Эффективное дообучение больших языковых моделей в задаче выделения информации из контекста

Вицын Семён Сергеевич

МФТИ, ФПМИ
Лаборатория машинного интеллекта
Научный руководитель Гончаров А.В.

21 мая 2025

Введение: Существующие проблемы LLM

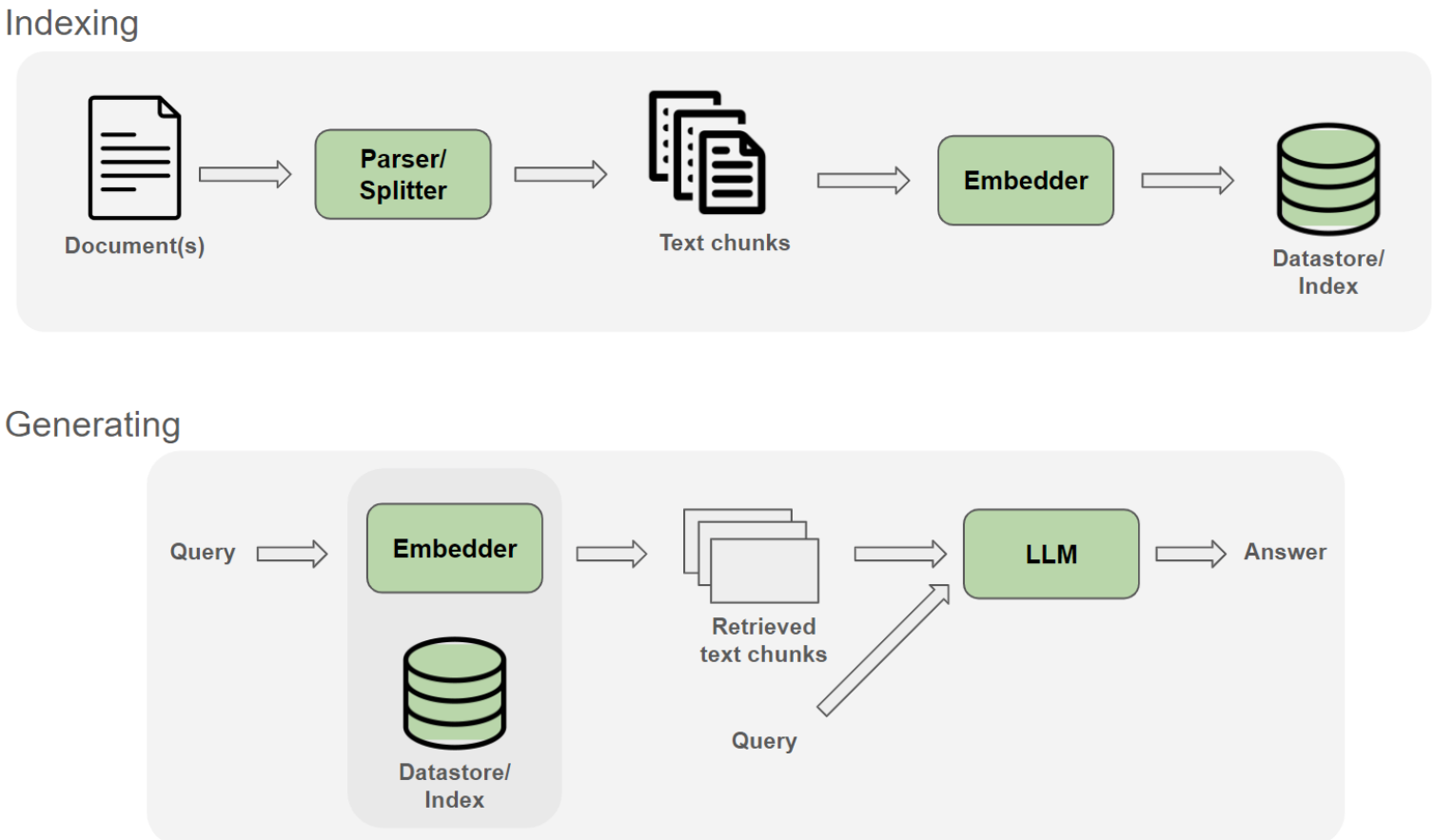
Темпы развития LLM очень высокие, однако у архитектур по-прежнему сохраняется ряд проблем.

