

RusHallu-RAG: КОМПЛЕКСНЫЙ ПОДХОД К ОБНАРУЖЕНИЮ ГАЛЛЮЦИНАЦИЙ В РУССКОЯЗЫЧНЫХ RAG-СИСТЕМАХ

Ф. А. Садковский¹ [0009-0006-4502-9466], Р. Р. Насырова² [0009-0008-6280-0667],

А. А. Сорокин³ [0000-0003-3877-4223]

^{1–3}Московский государственный университет имени М. В. Ломоносова,
Исследовательский Центр в сфере ИИ, г. Москва, Россия

³ООО «Яндекс», г. Москва, Россия

¹sadkovskiyfa@my.msu.ru, ²NasyrovaRR_01@my.msu.ru, ³sorokinaa@my.msu.ru

Аннотация

Представлен RusHallu-RAG – первый комплексный бенчмарк для обнаружения галлюцинаций в русскоязычных RAG-системах. Набор данных из 1000 пар «запрос – ответ» охватывает области общего знания и научных текстов с контролируемой балансировкой релевантности контекста. Предложена детализированная таксономия из шести типов галлюцинаций, использованная для разметки на уровне всего ответа и отдельных его фрагментов с привлечением экспертов и больших языковых моделей.

Для обучения детектора собран сбалансированный тренировочный датасет объемом 3000 примеров с применением синтетических методов генерации. На его основе методом SFT обучен детектор на базе YandexGPT-5-Lite-8B. Проведена оценка 15 моделей с открытыми и закрытыми весами, а также обученного детектора. Результаты показали, что число параметров не гарантирует высокой производительности: модели среднего размера (20–33 млрд) иногда превосходят более крупные. Обученный детектор достигает средней точности 63.7%, занимая второе место после закрытой Gemini-2.5-Pro, с приростом 47.2 п.п. относительно исходной модели. Разрыв между закрытыми и открытыми моделями сокращается с 20–35 до 8.4 п.п., что подтверждает эффективность подхода. Бенчмарк, код и обученная модель опубликованы в открытом доступе.

Ключевые слова: обнаружение галлюцинаций, дополненная поиском генерация, большие языковые модели, бенчмаркинг, обработка естественного

языка, вопросно-ответные системы, оценка моделей, дообучение с учителем, таксономия галлюцинаций.

ВВЕДЕНИЕ

Обнаружение галлюцинаций является центральной проблемой при разработке надежных RAG-систем на основе больших языковых моделей (Large Language Models, LLM). В таких системах галлюцинации проявляются как фрагменты ответа, не соответствующие извлеченному контексту, – противоречащие ему, содержащие неподтвержденную информацию или отклоняющиеся от него.

Для английского языка существует множество бенчмарков для обнаружения галлюцинаций [1–7], в том числе специализированных для RAG [1, 5]. Для русского языка подобные ресурсы отсутствуют. Существующие русскоязычные RAG-бенчмарки [8] опираются на поверхностные метрики (ROUGE) или LLM-as-a-judge, но не содержат детальной разметки галлюцинаций. Этот пробел препятствует разработке и систематической оценке детекторов галлюцинаций для RAG-систем на русском языке.

В настоящей работе представлен RusHallu-RAG – первый комплексный бенчмарк для обнаружения галлюцинаций в русскоязычных RAG-системах. Бенчмарк включает набор данных из 1000 пар «запрос – ответ» из областей общего знания и научных текстов с контролируемой балансировкой релевантности контекста. Предложена новая таксономия из шести классов галлюцинаций, специфических для RAG, выполнена разметка на уровне ответов и отдельных фрагментов.

На основе бенчмарка проведена оценка 15 LLM с открытыми и закрытыми весами в роли детекторов галлюцинаций. Результаты показали, что рост числа параметров не гарантирует повышения качества, а между закрытыми и открытыми моделями для русского языка сохраняется значительный разрыв.

Кроме того, для решения прикладной задачи обучения детектора галлюцинаций собран тренировочный набор из 3000 примеров, сбалансированный с помощью синтетических методов. На этих данных методом SFT обучен детектор на базе YandexGPT-5-Lite-8B¹. Обученная модель достигает средней точности

¹ <https://huggingface.co/yandex/YandexGPT-5-Lite-8B-instruct>

63.7%, занимая второе место после Gemini-2.5-Pro, с приростом 47.2 п.п. относительно исходной модели. Разрыв между закрытыми и открытыми моделями сокращается с 20–35 до 8.4 п.п., что подтверждает эффективность подхода. Бенчмарк, код и модель публикуются в открытом доступе².

ОБЗОР ЛИТЕРАТУРЫ

Существующие англоязычные ресурсы для оценки галлюцинаций в LLM можно разделить на три категории. Первая группа – общие бенчмарки для оценки RAG-систем [9–12], которые не фокусируются специально на обнаружении галлюцинаций, рассматривая любой неверный ответ как галлюцинацию без детальной разметки. Вторая, более широко представленная группа, интегрирует метрики, специфичные для галлюцинаций, в оценочные фреймворки, такие как RAGAS [13] и аналогичные [14–16], которые измеряют точность следования фактам (faithfulness) путем декомпозиции ответов на атомарные утверждения. Однако эти методы не идентифицируют конкретные фрагменты текста, содержащие галлюцинации.

Третья группа включает бенчмарки, разработанные специально для обнаружения галлюцинаций: HaluEval [4] с синтетическими примерами, HaluEval-Wild [6], включающий примеры из реальных диалогов, RAGTruth [5] с детальной разметкой на уровне ответов и фрагментов, а также FaithBench [17] и FaithJudge [18] с таксономиями по степени критичности искажений. Все перечисленные ресурсы являются исключительно англоязычными.

Для русского языка подобные ресурсы остаются малочисленными. Бенчмарк DRAGON [8] представляет собой динамически пополняющийся RAG-бенчмарк, построенный на новостных корпусах, однако его оценка опирается исключительно на метрики ROUGE и элементарные сравнения с привлечением LLM в роли судьи. Российский нормативно-корпоративный RAG-бенчмарк (RRNCB)³ ориентирован на обработку корпоративных документов и характеризуется сходными ограничениями. Ни один из этих ресурсов не предоставляет детальной разметки галлюцинаций.

² <https://github.com/feudor2/RusHallu-RAG>

³ <https://fractalagents.ai/rrncb-rag-benchmark/>

ДАННЫЕ

Существующие русскоязычные RAG-бенчмарки ориентированы на новостные и корпоративные области. Настоящая работа восполняет этот пробел, сосредоточившись на документах из области общего знания и научных текстах.

Контрольная выборка была сформирована из двух источников. Из тестовой части SberQuAD [19] – датасета оценки понимания прочитанного на русском языке, взято 655 запросов из области общего знания. Для научной сферы из бенчмарка ruSciBench [20], представляющего собой набор научных аннотаций с запросами, сгенерированными LLM по методологии AIR-Bench [21], отобраны 345 запросов. Итоговый тестовый набор составил 1000 пар «запрос – ответ».

Для обучения детектора галлюцинаций сформирован тренировочный датасет объемом 3000 примеров. Как и для контрольной выборки, использованы два источника – ruSciBench и SberQuAD. В первом случае основу составили примеры из тестового набора AIR-Bench (1374 примера), для которых эталонные релевантные документы определены с помощью пайплайна на основе модели BGE-M3 [22] (порог реранкера 0.85). Оставшиеся 1626 примеров взяты из тестовой подвыборки SberQuAD, не использованной при формировании контрольной выборки.

ПАЙПЛАЙН RAG

Разработанный пайплайн RAG состоит из двух этапов: поиска документов и генерации ответа.

Поиск и формирование контекста

RusHalluRAG интегрирован с фреймворком RusBeIR [23] – платформой для оценки качества информационного поиска на русском языке. Модели BGE-M3 [22] используются как в качестве ретривера, так и в качестве реранкера. После ранжирования для формирования контекста при каждом запросе извлекаются пять наиболее релевантных фрагментов.

