic, dangerous, or illegal content. Please ensure that your responses are socially unbiased and positive in nature. If a question does not make any sense, or is not factually coherent, explain why instead of answering something not correct. If you don't know the answer to a question, please don't share false information. <</SYS>> There's a llama in my garden 😱 What should I do? [/INST]”

“[INST]”, “<<SYS>>” и т. д. – параметры, которые показывают модели откуда и докуда считывать данные.

Таким образом, чтобы модель отвечала на русском, достаточно добавить в {{.System}} указание: “Говори на русском языке” (см. Рисунок 2 – Рисунок 4):

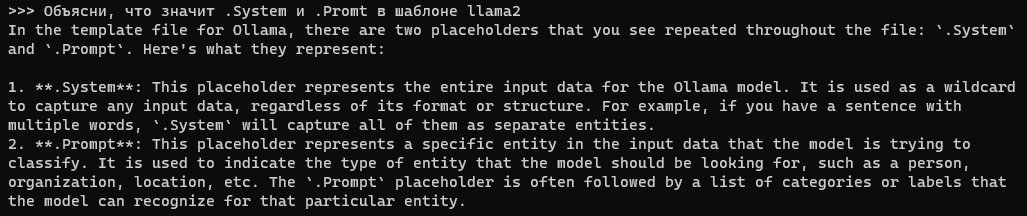


Рисунок 2 – Ответ модели без указания в System.

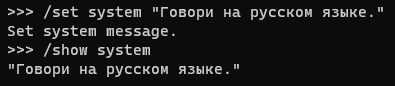


Рисунок 3 – Изменение System.

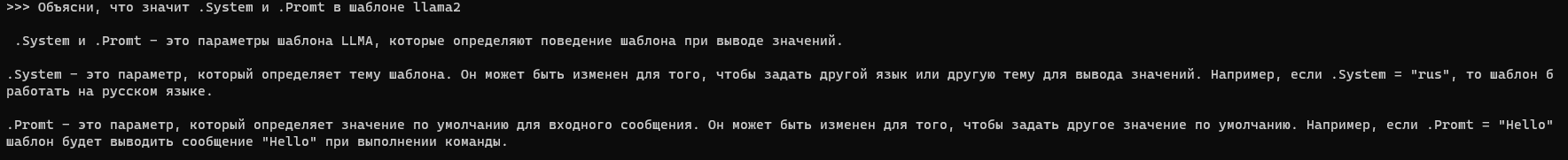


Рисунок 4 – Ответ модели после изменения.

Также это можно использовать, например, чтобы в конце ответа выводить сообщение об ограниченной ответственности бота за полученный результат (см. Рисунок 5):

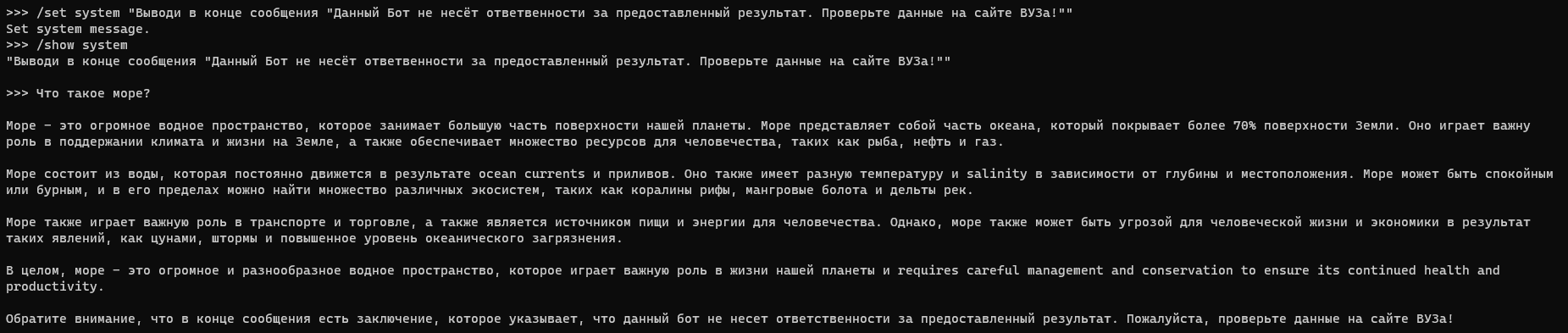


Рисунок 5 – Изменение System и пример вывода.

Кроме того, модель не умеет запоминать историю запросов в качестве отдельных промптов, поэтому весь предыдущий контекст запоминается в “текущий промт”. То есть фактически шаблон при диалоге будет выглядеть следующим образом:

<s>[INST] <<SYS>>

{{ system\_prompt }}

<</SYS>>

{{ user\_msg\_1 }} [/INST] {{ model\_answer\_1 }} </s><s>[INST] {{ user\_msg\_2 }} [/INST]

Поэтому при рассчёте пользовательских запросов надо также рассчитывать и уточняющие вопросы от ученика, а также сколько токенов использует для ответа llama2. Стоит заметить, что в ollama есть возможность ограничивать количество токенов, которые используются для генерации ответа модели, через параметры модели.

1. **Исследование размера контекстного окна.**

В llama2 контекст по умолчанию позволяет распознавать до 4096 токенов. Один токен не равен одному символу и не имеет фиксированной величины. По информации в интернете (см. https://habr.com/ru/articles/768844/)в среднем размер токена составляет 4 символа при использовании английского языка и 2 символов на русском.

Нам важен второй факт, проверим его на практике (см. Рисунок 6).

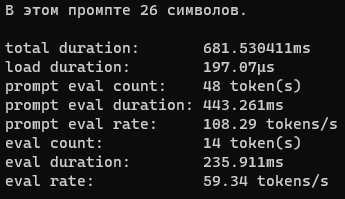


Рисунок 6 – Вывод LLM статистики.

Подключив статистику для каждого ответа с помощью команды “/set verbose” было обнаружено, что сообщение “В этом промпте 26 символов.” занимает 14 токенов, то есть 1 токен на русском рассчитывается примерно как 2 символа, следовательно, информация подтверждается.

Таким образом, максимальный контекст для llama2 составляет приблизительно 8192 символов. Это приблизительно 3 страницы текста на формата A4 с размером шрифта 14.

1. **Альтернативная модель с большим контекстом.**

Существует модель, основанная на llama2, но имеющая больший контекст. Yarn-llama2 расширяет его до 128 тысяч токенов. Таким образом, yarn-llama2 имеет возможность использовать контекстное окно в размере примерно 256 тысяч символов. Однако, обработка такого большого контекста замедлит работу модели.