Проблемы

- Отсутствие возможности актуализировать знания модели без постоянного дообучения.
- Дорогостоящий процесс адаптации к конкретной предметной области.
- Фактологические ошибки.

Введение: RAG

Одним из наиболее популярных способов решения указанных проблем является подход Retrieval Augmented Generation (RAG).



Введение: Существующие проблемы RAG

Однако, несмотря на высокую эффективность, и у этого подхода есть несколько известных проблем.

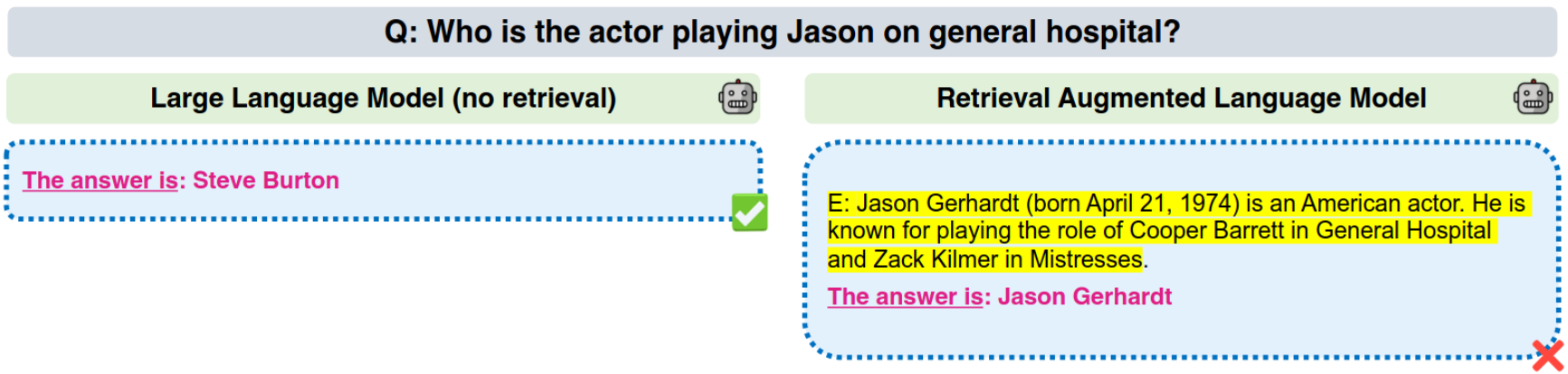
Проблемы

- Опора на нерелевантную или противоречивую информацию.
- Зависимость от положения релевантной информации в контексте.

Введение: Нерелевантный контекст

Языковые модели легко спровоцировать на неверный ответ, если у них в контексте генерации присутствует нерелевантная, отвлекающая информация. Это может стать причиной ухудшения общего качества генерации.