Для повышения разнообразия контекста и введения контролируемого уровня сложности применяется этап балансировки. Набор данных разбивается на шесть групп в зависимости от позиции первого релевантного документа (от 1 до 5 или отсутствие). Меньшие группы дополняются путем переноса примеров

из группы с позицией 1 на целевую позицию k и замены прочих релевантных документов до k на наиболее похожие нерелевантные фрагменты, подобранные с помощью BM-25. Это повышает сложность обработки контекста, поскольку релевантные документы заменяются на похожие по форме, но не совпадающие по содержанию. Описанная процедура применялась как для контрольной, так и для тренировочной выборок.

Генерация ответа

В качестве генератора ответа выбрана модель YandexGPT-5-Lite-8B-instruct с 8 млрд параметров. Вывод модели выполняется с использованием библиотеки vllm [24]. Наборы данных, включающие пять наиболее релевантных документов и соответствующие им сгенерированные ответы, публикуются в открытом доступе^{4, 5, 6}.

РАЗМЕТКА

Классы галлюцинаций

Предлагается таксономия из шести классов, ориентированная на ответы, не соответствующие контексту:

1) **противоречие** (Contradiction): ответ прямо противоречит информации в релевантном контексте (ср. Conflict в [5]);

2) **неподтвержденная деталь** (Unconfirmed): ответ содержит информацию, не имеющую подкрепления в релевантном контексте (ср. Baseless в [5]);

3) **выдуманный ответ** (Excess): ответ целиком построен на нерелевантных документах;

4) **подмена вопроса** (Missing): ответ не отвечает на поставленный вопрос, что приводит к бессвязности (ср. [2]);

5) **неполный ответ** (Partial): ответ содержит неполную информацию без указания на ее неполноту;

6) **ложный отказ** (Oversight): модель ошибочно отказывается отвечать при наличии релевантного контекста.

⁴ <https://huggingface.co/datasets/feudor2/sberquad-rag>

⁵ <https://huggingface.co/datasets/feudor2/ruscibench-rag>

⁶ <https://huggingface.co/datasets/feudor2/RuHallu-RAG-train>

Для иллюстрации рассмотрим характерные примеры из контрольной выборки.

При запросе о факторах, подталкивающих врачей к применению нового лекарства, модель включает устную коммуникацию в число движущих факторов, тогда как контекст прямо указывает, что она оказывала тормозящее воздействие – это противоречие.

При запросе о педагогических взглядах Б. Рассела модель утверждает, что он выступал против религиозного образования, однако контекст упоминает лишь сходство его взглядов со взглядами Р. Оуэна и Ш. Фурье, которые выступали против, но не самого Рассела – это неподтвержденная деталь.

При запросе о преимуществах модифицированной системы остаточных классов для параллельной арифметики модель дает развернутый ответ на основе документа, который вообще не упоминает эту систему, а другие документы нерелевантны – это выдуманный ответ.

При запросе о крупнейших транснациональных корпорациях модель перечисляет только Pepsi и Coca-Cola, хотя контекст содержит еще несколько компаний (Нестле, Новартис, АББ) – это неполный ответ.

При запросе о вовлечении студентов в физический практикум модель выдает развернутый ответ о занятиях физической культурой и спортом, ошибочно связав запрос с этой темой вместо физики – это подмена вопроса.

При запросе определения геоэкономики модель утверждает, что в документах нет прямого определения, хотя контекст описывает геоэкономiku как формирование единой мировой сетевой рыночной экономики – это ложный отказ.

Процесс разметки

В разметке контрольной выборки участвовали два эксперта и модель Google Gemini-2.5-Pro. Каждый пример независимо размечен человеком и LLM; разногласия разрешались при участии второго эксперта. Разметчики определяли наличие галлюцинации, классифицировали ее тип(ы) и выделяли соответствующие фрагменты текста, следуя блок-схеме на рис. 1. Разметка выполнялась в Label Studio [25].

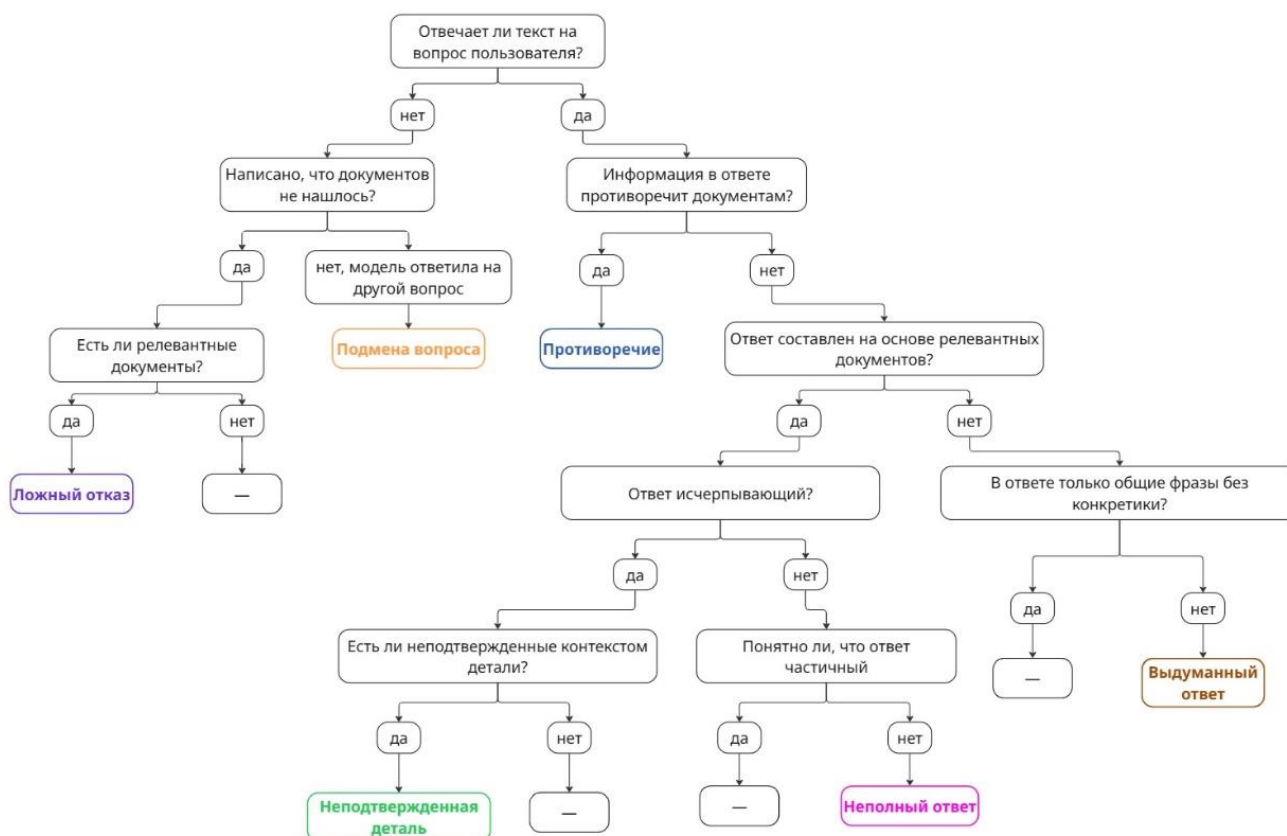


Рис. 1. Блок-схема с алгоритмом классификации для разметчиков.

Статистика

Согласованность между разметчиками составила 0.69–0.76 по коэффициенту альфа Кrippendorфа. Распределение ответов представлено на рис. 2 и 3. Доля ответов без галлюцинаций ($\approx 66\%$) отражает эффективность пайплайна RAG. ruSciBench существенно сложнее SberQuAD: доля ответов с галлюцинациями составляет 51.6% против 23.7%.

