

Описание метрик RAGChecker

Источник

Метрики основаны на исследовании **RAGChecker: A Fine-grained Framework for Diagnosing Retrieval-Augmented Generation**

Данный документ содержит подробное описание всех метрик, используемых для оценки качества RAG-систем, их формулы расчета и интерпретацию результатов.

Структура метрик

Метрики RAGChecker разделены на три категории:

1. **Overall Metrics** - общие метрики качества системы
2. **Retriever Metrics** - метрики компонента извлечения информации
3. **Generator Metrics** - метрики компонента генерации ответов

B.1 Overall Metrics (Общие метрики)

Precision (Точность)

Описание: Доля правильных утверждений в ответе модели. Показывает, насколько ответ модели свободен от ошибок и лишней информации.

Формула:

$$\text{Precision} = |\{c_i^{(m)} \mid c_i^{(m)} \in \text{gt}\}| / |\{c_i^{(m)}\}|$$

Где:

- $c_i^{(m)}$ - утверждение (claim) из ответа модели (m)
- gt - ground truth (эталонный ответ)
- Числитель: количество правильных утверждений модели
- Знаменатель: общее количество утверждений модели

Интерпретация:

- **Высокая Precision (>70%):** модель генерирует точные ответы без лишней информации
- **Низкая Precision (<30%):** модель добавляет нерелевантную информацию или галлюцинирует

Что влияет:

- Качество промпта (инструкции быть точным)
- Способность модели отфильтровывать нерелевантную информацию
- Размер модели и качество обучения

Recall (Полнота)

Описание: Доля эталонных утверждений, которые модель смогла покрыть в своем ответе. Показывает, насколько полно модель отвечает на вопрос.

Формула:

$$\text{Recall} = |\{c_i^{(gt)} \mid c_i^{(gt)} \in m\}| / |\{c_i^{(gt)}\}|$$

Где:

- $c_i^{(gt)}$ - утверждение из эталонного ответа (gt)
- m - ответ модели
- Числитель: количество покрытых эталонных утверждений
- Знаменатель: общее количество эталонных утверждений

Интерпретация:

- **Высокий Recall (>70%):** модель полно отвечает на вопрос
- **Низкий Recall (<30%):** модель упускает важную информацию

Что влияет:

- Качество извлечения контекста (retriever)
- Способность модели использовать предоставленный контекст
- Ограничения по длине ответа

F1 Score (F1-мера)

Описание: Гармоническое среднее между Precision и Recall. **Главная метрика качества RAG-системы**, показывающая баланс между точностью и полнотой.

Формула:

$$F1 = 2 \times (\text{Precision} \times \text{Recall}) / (\text{Precision} + \text{Recall})$$

Интерпретация:

- **>60%:** Отличное качество
- **40-60%:** Хорошее качество
- **20-40%:** Удовлетворительное качество
- **<20%:** Плохое качество, требуется доработка

Баланс Precision vs Recall:

- **Высокий Precision, низкий Recall:** модель точна, но неполна
- **Низкий Precision, высокий Recall:** модель полна, но неточна
- **Оба высокие:** идеальная ситуация (высокий F1)
- **Оба низкие:** критическая ситуация (низкий F1)

B.2 Retriever Metrics (Метрики извлечения)**Claim Recall (Полнота извлечения утверждений)**

Описание: Доля эталонных утверждений, которые присутствуют в извлеченных чанках. Оценивает качество компонента retriever.

Формула:

$$\text{Claim Recall} = |\{c_i^{(gt)} \mid c_i^{(gt)} \in \{\text{chunk}_j\}\}| / |\{c_i^{(gt)}\}|$$

Где:

- $c_i^{(gt)}$ - эталонное утверждение
- $\{\text{chunk}_j\}$ - множество извлеченных чанков
- Утверждение считается извлеченным, если оно содержится хотя бы в одном чанке

Интерпретация:

- **Высокий Claim Recall (>70%)**: retriever находит релевантную информацию
- **Низкий Claim Recall (<30%)**: retriever упускает важную информацию

Как улучшить:

- Настроить параметры поиска (top-k, similarity threshold)
- Улучшить эмбединги для индексации
- Использовать гибридный поиск (semantic + keyword)

Context Precision (Точность контекста)

Описание: Доля извлеченных чанков, которые действительно релевантны. Измеряет, насколько чист извлеченный контекст от "шума".