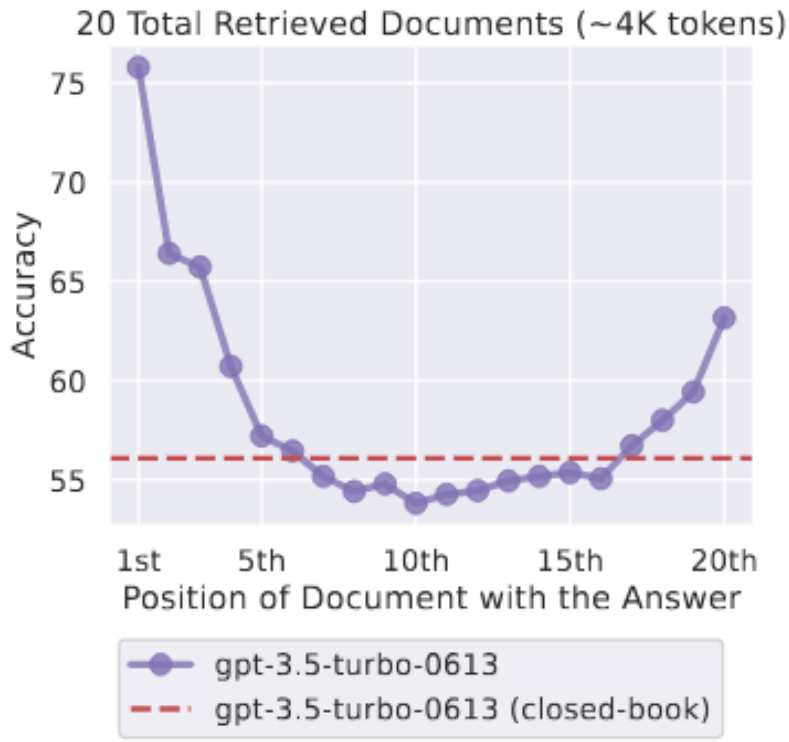
$$L(G(q, \{d^* \cup d\}), a) > L(G(q, d^*), a)$$



Введение: Lost in the middle

При генерации с большим контекстным окном возникает проблема «lost in the middle», когда модель практически не использует информацию в середине контекста.

$$L(G(q, d_{mid}), a) > L(G(q, d_{head}), a)$$
$$L(G(q, d_{mid}), a) > L(G(q, d_{tail}), a)$$



Цели и задачи исследования

Цель исследования

Построение эффективного пайплайна дообучения языковых моделей в задаче извлечения информации из контекста на примере русскоязычного RAG.

Задачи исследования

- Исследование проблем и оценка существующих методов.
- Комбинация эффективных подходов.
- Оценка качества генерации на созданном бенчмарке.

Существующие решения: RAFT

Подход Retrieval Augmented Fine–Tuning (RAFT) предлагает следующую процедуру обогащения обучающих SFT датасетов:

1. ***Обогащение контекста*** - к исходному контексту запроса добавляются фрагменты из других запросов.

$$P(a \mid q, \{d^* \cup d\}) \approx P(a \mid q, d^*)$$

2. ***Chain-of-Thought (CoT)*** - ответы аннотируются с явным выделением логических цепочек рассуждений.

$$P(a \mid x) \approx P(r_1 \mid x) \cdot P(r_2 \mid x, r_1) \cdot \ldots \cdot P(a_{final} \mid x, r_1, \ldots, r_n)$$

3. ***Добавление негативных примеров*** - в обучающий набор включаются запросы без релевантной информации, где ответом является отказ.

$$P([REFUSE] \mid q, d) \to 1, \textit{ если } Relevance(q, d) \approx 0$$

Существующие решения: Синтетические данные

«From Artificial Needles to Real Haystacks: Improving Retrieval Capabilities in LLMs by Finetuning on Synthetic Data» предлагает дообучение на синтетических словарях с поиском по ключу.

$$P(a \mid q, d_{head}) \approx P(a \mid q, d_{mid}) \approx P(a \mid q, d_{tail})$$

Multi-subkey dictionary key-value retrieval

```
Do a task using the list of dictionaries below.

Dictionary [1] {(141, 986, 163): 2528, (726, 947, 349, 820): 4130}
Dictionary [2] {(555, 710, 424): 5756, (623, 141, 997): 1633, (957, 634, 969): 7871}
...
Dictionary [6] {(645, 417, 847): 6409, (141, 623, 616): 5617}
...
Dictionary [49] {(710, 105, 141, 799): 5369, (623, 210, 477): 8971, (899, 126, 999):
4409}

Above is a list of dictionaries such that each key is a tuple of integers and each value
is an integer. Report the key that contains the integers 616, 141, 623 (not necessarily
in order), its value, and the dictionary it is in.
```

```
Desired answer: The key that contains the integers 616, 141, 623 is (141, 623, 616).
Its value is 5617 and it is in Dictionary [6].
```