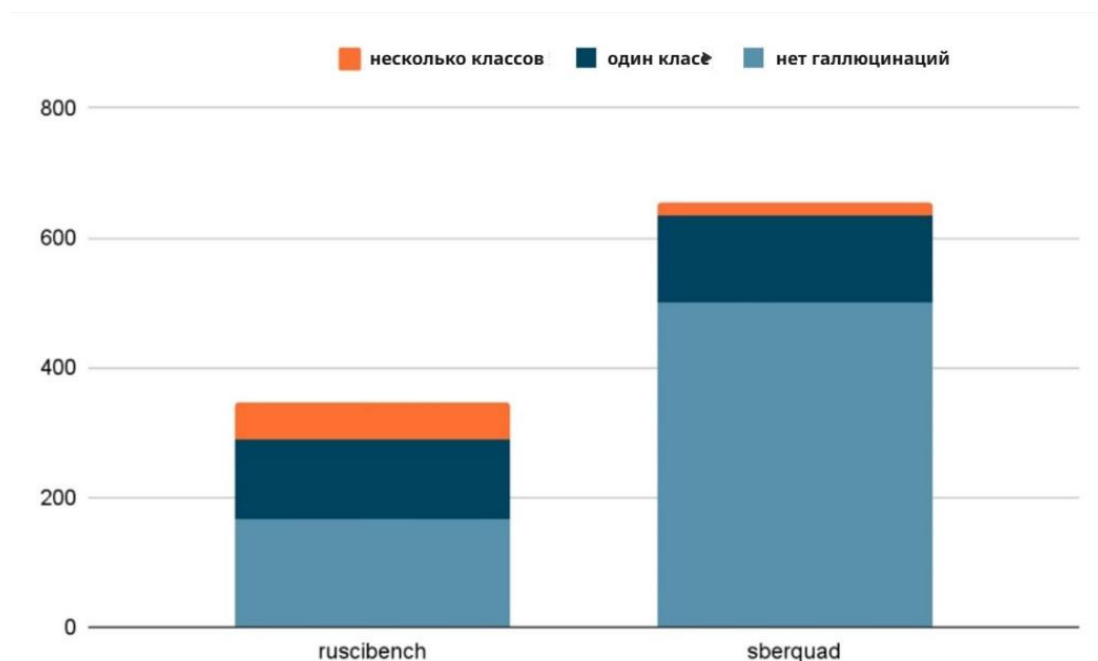


Рис. 2. Распределение ответов без галлюцинаций, ответов с одной галлюцинацией и ответов с несколькими галлюцинациями в контрольной подвыборке.

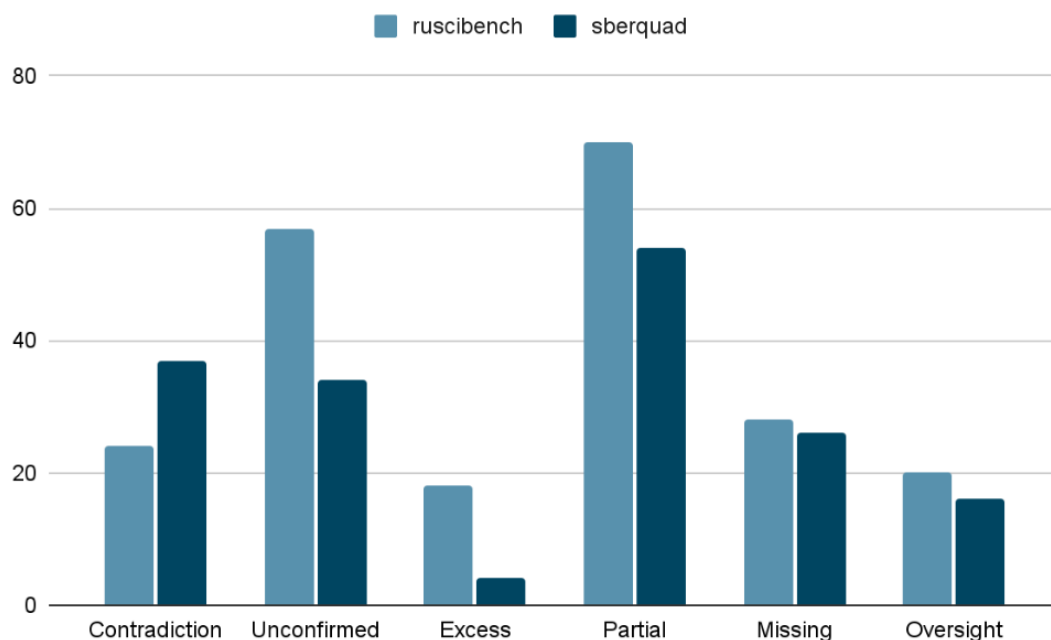


Рис. 3. Распределение галлюцинаций по каждому типу в контрольной подвыборке.

Для тренировочного набора (3000 примеров) разметка выполнена автоматически с использованием Gemini-2.5-Pro. Исходное распределение было несбалансированно: 71% примеров без галлюцинаций, среди галлюцинаций преобладают «Неподтвержденная деталь» (19.4%) и «Противоречие» (4.6%), остальные классы – менее 2.2%. Доля примеров с несколькими галлюцинациями составила 5.6%. Для устранения дисбаланса была разработана процедура синтетической балансировки, которая позволила поднять число примеров всех классов галлюцинаций до 200 за счет более представленных классов.

МЕТРИКИ

Качество обнаружения галлюцинаций оценивается на трех уровнях детализации, соответствующих схеме разметки: на уровнях бинарной классификации (наличие или отсутствие галлюцинации), многоклассовой классификации и выделения фрагментов, содержащих галлюцинации.

Для бинарной классификации использованы *точность* (Accuracy), *макро-усредненные точность* (Precision), *полнота* (Recall) и *F1-мера*. Для многоклассовой классификации между наборами меток вычислялись *точность совпадения* (Exact Match Accuracy), *расстояние Хэмминга* (Hamming Loss) и *коэффициент Жаккара* (Jaccard Similarity). Для оценки выделения фрагментов применялись *ROUGE-L* [26] для оценки пересечения на уровне токенов, а также *точность выделения фрагментов* (Span Accuracy), *коэффициент Жаккара* и *расстояние Хэмминга* для множеств фрагментов. Вычислялся также коэффициент Success Rate – доля ответов с корректным форматированием JSON.

МОДЕЛИ

Выбор базовых моделей

Набор базовых моделей включает модели, перечисленные в табл. 1.

Табл. 1. Базовые модели и их параметры.

Модель	Число параметров	Доступность
YandexGPT-5-Lite-8B-instruct ⁷	8 млрд	открытые веса
Qwen3-8B ⁸	8 млрд	открытые веса

⁷ <https://huggingface.co/yandex/YandexGPT-5-Lite-8B-instruct>

⁸ <https://huggingface.co/Qwen/Qwen3-8B>

T-lite-it-2.1 ⁹	8 млрд	открытые веса
gpt-oss-20b ¹⁰	20 млрд	открытые веса
gemma-3-27b-it ¹¹	27 млрд	открытые веса
T-pro-it-2.0 ¹²	33 млрд	открытые веса
Llama-3.3-70B-Instruct ¹³	70 млрд	открытые веса
glm-4.6v ¹⁴	106 млрд	открытые веса
Qwen3-235B-A22B-Instruct-2507 ¹⁵	235 млрд (22 млрд активных)	открытые веса
DeepSeek-v3-0324 ¹⁶	685 млрд	открытые веса
kimik2-thinking ¹⁷	1.1 трлн	открытые веса
gemini-2.5-flash ¹⁸	неизвестно	по API
GPT5.1-pro ¹⁹	неизвестно	по API
YandexGPT-5.1 ²⁰	неизвестно	по API
gemini-2.5-pro ²¹	неизвестно	по API

Такой выбор позволил провести систематический анализ взаимосвязи между размером модели и качеством обнаружения галлюцинаций, а также учесть архитектурное разнообразие в сопоставимых диапазонах размеров.