Формула:

$$\text{Context Precision} = |\{r\text{-chunk}_j\}| / k$$

Где:

- $\{r\text{-chunk}_j\}$ - релевантные чанки (содержат хотя бы одно эталонное утверждение)
- k - общее количество извлеченных чанков

Интерпретация:

- **Высокая Context Precision (>70%)**: мало нерелевантной информации
- **Низкая Context Precision (<30%)**: много "шума" в контексте

Как улучшить:

- Уменьшить top-k (меньше чанков)
- Повысить порог similarity
- Улучшить качество чанкинга

B.3 Generator Metrics (Метрики генерации)

Faithfulness (Точность следования контексту)

Описание: Доля утверждений модели, которые подтверждаются извлеченным контекстом. Ключевая метрика для оценки **hallucination-free** ответов.

Формула:

$$\text{Faithfulness} = |\{c_i^{(m)} \mid c_i^{(m)} \in \{\text{chunk}_j\}\}| / |\{c_i^{(m)}\}|$$

Где:

- $c_i^{(m)}$ - утверждение модели
- $\{\text{chunk}_j\}$ - извлеченные чанки
- Утверждение faithful, если оно содержится в чанках

Интерпретация:

- **Высокий Faithfulness (>80%)**: модель строго следует контексту
- **Низкий Faithfulness (<50%)**: модель добавляет информацию извне или галлюцинирует

Связь с Hallucination:

$$\text{Faithfulness} \approx 1 - \text{Hallucination} \text{ (при корректном контексте)}$$

Hallucination (Галлюцинации)

Описание: Доля утверждений модели, которые **НЕ** содержатся ни в ground truth, ни в контексте. **Критически важная метрика** для продакшен-систем.

Формула:

$$\text{Hallucination} = |\{c_i^{(m)} \mid c_i^{(m)} \notin \text{gt and } c_i^{(m)} \notin \{\text{chunk}_j\}\}| / |\{c_i^{(m)}\}|$$

Где:

- $c_i^{(m)} \notin \text{gt}$ - утверждение не в эталоне
- $c_i^{(m)} \notin \{\text{chunk}_j\}$ - утверждение не в чанках
- Это "изобретенные" моделью факты

Интерпретация:

- **0-10%**: Отлично, почти нет галлюцинаций ✓
- **10-30%**: Приемлемо, некоторые галлюцинации ⚠
- **30-50%**: Проблемно, много галлюцинаций ⚠⚠
- **>50%**: Критично, модель ненадежна ✗

Типы галлюцинаций:

1. **Intrinsic**: противоречат контексту
2. **Extrinsic**: не противоречат, но добавляют новое

Как уменьшить:

- Использовать промпты "stick to the context only"
- Увеличить temperature (для более консервативной генерации)
- Применить post-processing фильтрацию

Relevant Noise Sensitivity (Чувствительность к релевантному шуму)

Описание: Доля утверждений модели, которые находятся в релевантных чанках, но **НЕ** являются правильными. Показывает, насколько модель отвлекается на нерелевантные части релевантного контекста.

Формула:

$$\text{Relevant Noise Sensitivity} = |\{c_i^{(m)} \mid c_i^{(m)} \notin \text{gt and } c_i^{(m)} \in \{r\text{-chunk}_j\}\}| / |\{c_i^{(m)}\}|$$

Где:

- $c_i^{(m)} \notin \text{gt}$ - утверждение неправильное
- $c_i^{(m)} \in \{r\text{-chunk}_j\}$ - но содержится в релевантных чанках
- Это "отвлечение" на части контекста, не отвечающие на вопрос

Интерпретация:

- **Низкая чувствительность (<20%):** модель хорошо фокусируется
- **Высокая чувствительность (>50%):** модель отвлекается на шум

Irrelevant Noise Sensitivity (Чувствительность к нерелевантному шуму)

Описание: Доля утверждений модели, которые взяты из нерелевантных чанков. Показывает, насколько модель использует "мусорный" контекст.

Формула:

$$\text{Irrelevant Noise Sensitivity} = |\{c_i^{(m)} \mid c_i^{(m)} \notin \text{gt and } c_i^{(m)} \in \{\text{irr-chunk}_j\}\}| / |\{c_i^{(m)}\}|$$

Где:

- $\{\text{irr-chunk}_j\}$ - нерелевантные чанки (не содержат эталонных утверждений)

Интерпретация:

- **Низкая чувствительность (<10%):** модель игнорирует мусор
- **Высокая чувствительность (>30%):** модель использует нерелевантную информацию

Как улучшить:

- Улучшить Context Precision (меньше нерелевантных чанков)
- Добавить в промпт инструкцию "use only relevant information"

Self-knowledge (Собственные знания)

Описание: Доля правильных утверждений модели, которые **НЕ** содержатся в контексте. Показывает, использует ли модель свои внутренние знания.