Способы оценки качества

Был собран RAG-бенчмарк, содержащий следующие домены:

- **Научные статьи и регламенты**
- Техническая документация
- Финансовые и аналитические отчёты

И следующую типизацию вопросов:

1. Simple
2. With errors
3. Trash
4. Reformulation
5. Incorrect by design
6. Table
7. No Info
8. Double
9. Multi Block
10. Logical Thinking

Конфигурация

Данные и методы

- WebGLM-QA в качестве основного датасета.
- Обогащение WebGLM-QA под RAFT: 3 отвлекающих документа, 10% негативных примеров.
- Дообучение через LoRA адаптеры.

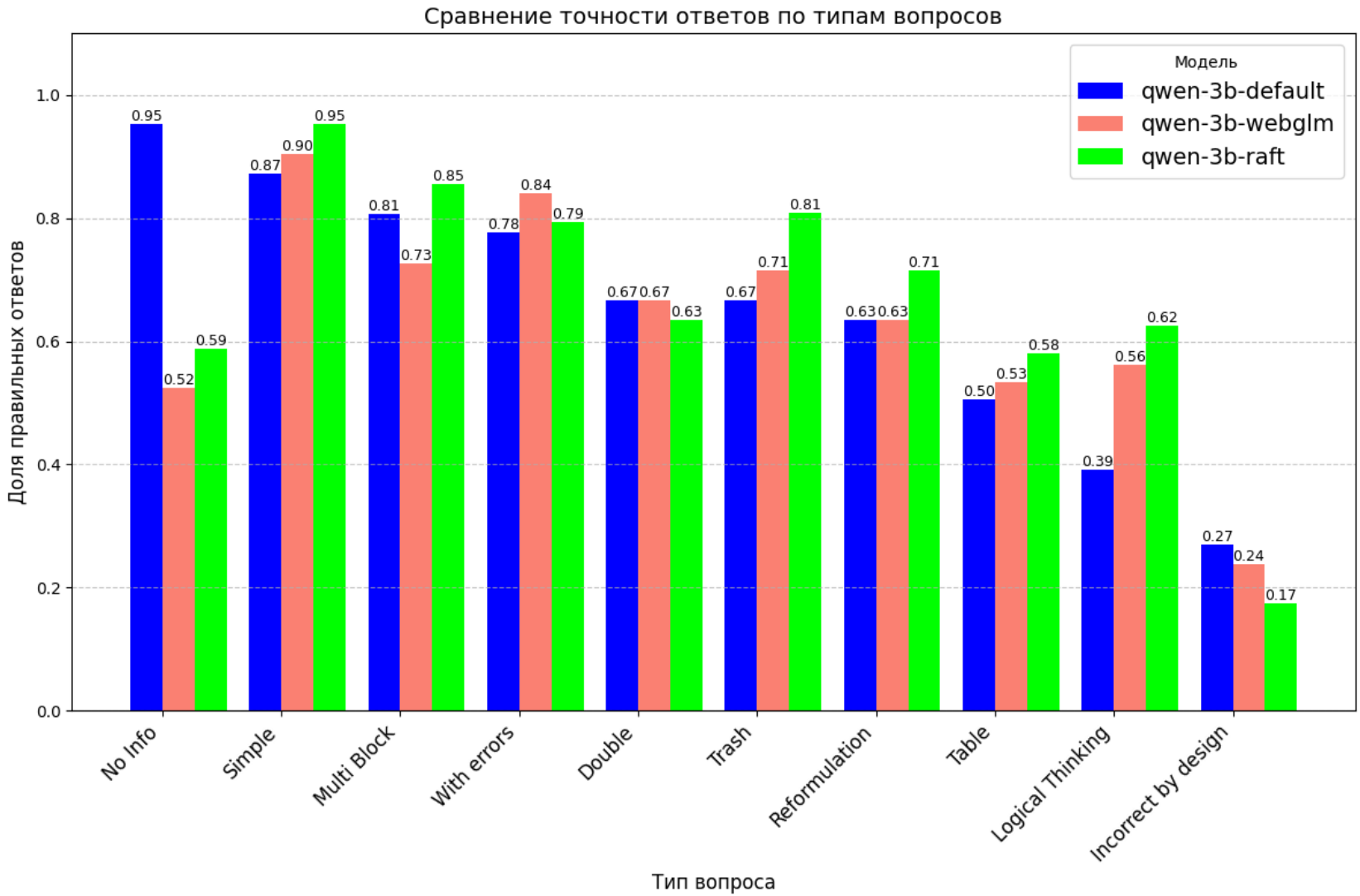
Метрики

- LLM-as-Judge — score (1-5), correct (0-1).
- ROUGE-L — precision, recall и f1.

$$\textit{ROUGE-L Precision} = \frac{\textit{LCS}(G, R)}{\textit{Количество униграм в } G}$$

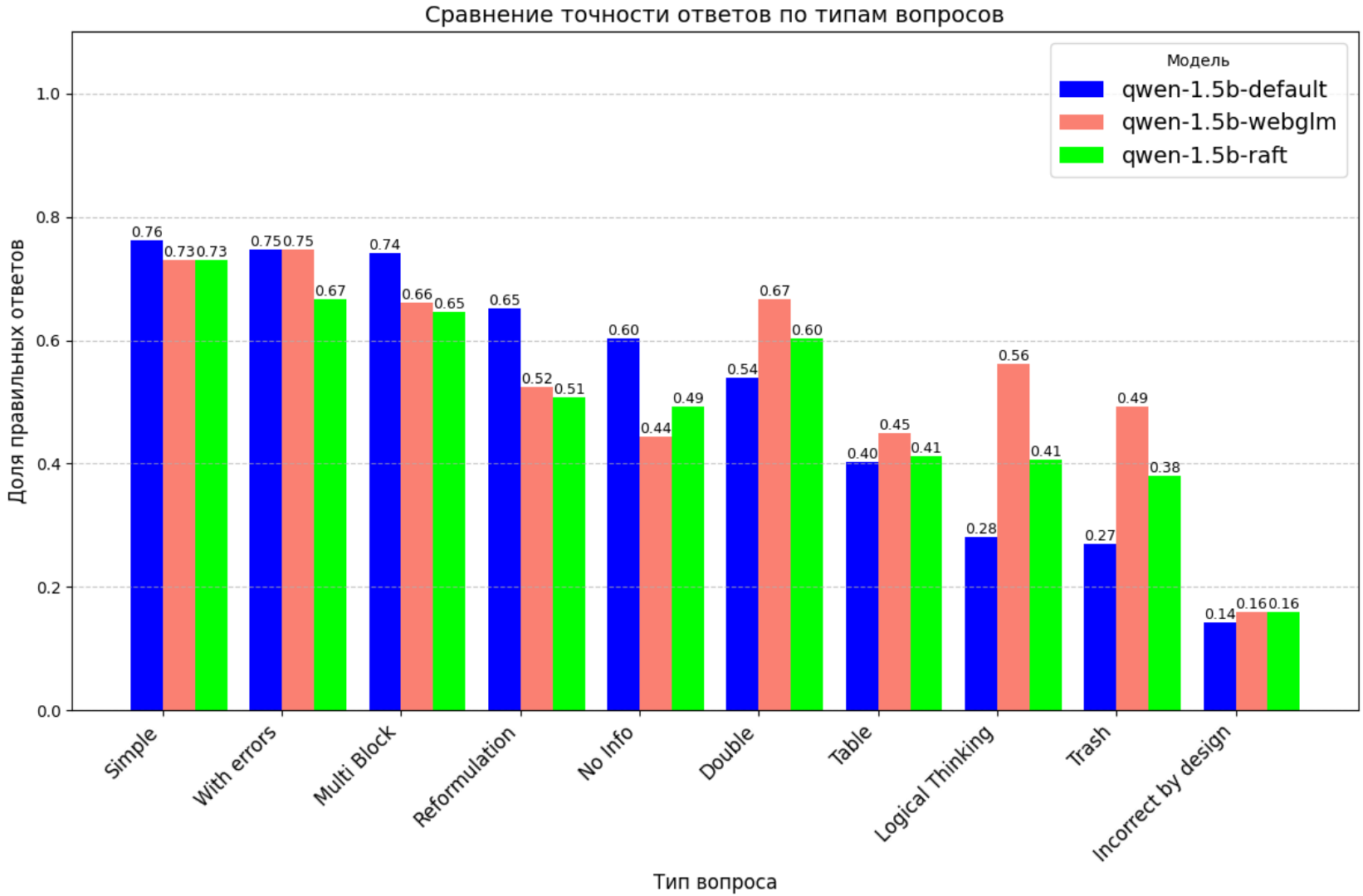
$$\textit{ROUGE-L Recall} = \frac{\textit{LCS}(G, R)}{\textit{Количество униграм в } R}$$