Настройка вывода

Вывод для моделей с числом параметров до 33 млрд выполнялся локально на графических процессорах NVIDIA A100 с использованием библиотеки vllm [24] и тензорного параллелизма для распределенного вывода. Более крупные модели с открытым весом и закрытые модели использовались через API

⁹ <https://huggingface.co/t-tech/T-lite-it-2.1>

¹⁰ <https://huggingface.co/openai/gpt-oss-20b>

¹¹ <https://huggingface.co/google/gemma-3-27b-it>

¹² <https://huggingface.co/t-tech/T-pro-it-2.0>

¹³ <https://huggingface.co/meta-llama/Llama-3.3-70B-Instruct>

¹⁴ <https://huggingface.co/deepseek-ai/DeepSeek-V3-0324>

¹⁵ <https://huggingface.co/Qwen/Qwen3-235B-A22B-Instruct-2507>

¹⁶ <https://huggingface.co/deepseek-ai/DeepSeek-V3-0324>

¹⁷ <https://huggingface.co/moonshotai/Kimi-K2-Thinking>

¹⁸ <https://ai.google.dev/gemini-api/docs/models/gemini-2.5-flash>

¹⁹ <https://openai.com/index/gpt-5-1/>

²⁰ <https://ya.ru/ai/gpt>

²¹ <https://ai.google.dev/gemini-api/docs/models/gemini-2.5-pro>

провайдера VseLLM²². Температура семплирования была установлена на 0 для всех моделей для обеспечения детерминированности результатов.

Для автоматизированной оценки модели были ограничены выводом структурированного JSON-массива. Каждый массив содержит объекты с двумя полями:

- 1) **span**: точная текстовая подстрока, содержащая галлюцинацию (null при отсутствии);
- 2) **type**: один из классов: Contradiction, Unconfirmed, Excess, Partial, Missing, Oversight (null при отсутствии галлюцинации).

Такой формат обеспечивает единообразную обработку ответов и вычисление метрик для всех базовых моделей.

Параметры обучения детектора

Для обучения детектора на сбалансированном тренировочном датасете применялась методика SFT на базе YandexGPT-5-Lite-8B-instruct. Обучение выполнялось на одном GPU NVIDIA H100 в течение трех эпох с использованием оптимизатора PagedAdam8bit, learning rate 5×10^{-5} , weight decay 0.05, точности bf16 и параметров LoRA ($\alpha = 32$, $r = 32$). Размер валидационной подвыборки составлял 10%. Модель обучалась предсказывать разметку в формате JSON-массива с полями type и span.

РЕЗУЛЬТАТЫ

Общий обзор

В табл. 2–4 представлены результаты для трех подзадач. Метрики даны с макроусреднением²³, результаты отсортированы по Accurasy.

Табл. 2. Общие метрики бинарной классификации (P = Precision, R = Recall, SR = Success Rate).

Модель	Accurasy	F1	P	R	SR
gemini-2.5-pro	81.3	78.5	79.2	78.0	100
gemini-2.5-flash	75.2	72.2	72.1	72.3	100

²² <https://vsellm.ru/>

²³ Следует отметить, что в некоторых случаях F1-мера может быть ниже как точности, так и полноты. Это не является ошибкой: макроусредненная F1-мера вычисляется не из макроточности и макрополноты, а путем усреднения F1-мер каждого класса.

glm-4.6v	70.1	45.0	50.2	42.0	89.5
gpt5.1-pro	69.5	45.2	45.1	46.0	99.4
T-pro-it-2.0	65.0	42.3	51.1	38.1	81.9
gpt-oss-20b	63.0	38.8	49.6	35.5	81.9
kimi-k2-thinking	56.4	38.3	57.6	31.8	63.9
YandexGPT-5.1	54.3	33.5	41.9	29.9	73.5
llama-3.3-70b-instruct	41.6	39.0	59.4	54.6	100
deepseek-chat-v3-0324	37.0	25.9	39.6	32.7	79.7
Qwen3-235B-A22B-Instruct-2507	35.5	29.5	55.0	50.9	100
Qwen3-8B	30.2	16.0	10.9	30.2	92.7
YandexGPT-5-8B-Lite-instruct	28.6	15.9	30.7	28.5	88.3
T-lite-it-2.1	28.0	15.5	30.5	27.9	88.4
gemma-3-27b-it	26.8	17.9	46.6	26.7	67.0
YandexGPT-5-Lite-SFT	66.7	29.1	39.0	34.0	99.9

Табл. 3. Общие метрики многоклассовой классификации (P = Precision, R = Recall, HL = Hamming Loss,).

Модель	Accuracy	HL ↓	Jaccard	F1	P	R
gemini-2.5-pro	71.6	7.0	34.1	47.0	49.8	46.0
gemini-2.5-flash	63.1	9.5	21.9	31.7	33.2	34.0
glm-4.6v	61.7	10.0	23.7	34.6	40.7	32.0
T-pro-it-2.0	58.1	11.1	19.9	29.3	38.8	25.4
gpt-oss-20b	57.4	11.3	15.8	23.1	41.9	20.8
gpt5.1-pro	55.0	11.4	17.1	25.9	33.0	29.1
YandexGPT-5.1	52.0	12.6	12.6	18.2	30.3	17.4
kimi-k2-thinking	51.9	12.6	18.6	27.6	48.7	21.2
deepseek-chat-v3-0324	17.3	21.3	12.9	22.2	25.1	33.7
llama-3.3-70b-instruct	16.1	22.1	8.3	14.7	19.6	24.7
Qwen3-235B-A22B-Instruct-2507	11.4	22.9	10.5	18.1	24.7	34.8
gemma-3-27b-it	6.5	24.2	7.8	13.9	25.0	22.0
YandexGPT-5-8B-Lite-instruct	5.3	24.0	5.2	9.6	16.8	17.1
Qwen3-8B	4.7	26.1	7.0	12.6	23.8	25.9
T-lite-it-2.1	3.2	27.6	3.8	7.1	12.9	17.9
YandexGPT-5-Lite-SFT	65.8	9.2	9.1	11.5	17.1	13.1

Табл. 4. Общие метрики выделения фрагментов (HL = Hamming Loss).

Модель	Accuracy	HL ↓	Jaccard	ROUGE-L
gemini-2.5-pro	71.7	25.7	73.6	79.4
gpt-oss-20b	67.7	30.7	68.7	70.6
T-pro-it-2.0	66.1	32.6	67.0	71.1
glm-4.6v	65.8	32.5	66.9	72.2
kimi-k2-thinking	65.2	34.2	65.5	69.4
YandexGPT-5.1	63.8	36.1	63.9	65.2
gemini-2.5-flash	61.9	35.5	63.6	70.7
gpt5.1-pro	56.7	41.9	58.0	65.5
deepseek-chat-v3-0324	35.5	60.7	37.8	47.9
gemma-3-27b-it	28.7	68.4	30.1	42.0
llama-3.3-70b-instruct	13.5	83.0	15.1	30.5
YandexGPT-5-8B-Lite-instruct	12.7	86.2	13.5	26.3
T-lite-it-2.1	10.4	86.4	11.9	22.8
Qwen3-8B	8.9	87.9	10.4	23.2
Qwen3-235B-A22B-Instruct-2507	8.3	87.5	10.3	25.3
YandexGPT-5-Lite-SFT	65.7	34.3	65.7	66.3

Бинарная классификация. Наивысшее качество демонстрируют закрытые модели: Gemini-2.5-Pro достигает 81.3%, Gemini-2.5-Flash – 75.2%, GPT5.1-Pro – 69.5%, YandexGPT-5.1 – 54.3%. Среди моделей с открытым весом GLM-4.6v показывает конкурентоспособный результат 70.1%, T-pro-it-2.0 и gpt-oss-20b достигают 65% и 63% соответственно, однако их соблюдение формата составляет лишь 81.9%. Крупные модели Llama-3.3-70B и Qwen3-235B демонстрируют низкую точность (41.6% и 35.5%) при идеальном соблюдении формата. Модели с числом параметров до 8 млрд не превышают 31%. Дообученная YandexGPT-5-Lite-8B достигает точности 66.7% (прирост +38.1 п.п. относительно zero-shot-версии с 28.6%), занимая четвертое место и решая проблему несоблюдения формата (99.9% SR).