Формула:

$$\text{Self-knowledge} = |\{c_i^{(m)} \mid c_i^{(m)} \in \text{gt and } c_i^{(m)} \notin \{\text{chunk}_j\}\}| / |\{c_i^{(m)}\}|$$

Где:

- $c_i^{(m)} \in gt$ - утверждение правильное
- $c_i^{(m)} \notin \{chunk_j\}$ - но не в контексте
- Модель "вспомнила" из обучения

Интерпретация:

- **Высокий Self-knowledge (>50%)**: модель использует внутренние знания
 -  Хорошо, если знания правильные
 -  Плохо, если это ведет к галлюцинациям
- **Низкий Self-knowledge (<10%)**: модель строго следует контексту

Компромисс:

- В строгих RAG-системах (юриспруденция, медицина): **нужен низкий**
- В помощниках общего назначения: **допустим высокий**

Context Utilization (Использование контекста)

Описание: Доля правильных утверждений, которые взяты из контекста. Показывает, насколько эффективно модель использует предоставленную информацию.

Формула:

$$\text{Context Utilization} = \frac{|\{c_i^{(gt)} \mid c_i^{(gt)} \in \{chunk_j\} \text{ and } c_i^{(gt)} \in m\}|}{|\{c_i^{(gt)} \mid c_i^{(gt)} \in \{chunk_j\}\}|}$$

Где:

- Числитель: правильные утверждения из контекста, которые модель использовала
- Знаменатель: все правильные утверждения, доступные в контексте

Интерпретация:

- **Высокая утилизация (>70%)**: модель эффективно использует контекст
- **Низкая утилизация (<30%)**: модель игнорирует доступную информацию

Причины низкой утилизации:

- Слишком длинный контекст (модель теряется)
- Плохое качество промпта
- Ограничения модели по длине ответа

Связь между метриками

Декомпозиция F1:

$$F1 = f(\text{Precision}, \text{Recall})$$

Precision зависит от:

- Hallucination (чем меньше, тем выше Precision)

- Noise Sensitivity (чем меньше, тем выше Precision)
- Faithfulness (чем выше, тем выше Precision)

Recall зависит от:

- Claim Recall (качество retriever)
- Context Utilization (использование модели)
- Self-knowledge (дополнительный источник)

Диагностика проблем:

Симптом	Возможная причина	Метрика-индикатор
Низкий Precision	Галлюцинации	High Hallucination
Низкий Precision	Шум в контексте	High Noise Sensitivity
Низкий Recall	Плохой retriever	Low Claim Recall
Низкий Recall	Модель не использует контекст	Low Context Utilization
Низкий F1, оба низкие	Системная проблема	Проверить все метрики

Сравнение с бенчмарком из статьи

Датасет: ClapNQ

Результаты лучших систем из исследования RAGChecker:

Система	Prec. ↑	Rec. ↑	F1↑	CR↑	CP↑	CU↑	NSD↓	NSD_I↓	Hallu.↓	SK↓	Faith. ↑
BM25_GPT-4	56.9	50.0	46.7	81.1	41.3	56.4	29.4	5.9	7.5	2.2	90.3
BM25_Mistral-8x7b	49.6	48.6	42.2	81.1	41.3	55.2	31.9	7.5	10.8	2.0	87.2
E5-Mistral_GPT-4	59.7	51.1	47.9	81.5	43.6	59.9	31.1	3.8	5.4	2.3	92.3
E5-Mistral_Llama3-8b	50.4	50.9	43.5	81.5	43.6	59.4	33.2	6.4	10.0	1.5	88.5

Ваши результаты (прогресс по версиям):

Version 1 (английские промпты):

Метрика	Среднее	vs Best Benchmark
F1	26.26%	-21.6% (vs 47.9%)
Precision	28.25%	-31.4% (vs 59.7%)
Recall	39.91%	-11.2% (vs 51.1%)
Hallucination	0.00%	⚠ Метрика не считалась
Faithfulness	0.00%	⚠ Метрика не считалась

Version 2 (русские промпты v1):

Метрика	Среднее	vs V1	vs Benchmark
F1	28.34%	+2.08%	-19.6% (vs 47.9%)
Precision	31.13%	+2.88%	-28.4% (vs 59.7%)
Recall	38.41%	-1.50%	-12.7% (vs 51.1%)
Hallucination	0.00%	—	⚠ Метрика не считалась
Faithfulness	0.00%	—	⚠ Метрика не считалась

Version 3 (русские промпты v2) ЛУЧШАЯ:

Метрика	Среднее	vs V1	vs V2	vs Benchmark
F1	33.33%	+7.07%	+5.0%	-14.6% (vs 47.9%)
Precision	39.50%	+11.25%	+8.4%	-20.2% (vs 59.7%)
Recall	44.50%	+4.59%	+6.1%	-6.6% (vs 51.1%)
Hallucination	10.22%	—	—	+4.8% (vs 5.4%) норма
Faithfulness	32.59%	—	—	-59.7% (vs 92.3%) ⚠