Результаты: RAFT 3b



Доля отказов: Default - 30% (70% неуместно), WebGLM - 9% (49% неуместно), RAFT - 8% (36% неуместно).

Результаты: RAFT 1.5b



Сводные результаты: LLM-as-Judge

Таблица: *Оценки LLM.*

Model	AVG Score	Accuracy	Irrelevant Refuse
Qwen2.5-1.5b-default	3.31	0.50	0.18
Qwen2.5-1.5b-WebGLM	**3.37**	**0.53**	**0.06**
Qwen2.5-1.5b-RAFT	3.32	0.50	0.15
Qwen2.5-3b-default	3.86	0.64	0.21
Qwen2.5-3b-WebGLM	3.73	0.63	0.05
Qwen2.5-3b-RAFT	**3.86**	**0.67**	**0.03**
Qwen2.5-32b-default	4.28	0.77	0.12

Сводные результаты: ROUGE-L

Таблица: *ROUGE-L метрики.*

Model	Precision	Recall	F1
Qwen2.5-1.5b-default	0.18	0.34	0.20
Qwen2.5-1.5b-WebGLM	**0.19**	0.36	**0.22**
Qwen2.5-1.5b-RAFT	0.14	**0.40**	0.19
Qwen2.5-3b-default	0.18	0.40	0.23
Qwen2.5-3b-WebGLM	**0.23**	0.41	**0.27**
Qwen2.5-3b-RAFT	0.14	**0.51**	0.20
Qwen2.5-32b-default	0.36	0.53	0.40

Оценка влияния положения документов в контексте

Таблица: *Качество Qwen2.5-3b-default при разных положениях документов, метрика - среднее значение accuracy.*

Documents	**Default order**	**Reverse order**	**Random**
Top-5 (2k context)	**0.68**	0.64	**0.68**
Top-10 (4k context)	**0.72**	0.71	0.68
Top-20 (8k context)	0.68	0.68	**0.71**

- На таком размере контекста у современных моделей нет явно выраженного эффекта «lost in the middle».
- Количество документов имеет слабое влияние при достаточно качественной retriever модели.
(e5-large: *recall*@1 > 0.7 и *recall*@10 > 0.9)

Другие эксперименты

Также были проведены эксперименты с дополнительными этапами обучения:

- ***Предварительный этап русификации*** — дообучение на русскоязычном QA датасете Saiga scored. Этот этап приводил к ухудшению итогового качества на 5%. В этом нет необходимости для современных мультиязычных моделей.
- ***Финальное дообучение на синтетике*** — не привело к существенных изменениям, т.к. для 1.5B задача была слишком сложной, а для 3B слишком простой. Учитывая предыдущие результаты, этап можно считать неактуальным.

Выводы

- Модели размером 1.5B и 3B можно успешно адаптировать к задаче RAG, хотя для этого и подходят разные методы.
- Проблема «lost in the middle» не возникает в сценариях RAG для современных моделей.
- Несмотря на фактическую возможность использовать небольшие языковые модели в качестве генераторов в RAG–системах, применять их следует только после адаптации под конкретный домен.