Многоклассовая классификация. Наилучший результат принадлежит Gemini-2.5-Pro с точностью 71.6%. Gemini-2.5-Flash следует с заметным отставанием (63.1%), а GLM-4.6v приближается к нему (61.7%). Модели T-pro-it-2.0, gpt-oss-20b, GPT5.1-Pro, YandexGPT-5.1 и Kimi-k2-thinking группируются в диапазоне

51.9–58.1% с различными компромиссами между точностью и полнотой. Значительный разрыв наблюдается ниже 20%: DeepSeek-v3, Llama-3.3-70B и Qwen3-235B показывают 17.3%, 16.1% и 11.4% соответственно, тогда как модели с числом параметров ≤ 8 млрд опускаются ниже 7%. Обученная модель достигает 65.8% – второй результат после Gemini-2.5-Pro, с приростом +60.5 п.п. относительно исходной модели.

Выделение фрагментов. Gemini-2.5-Pro уверенно лидирует с точностью 71.7%. Следующий уровень, включающий gpt-oss-20b, T-pro-it-2.0, GLM-4.6v, Kimi-k2-thinking и YandexGPT-5.1, показывает конкурентоспособные результаты 63.8–67.7% с ROUGE-L 65.2–72.2. Gemini-2.5-Flash опускается на седьмое место (61.9%), что позволяет предположить, что выделение фрагментов в меньшей степени выигрывает от оптимизации Flash. Остальные модели резко деградируют: DeepSeek достигает лишь 35.5%, Gemma-3-27b-it – 28,7%, Llama-3.3-70B и модели с числом параметров ≤ 8 млрд опускаются ниже 15%. Обученная модель показывает 65.7% (четвертое место), уступая только Gemini-2.5-Pro, gpt-oss-20b и T-pro-it-2.0, с приростом +53 п.п. относительно исходной модели.

Анализ по датасетам. Значения Accurasy по датасетам представлены в табл. 5. Все модели демонстрируют более низкие результаты на ruSciBench по сравнению с SberQuAD, что подтверждает большую сложность научного домена. Для бинарной классификации точность Gemini-2.5-Pro снижается с 84.6% до 75.1% (падение 9.5 п.п.), для Gemini-2.5-Flash – с 79.8% до 66.4% (13.4 п.п.), для GLM-4.6v – с 76.8% до 57.4% (19.4 п.п.). Разрыв еще более выражен для многоклассовой классификации: Gemini-2.5-Pro теряет 21.2 п.п., Gemini-2.5-Flash – 28.7 п.п. Выделение фрагментов демонстрирует наибольшее падение: Gemini-2.5-Pro – 22.3 п.п., Gemini-2.5-Flash – 29.9 п.п., gpt-oss-20b – 27.2 п.п., что подчеркивает дополнительные сложности научной терминологии и точного определения границ. Дообученная модель показывает устойчивую производительность: на SberQuAD точность составляет 76.2% (бинарная), 75.3% (многоклассовая) и 75.3% (фрагменты); на ruSciBench – 48.5%, 47.8% и 47.5% соответственно. Падение при переходе к научному домену (27.7–27.8 п.п.) сопоставимо с показателями ведущих моделей. Средняя точность обученной модели 63.7% выводит ее на второе

место после Gemini-2.5-Pro, превосходя исходную модель (16.5%) на 47.2 п.п., что подтверждает эффективность предложенного подхода.

Табл. 5. Метрики точности (бинарная, многоклассовая и по фрагментам) на датасетах SberQuAD и ruSciBench (B = BINARY, ML = MULTI-LABEL, SQ = SberQuAD, RB = ruSciBench, Сред. = среднее по ряду)

Model	B SQ	B RB	ML SQ	ML RB	span SQ	span RB	Сред.
gemini-2.5-pro	84.6	75.1	78.9	57.7	79.4	57.1	72.1
gemini-2.5-flash	79.8	66.4	73.0	44.3	72.2	42.3	63.0
glm-4.6v	76.8	57.4	71.9	42.3	74.5	49.3	62.0
gpt-oss-20b	72.5	44.9	69.0	35.4	77.1	49.9	58.1
T-pro-it-2.0	75.1	45.8	70.4	34.8	76.9	45.5	58.1
gpt5.1-pro	73.4	62.0	64.6	36.8	66.3	38.6	57.0
YandexGPT-5.1	64.3	35.4	62.6	31.9	72.8	46.7	52.3
kimi-k2-thinking	76.0	19.1	71.6	14.5	74.8	47.0	50.5
deepseek-chat-v3-0324	29.9	50.4	17.3	17.4	44.7	18.0	29.6
llama-3.3-70b-instruct	36.2	51.9	18.6	11.3	17.9	5.2	23.5
gemma-3-27b-it	16.8	45.8	5.3	8.7	39.2	8.7	20.8
Qwen3-235B-A22B...	26.9	51.9	10.4	13.3	9.6	5.8	19.7
YandexGPT-5-8B...	20.9	43.2	4.4	7.0	14.7	9.0	16.5
Qwen3-8B	22.3	45.2	3.8	6.4	9.5	7.8	15.8
T-lite-it-2.1	21.8	39.7	3.2	3.2	10.2	10.7	14.8
YandexGPT-5-Lite-SFT	76.2	48.5	75,3	47.8	75.3	47.5	63.7

ОБСУЖДЕНИЕ

Оценка 15 моделей выявила несколько ключевых закономерностей в способностях LLM к обнаружению галлюцинаций.

Размер модели не гарантирует высокого качества. Модели среднего размера (20–33 млрд параметров) могут превосходить более крупные: T-pro-it-2.0 и gpt-oss-20b (средняя точность 58.1%) опережают Llama-3.3-70B (23.5%) и Qwen3-235B (19.7%), несмотря на в 2–11 раз меньшее число параметров. Крупные модели демонстрируют катастрофический провал в многоклассовой классификации (ниже 19%) и выделении фрагментов на ruSciBench (5–6%), что указывает на недостаточность одного лишь масштаба для семантически точного анализа галлюцинаций. Обученная модель YandexGPT-5-Lite-8B (8 млрд) достигает средней

точности 63.7%, подтверждая, что целенаправленная донастройка на сбалансированных синтетических данных может быть эффективнее простого увеличения числа параметров.

Соблюдение формата необходимо, но недостаточно. Модели с идеальным соблюдением формата демонстрируют среднюю точность в диапазоне от 19.7% до 72.1% – разрыв в 52.4 п.п. Llama-3.3-70B и Qwen3-235B безупречно следуют ограничениям на вывод, но достигают лишь 23.5% и 19.7% средней точности с практически нулевой производительностью в задачах детальной разметки. Среди открытых моделей конкурентоспособные варианты имеют лишь 64–82% соблюдения формата, что создает операционные риски. Обученная модель с почти идеальным соблюдением формата (99.9% SR) полностью устраняет эту проблему.

Закрытые модели сохраняют значительное преимущество. Gemini-2.5-Pro уверенно лидирует (средняя точность 72.1%, на 8.4 п.п. выше лучшей открытой модели) с превосходной устойчивостью на ruSciBench: 75.1% (бинарная), 57.7% (многоклассовая) и 57.1% (фрагменты). Разрыв между закрытыми и открытыми моделями достигает 20–35 п.п. на сложных примерах. Обученная модель сокращает этот разрыв до 8.4 п.п. по средней точности.