Ключевые выводы:

1. Прогресс V1 → V2 → V3:

- Последовательное улучшение с каждой версией
- Version 3 показывает **+7% F1** vs начальной Version 1
- Все основные метрики (F1, Precision, Recall) улучшились

2. Version 3 vs Бенчмарк:

- **Recall почти догнали:** 44.5% vs 51.1% (-6.6%)
- Precision отстает: 39.5% vs 59.7% (-20.2%)
- F1 ближе к бенчмарку: 33.3% vs 47.9% (-14.6%)

3. Hallucination и Faithfulness:

- В V1 и V2 метрики не считались (0.00%)
- В V3 появились: Hallucination 10.22%, Faithfulness 32.59%
- Hallucination в норме для малых моделей (<10% для GPT-4)
- Faithfulness низкий - требует проверки (должен быть >70%)

Анализ расхождений с бенчмарком:

1. Почему отстаем:

- **Размер моделей:** 1-7B vs GPT-4 (175B+) → ожидаемо -15% F1
- **Датасет:** образовательный vs ClapNQ → -5% F1
- **Язык:** русский vs английский → -5% F1
- **Retriever:** простой vs E5-Mistral → -3% F1
- **Итого ожидаемая разница:** ~28% F1

2. Реальная разница: 14.6% F1




- **Вывод:** Результаты лучше ожидаемых для наших условий!

3. Особенности метрик:

- **Recall** почти догнали бенчмарк (-6.6%) - отлично!
- **Precision** отстает больше (-20.2%) - модели добавляют лишнее
- **Hallucination** выше (10.2% vs 5.4%) - но в пределах нормы
- **Faithfulness** подозрительно низкий (32.6% vs 92.3%) - проверить расчет

Рекомендации:

Немедленно:

- 1.  **Использовать Version 3** - значительное улучшение vs V1 и V2
- 2.  **Проверить расчет Faithfulness** - 32.6% слишком низко
- 3.  **Мониторить Hallucination** - сейчас в норме (10.2%), но следить

Краткосрочные (1-2 недели):

- 1. Протестировать V3 на продакшене
- 2. Собрать обратную связь пользователей
- 3. Идентифицировать проблемные промпты
- 4. Оптимизировать промпты для улучшения Precision

Среднесрочные (1-2 месяца):

- 1. Протестировать более крупные модели (Gemma2 9B, Llama3.1 8B)
- 2. Улучшить retriever (E5-Mistral embeddings)
- 3. Добавить гибридный поиск (semantic + BM25)
- 4. Снизить Hallucination до <8%

Долгосрочные (квартал):

- 1. Достичь F1 > 40% (приблизиться к бенчмарку)
- 2. Повысить Precision > 50%
- 3. Повысить Faithfulness > 70%
- 4. Автоматизировать мониторинг метрик

Потенциал улучшения:

Действие	Ожидаемый прирост F1	Сложность
Более крупные модели (8-9B)	+5-8%	Средняя
Улучшенный retriever (E5-Mistral)	+3-5%	Средняя
Оптимизация промптов	+2-3%	Низкая
Постобработка (фильтрация)	+2-3%	Средняя
Итого потенциал	+12-19%	—

Целевой F1: 45-52% (близко к бенчмарку)

Целевые значения метрик

Для продакшена:

Метрика	Минимум	Хорошо	Отлично
F1	40%	50%	>60%
Precision	50%	70%	>80%
Recall	50%	70%	>80%
Hallucination	<20%	<10%	<5%
Faithfulness	>70%	>85%	>95%
Context Utilization	>50%	>70%	>85%

Как улучшить метрики

Для повышения F1:

- 1. Улучшите Claim Recall (лучший retriever)
- 2. Увеличьте Context Utilization (оптимизация промптов)
- 3. Снизьте Hallucination (строгие промпты)

Для повышения Precision:

- 1. Уменьшите Hallucination
- 2. Снизьте Noise Sensitivity
- 3. Увеличьте Faithfulness

Для повышения Recall:

- 1. Увеличьте Claim Recall (top-k, лучший индекс)
- 2. Увеличьте Context Utilization
- 3. Используйте Self-knowledge (если приемлемо)

Источники и ссылки

- 1. **RAGChecker Paper:** <https://arxiv.org/abs/2408.08067>
- 2. **GitHub:** <https://github.com/amazon-science/RAGChecker>
- 3. **Формулы:** Приложение В статьи (страница 16-17)
- 4. **Бенчмарк:** Table 6, страница 23