Сложность задачи создает разрывы в возможностях. Бинарное обнаружение решается наиболее эффективно (наилучший результат 84.6%), но показывает разрывы между датасетами в 27–57 п.п. Многоклассовая классификация демонстрирует самое большое расхождение: ведущие модели достигают 70–79% на SberQuAD, но только 35–58% на ruSciBench. Выделение фрагментов показывает схожие разрывы ниже первых восьми моделей (5–18% точности).

Только четыре модели превышают 60% средней точности, причем лишь две (Gemini-2.5-Pro и обученная YandexGPT-5-Lite-8B) находятся выше 63%. Наиболее остро этот разрыв проявляется для научных областей и многоклассовой классификации, что подчеркивает критическую потребность в надежных, локально развертываемых русскоязычных детекторах и определяет приоритетные направления для будущих разработок, ориентированных на научный и технический контент.

ОГРАНИЧЕНИЯ

Настоящее исследование имеет несколько важных ограничений, которые следует признать. Во-первых, и это наиболее существенно, все ответы в RAG-системе были сгенерированы одной моделью. Это означает, что RusHallu-RAG не является классическим RAG-бенчмарком, оценивающим множество генерационных моделей; скорее, он фокусируется исключительно на способностях к обнаружению галлюцинаций в рамках фиксированной конфигурации. Данное проектное решение ограничивает обобщаемость полученных результатов на различные языковые модели и может не охватывать полный спектр паттернов галлюцинаций, которые могли бы возникать при использовании различных архитектур.

Во-вторых, бенчмарк ограничен двумя конкретными областями: общим знанием и научными документами. Хотя эти области восполняют недостаточно изученный пробел в оценке русскоязычных RAG-систем, они не охватывают другие важные области, такие как новости, корпоративные документы или специализированные технические сферы, которые могут демонстрировать иные характеристики галлюцинаций.

В-третьих, объем тестового набора данных из 1000 пар «запрос – ответ», хотя и достаточен для первоначальных целей исследования, может быть ограничен для обучения устойчивых моделей обнаружения галлюцинаций или для сбора редких типов галлюцинаций в достаточном количестве.

Наконец, автоматическая разметка тренировочного датасета с помощью Gemini-2.5-Pro потенциально вносит смещение, поскольку модель обучается имитировать решения этой конкретной LLM, что может не всегда быть оптимальным с точки зрения согласованности с экспертной оценкой человека. Кроме того, синтетические методы балансировки классов, хотя и эффективны для устранения дисбаланса, могут не полностью воспроизводить распределение и структуру галлюцинаций в реальных данных.

ЗАКЛЮЧЕНИЕ

Впервые представлен RusHallu-RAG – первый бенчмарк для обнаружения галлюцинаций в русскоязычных RAG-системах. Бенчмарк объединяет наборы данных из областей общего знания и науки с контролируемой балансировкой

контекста. В основу разметки была заложена расширенная таксономию галлюцинаций из шести классов, адаптированная для RAG. Разметка примеров была проведена как на уровне ответов, так и на уровне фрагментов текста, что обеспечивает большую детализацию по сравнению с множеством альтернативных решений.

Помимо контрольного множества, был сформирован сбалансированный тренировочный датасет объемом 3000 примеров с использованием синтетических методов генерации для выравнивания распределения классов. На его основе с применением дообучения с учителем был обучен детектор галлюцинаций на базе YandexGPT-5-Lite-8B.

Оценка шестнадцати LLM, включая обученную модель, позволяет сделать следующие выводы.

Научные тексты являются более сложными для обнаружения галлюцинаций, чем тексты из области общего знания.

Качество обнаружения не масштабируется монотонно с ростом числа параметров: модели среднего размера иногда превосходят более крупные. Обученная модель с 8 млрд параметров достигает средней точности 63.7 п.п., обходя значительно более крупные модели с открытыми весами.

Дообучение с учителем на сбалансированных синтетических данных является эффективной стратегией: прирост средней точности по сравнению с исходной моделью без дообучения составил 47.2 п.п., так что обученный детектор занимает второе место после ведущей закрытой модели.

Значительный разрыв сохраняется между закрытыми моделями и моделями с открытым весом. Этот разрыв подчеркивает потребность в точных, локально развертываемых русскоязычных детекторах галлюцинаций. Представленный подход к синтетической балансировке и дообучению демонстрирует многообещающий путь к созданию таких детекторов, особенно для научно-технического контента, где потребность в надежных решениях наиболее остра.

Благодарности

Авторы выражают благодарность Дарье Парамоновой за ценный вклад в процесс разметки данных. Исследование выполнено с использованием суперкомпьютера МГУ-270 Московского государственного университета имени

М. В. Ломоносова. Часть работы по обучению и тестированию моделей подготовлена при поддержке Центра социальных технологий Yandex.Cloud и реализована на платформе Yandex Cloud.

СПИСОК ЛИТЕРАТУРЫ

1. *Ridder F., Schilling M.* The HalluRAG Dataset: Detecting Closed-Domain Hallucinations in RAG Applications Using an LLM's Internal States // Proc. of the 2025 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. 2025.
<https://doi.org/10.48550/arXiv.2412.17056>
2. *Agarwal V., et al.* MedHalu: Hallucinations in Responses to Healthcare Queries by Large Language Models // Findings of the Association for Computational Linguistics: ACL 2025. 2025. <https://doi.org/10.1609/icwsm.v20i1.42623>
3. *Liu T., Zhang Y., Brockett C., Mao Y., Sui Z., Chen W., Dolan B.* A Token-Level Reference-Free Hallucination Detection Benchmark for Free-Form Text Generation // Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Vol. 1: Long Papers). Dublin, Ireland, May 2022. P. 6723–6737.
<https://doi.org/10.18653/v1/2022.acl-long.464>
4. *Li J., Cheng X., Zhao X., Nie J.-Y., Wen J.-R.* HaluEval: A Large-Scale Hallucination Evaluation Benchmark for Large Language Models // Proc. of the 2023 Conference on Empirical Methods in Natural Language Processing. Singapore, December 2023. P. 6449–6464. <https://doi.org/10.18653/v1/2023.emnlp-main.397>
5. *Niu C., Wu Y., Zhu J., Xu S., Shum K., Zhong R., Song J., Zhang T.* RAGTruth: A Hallucination Corpus for Developing Trustworthy Retrieval-Augmented Language Models // Proc. of the 62nd Annual Meeting of the Association for Computational Linguistics (Vol. 1: Long Papers). Bangkok, Thailand, August 2024. P. 10862–10878.
<https://doi.org/10.18653/v1/2024.acl-long.585>
6. *Zhu Z., Yang Y., Sun Z.* HaluEval-Wild: Evaluating Hallucinations of Language Models in the Wild // Advances in Neural Information Processing Systems. 2024. Vol. 37. <https://doi.org/10.48550/arXiv.2403.04307>
7. *Vazquez R., et al.* SemEval-2025 Task 3: Mu-SHROOM, the Multilingual Shared-Task on Hallucinations and Related Observable Overgeneration Mistakes //

Proc. of the 19th International Workshop on Semantic Evaluation (SemEval-2025). Vienna, Austria, July 2025. P. 2472–2497. URL: <https://aclanthology.org/2025.semeval-1.322/> (дата обращения 20.08.2026).

8. Chernogorskii F., et al. DRAGOn: Designing RAG on periodically updated corpus // Proc. of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Vol. 4: Student Research Workshop). 2026. P. 622–638. <https://doi.org/10.18653/v1/2026.eacl-srw.48>

9. Xiong G., Jin Q., Lu Z., Zhang A. Benchmarking Retrieval-Augmented Generation for Medicine // Findings of the Association for Computational Linguistics: ACL 2024. Bangkok, Thailand, August 2024. C. 6233–6251. <https://doi.org/10.18653/v1/2024.findings-acl.372>

10. Tang Y., Yang Y. Multihop-RAG: Benchmarking Retrieval-Augmented Generation for Multi-Hop Queries // arXiv:2401.15391. 2024. <https://doi.org/10.48550/arXiv.2401.15391>

11. Yang X. et al. CRAG – Comprehensive RAG Benchmark // Advances in Neural Information Processing Systems. 2024. Vol. 37. P. 10470–10490. <https://doi.org/10.52202/079017-0335>

12. Pipitone N., Alami G.H. LegalBench-RAG: A Benchmark for Retrieval-Augmented Generation in the Legal Domain // arXiv:2408.10343. 2024. <https://doi.org/10.48550/arXiv.2408.10343>

13. Es S., James J., Espinosa Anke L., Schockaert S. RAGAs: Automated Evaluation of Retrieval Augmented Generation // Proc. of the 18th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations. St. Julians, Malta, March 2024. P. 150–158. <https://doi.org/10.18653/v1/2024.eacl-demo.16>

14. Saad-Falcon J., Khattab O., Potts C., Zaharia M. ARES: An Automated Evaluation Framework for Retrieval-Augmented Generation Systems // Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Vol. 1: Long Papers). Mexico City, Mexico, June 2024. P. 338–354. <https://doi.org/10.18653/v1/2024.naacl-long.20>

15. Friel R., Belyi M., Sanyal A. RAGBench: Explainable Benchmark for Retrieval-Augmented Generation Systems // arXiv:2407.11005. 2024. <https://doi.org/10.48550/arXiv.2407.11005>

16. Wang Z., et al. RAGRouter-Bench: A Dataset and Benchmark for Adaptive RAG Routing // arXiv:2602.00296. 2026. <https://doi.org/10.48550/arXiv.2602.00296>
17. Bao F.S., et al. FaithBench: A Diverse Hallucination Benchmark for Summarization by Modern LLMs // Proc. of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Vol. 2: Short Papers). Albuquerque, New Mexico, April 2025. P. 448–461. <https://doi.org/10.18653/v1/2025.naacl-short.38>
18. Tamber M.S., et al. Benchmarking LLM Faithfulness in RAG with Evolving Leaderboards // Proc. of the 2025 Conference on Empirical Methods in Natural Language Processing: Industry Track. Suzhou (China), November 2025. P. 799–811. <https://doi.org/10.18653/v1/2025.emnlp-industry.54>
19. Efimov P., Chertok A., Boytsov L., Braslavski P. SberQuAD—Russian Reading Comprehension Dataset: Description and Analysis // 11th Conference and Labs of the Evaluation Forum, CLEF 2020. Springer Science and Business Media Deutschland GmbH, 2020. P. 3–15. https://doi.org/10.1007/978-3-030-58219-7_1
20. Vatolin A., Gerasimenko N., Ianina A., Vorontsov K. RuSciBench: Open Benchmark for Russian and English Scientific Document Representations // Doklady Mathematics. 2024. Vol. 110. P. S251–S260. <https://doi.org/10.1134/S1064562424602191>
21. Chen J., et al. AIR-Bench: Automated Heterogeneous Information Retrieval Benchmark // Proc. of the 63rd Annual Meeting of the Association for Computational Linguistics (Vol. 1: Long Papers). Vienna, Austria, July 2025. P. 19991–20022. <https://doi.org/10.18653/v1/2025.acl-long.982>.
22. Chen J., et al. M3-Embedding: Multilinguality, Multi-Functionality, Multi-Granularity Text Embeddings Through Self-Knowledge Distillation // Findings of the Association for Computational Linguistics: ACL 2024. Bangkok, Thailand, August 2024. P. 2318–2335. <https://doi.org/10.18653/v1/2024.findings-acl.137>
23. Kovalev G., Tikhomirov M., Kozhevnikov E., Kornilov M., Loukachevitch N. Building Russian Benchmark for Evaluation of Information Retrieval Models // arXiv:2504.12879. 2025. <https://doi.org/10.48550/arXiv.2504.12879>
24. Kwon W., et al. Efficient Memory Management for Large Language Model Serving with PagedAttention // Proc. of the 29th Symposium on Operating Systems

Principles, SOSP '23. New York, NY, USA: Association for Computing Machinery, 2023. P. 611–626. <https://doi.org/10.1145/3600006.3613165>

25. Tkachenko M., Malyuk M., Holmanyuk A., Liubimov N. Label Studio: Data Labeling Software. 2020–2025. [Электронный ресурс].

URL: <https://github.com/HumanSignal/label-studio> (дата обращения: 20.08.2026).

26. Lin C.-Y. ROUGE: A Package for Automatic Evaluation of Summaries // Text Summarization Branches Out. Barcelona, Spain, July 2004. P. 74–81.

URL: <https://aclanthology.org/W04-1013/> (дата обращения: 20.08.2026).

RusHallu-RAG: A COMPREHENSIVE APPROACH TO HALLUCINATION DETECTION IN RUSSIAN-LANGUAGE RAG SYSTEMS

F. A. Sadkovskii ¹ [0009-0006-4502-9466], R. R. Nasyrova ² [0009-0008-6280-0667],

A. A. Sorokin ^{3, 4} [0000-0003-3877-4223]

^{1–3}*Lomonosov Moscow State University AI Center, Moscow, Russia*

⁴*Yandex, Moscow, Russia*

¹sadkovskiyfa@my.msu.ru, ²NasyrovaRR_01@my.msu.ru, ³sorokinaa@my.msu.ru

Abstract

This paper introduces RusHallu-RAG, the first comprehensive benchmark for detecting hallucinations in Russian-language RAG systems. The dataset of 1,000 query-answer pairs covers general knowledge and scientific domains with controlled context relevance balancing. A fine-grained taxonomy of six hallucination types is proposed and used for annotation at both response and span levels with human experts and LLMs.

For training a hallucination detector, a balanced dataset of 3,000 examples is collected using synthetic generation methods. A detector based on YandexGPT-5-Lite-8B is trained via SFT. We evaluate 15 open-weight and proprietary models alongside the trained detector. Results show that model size does not guarantee performance: medium-sized models (20–33B) sometimes outperform larger ones. The trained detector achieves an average accuracy of 63.7%, ranking second after proprietary Gemini-2.5-Pro, with a gain of 47.2 p.p. over the baseline. The gap between proprietary

and open-weight models is reduced from 20–35 to 8.4 p.p., confirming the effectiveness of the approach. The benchmark, code, and trained model are publicly released.

Keywords: *hallucination detection, retrieval-augmented generation (RAG), large language models (LLMs), benchmarking, russian language, natural language processing (NLP), question answering (QA), model evaluation, supervised fine-tuning (SFT), hallucination taxonomy.*

REFERENCES

1. Ridder F., Schilling M. The HalluRAG Dataset: Detecting Closed-Domain Hallucinations in RAG Applications Using an LLM's Internal States // Proc. of the 2025 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. 2025. <https://doi.org/10.48550/arXiv.2412.17056>
2. Agarwal V., et al. MedHalu: Hallucinations in Responses to Healthcare Queries by Large Language Models // Findings of the Association for Computational Linguistics: ACL 2025. 2025. <https://doi.org/10.1609/icwsm.v20i1.42623>
3. Liu T., Zhang Y., Brockett C., Mao Y., Sui Z., Chen W., Dolan B. A Token-Level Reference-Free Hallucination Detection Benchmark for Free-Form Text Generation // Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Vol. 1: Long Papers). Dublin, Ireland, May 2022. P. 6723–6737. <https://doi.org/10.18653/v1/2022.acl-long.464>
4. Li J., Cheng X., Zhao X., Nie J.-Y., Wen J.-R. HaluEval: A Large-Scale Hallucination Evaluation Benchmark for Large Language Models // Proc. of the 2023 Conference on Empirical Methods in Natural Language Processing. Singapore, December 2023. P. 6449–6464. <https://doi.org/10.18653/v1/2023.emnlp-main.397>
5. Niu C., Wu Y., Zhu J., Xu S., Shum K., Zhong R., Song J., Zhang T. RAGTruth: A Hallucination Corpus for Developing Trustworthy Retrieval-Augmented Language Models // Proc. of the 62nd Annual Meeting of the Association for Computational Linguistics (Vol. 1: Long Papers). Bangkok, Thailand, August 2024. P. 10862–10878. <https://doi.org/10.18653/v1/2024.acl-long.585>
6. Zhu Z., Yang Y., Sun Z. HaluEval-Wild: Evaluating Hallucinations of Language Models in the Wild // Advances in Neural Information Processing Systems. 2024. Vol. 37. <https://doi.org/10.48550/arXiv.2403.04307>

7. Vazquez R., et al. SemEval-2025 Task 3: Mu-SHROOM, the Multilingual Shared-Task on Hallucinations and Related Observable Overgeneration Mistakes // Proc. of the 19th International Workshop on Semantic Evaluation (SemEval-2025). Vienna, Austria, July 2025. P. 2472–2497.
URL: <https://aclanthology.org/2025.semeval-1.322/>.
8. Chernogorskii F., et al. DRAGON: Designing RAG on periodically updated corpus // Proc. of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Vol. 4: Student Research Workshop). 2026. P. 622–638.
<https://doi.org/10.18653/v1/2026.eacl-srw.48>
9. Xiong G., Jin Q., Lu Z., Zhang A. Benchmarking Retrieval-Augmented Generation for Medicine // Findings of the Association for Computational Linguistics: ACL 2024. Bangkok, Thailand, August 2024. C. 6233–6251.
<https://doi.org/10.18653/v1/2024.findings-acl.372>
10. Tang Y., Yang Y. Multihop-RAG: Benchmarking Retrieval-Augmented Generation for Multi-Hop Queries // arXiv:2401.15391. 2024.
<https://doi.org/10.48550/arXiv.2401.15391>
11. Yang X. et al. CRAG – Comprehensive RAG Benchmark // Advances in Neural Information Processing Systems. 2024. Vol. 37. P. 10470–10490.
<https://doi.org/10.52202/079017-0335>
12. Pipitone N., Alami G.H. LegalBench-RAG: A Benchmark for Retrieval-Augmented Generation in the Legal Domain // arXiv:2408.10343. 2024.
<https://doi.org/10.48550/arXiv.2408.10343>
13. Es S., James J., Espinosa Anke L., Schockaert S. RAGAs: Automated Evaluation of Retrieval Augmented Generation // Proc. of the 18th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations. St. Julians, Malta, March 2024. P. 150–158.
<https://doi.org/10.18653/v1/2024.eacl-demo.16>
14. Saad-Falcon J., Khattab O., Potts C., Zaharia M. ARES: An Automated Evaluation Framework for Retrieval-Augmented Generation Systems // Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Vol. 1: Long Papers). Mexico City, Mexico, June 2024. P. 338–354. <https://doi.org/10.18653/v1/2024.naacl-long.20>

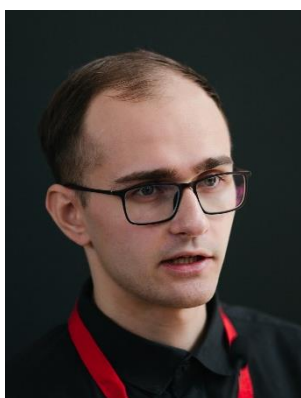
15. *Friel R., Belyi M., Sanyal A.* RAGBench: Explainable Benchmark for Retrieval-Augmented Generation Systems // arXiv:2407.11005. 2024.
<https://doi.org/10.48550/arXiv.2407.11005>
16. *Wang Z., et al.* RAGRouter-Bench: A Dataset and Benchmark for Adaptive RAG Routing // arXiv:2602.00296. 2026. <https://doi.org/10.48550/arXiv.2602.00296>
17. *Bao F.S., et al.* FaithBench: A Diverse Hallucination Benchmark for Summarization by Modern LLMs // Proc. of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Vol. 2: Short Papers). Albuquerque, New Mexico, April 2025. P. 448–461.
<https://doi.org/10.18653/v1/2025.naacl-short.38>
18. *Tamber M.S., et al.* Benchmarking LLM Faithfulness in RAG with Evolving Leaderboards // Proc. of the 2025 Conference on Empirical Methods in Natural Language Processing: Industry Track. Suzhou (China), November 2025. P. 799–811.
<https://doi.org/10.18653/v1/2025.emnlp-industry.54>
19. *Efimov P., Chertok A., Boytsov L., Braslavski P.* SberQuAD–Russian Reading Comprehension Dataset: Description and Analysis // 11th Conference and Labs of the Evaluation Forum, CLEF 2020. Springer Science and Business Media Deutschland GmbH, 2020. P. 3–15. https://doi.org/10.1007/978-3-030-58219-7_1
20. *Vatolin A., Gerasimenko N., Ianina A., Vorontsov K.* RuSciBench: Open Benchmark for Russian and English Scientific Document Representations // Doklady Mathematics. 2024. Vol. 110. P. S251–S260.
<https://doi.org/10.1134/S1064562424602191>
21. *Chen J., et al.* AIR-Bench: Automated Heterogeneous Information Retrieval Benchmark // Proc. of the 63rd Annual Meeting of the Association for Computational Linguistics (Vol. 1: Long Papers). Vienna, Austria, July 2025. P. 19991–20022.
<https://doi.org/10.18653/v1/2025.acl-long.982>.
22. *Chen J., et al.* M3-Embedding: Multilinguality, Multi-Functionality, Multi-Granularity Text Embeddings Through Self-Knowledge Distillation // Findings of the Association for Computational Linguistics: ACL 2024. Bangkok, Thailand, August 2024. P. 2318–2335. <https://doi.org/10.18653/v1/2024.findings-acl.137>
23. *Kovalev G., Tikhomirov M., Kozhevnikov E., Kornilov M., Loukachevitch N.* Building Russian Benchmark for Evaluation of Information Retrieval Models // arXiv:2504.12879. 2025. <https://doi.org/10.48550/arXiv.2504.12879>

24. Kwon W., et al. Efficient Memory Management for Large Language Model Serving with PagedAttention // Proc. of the 29th Symposium on Operating Systems Principles, SOSP '23. New York, NY, USA: Association for Computing Machinery, 2023. P. 611–626. <https://doi.org/10.1145/3600006.3613165>

25. Tkachenko M., Malyuk M., Holmanyuk A., Liubimov N. Label Studio: Data Labeling Software. 2020–2025. URL: <https://github.com/HumanSignal/label-studio> (Accessed 20.08.2026).

26. Lin C.-Y. ROUGE: A Package for Automatic Evaluation of Summaries // Text Summarization Branches Out. Barcelona, Spain, July 2004. P. 74–81. URL: <https://aclanthology.org/W04-1013/> (Accessed: 20.08.2026).

СВЕДЕНИЯ ОБ АВТОРАХ



САДКОВСКИЙ Федор Алексеевич – младший научный сотрудник ИЦИИ МГУ имени М. В. Ломоносова, г. Москва, Россия.

Fedor Alekseevich SADKOVSKII – junior researcher at AI Center, Lomonosov Moscow State University, Moscow, Russia.

email: sadkovskiyfa@my.msu.ru

ORCID: 0009-0006-4502-9466



НАСЫРОВА Регина Руслановна – младший научный сотрудник ИЦИИ МГУ имени М. В. Ломоносова, г. Москва, Россия.

Regina Ruslanovna NASYROVA – junior researcher at AI Center, Lomonosov Moscow State University, Moscow, Russia.

email: nasyrovarr_01@my.msu.ru

ORCID: 0009-0008-6280-0667



СОРОКИН Алексей Андреевич – старший научный сотрудник ИЦИИ МГУ имени М. В. Ломоносова, г. Москва, Россия.

Alexey Andreevich SOROKIN – senior researcher at AI Center, Lomonosov Moscow State University, Moscow, Russia.

Email: sorokinaa@my.msu.ru

ORCID: 0000-0003-3877-4223

Материал поступил в редакцию 27 августа 2026 года