Analytics Vidhya

Hands-On: Deep Dive into RAG Evaluation Metrics - Generator Metrics

Instructor

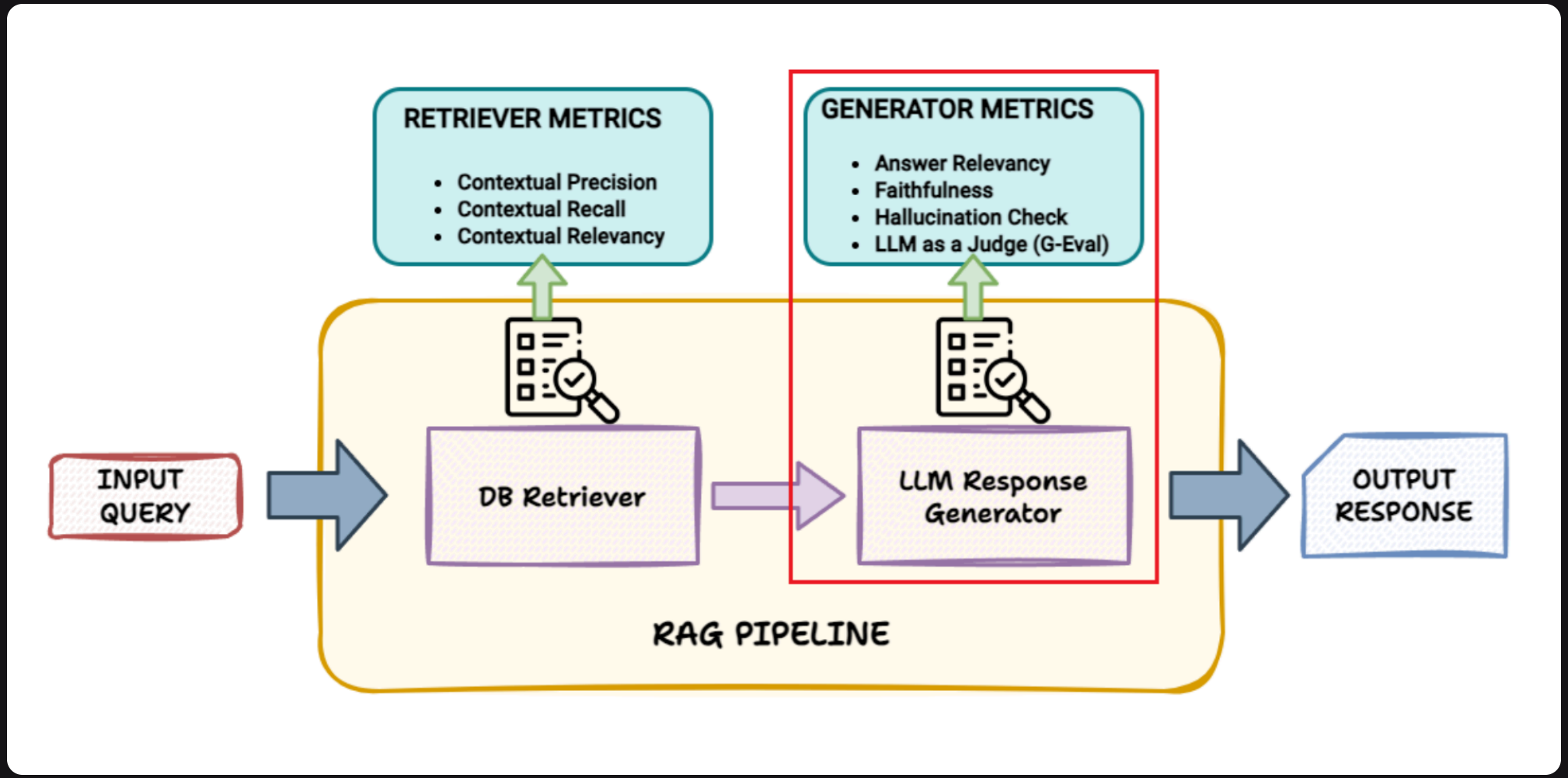
Dipanjan Sarkar

Head of Community & Principal AI Scientist at Analytics Vidhya

Google Developer Expert - ML & Cloud Champion Innovator

Published Author

Generator Evaluation Metrics



Generator Evaluation Metrics

- Answer Relevancy - LLM-Based
- Answer Relevancy - Similarity-Based
- Faithfulness
- Hallucination Check
- Custom LLM as a Judge (G-Eval)

Answer Relevancy - LLM-Based

Measures the relevancy of the information in the **generated response** to the provided **input query** using LLM as a Judge

Higher Answer Relevance shows the LLM Generator is able to generate better quality relevant responses for queries

Answer Relevancy - LLM-Based

$$\text{Answer Relevancy} = \frac{\text{Number of Relevant Statements}}{\text{Total Number of Statements}}$$

Explanation of Key Variables:

- **Number of Relevant Statements:**
  - The count of statements in the actual output (generated response) that are relevant to the input query.
- **Total Number of Statements:**
  - The total count of statements in the actual output (generated response).

How it Works:

- **Relevant Statements**:
  - The evaluation system, using an LLM, identifies statements in the generated response that are directly relevant to answering the input query.
- **Input Query**:
  - The specific question or topic the generated response aims to address.

A higher **Answer Relevancy** score indicates that a greater portion of the information in the generated response is directly relevant to the input query.

Answer Relevancy - LLM-Based

- Input Variables
  - **Input Query**: "What is AI?"
- **Generated Response:**
  - Statements: ["AI refers to machines mimicking human intelligence, such as problem-solving and learning.", "AI includes applications like virtual assistants, robotics, and autonomous vehicles."]

Answer Relevancy - LLM-Based

Computation:

Using the **Answer Relevancy** formula:

$$\text{Answer Relevancy} = \frac{\text{Number of Relevant Statements}}{\text{Total Number of Statements}}$$

1) Identify Relevant Statements:

- **Statement 1:** "AI refers to machines mimicking human intelligence, such as problem-solving and learning."
  - **Verdict:** Relevant
- **Statement 2:** "AI includes applications like virtual assistants, robotics, and autonomous vehicles."
  - **Verdict:** Relevant

2) Count Relevant Statements:

- Number of Relevant Statements: 2
- Total Number of Statements: 2

3) Calculate Answer Relevancy:

- **Answer Relevancy** = 2/2 = 1.0

Answer Relevancy - Similarity-Based

Measures the relevancy of the information in the **generated response** to the provided **input query** using semantic similarity between LLM generated queries from the response and the input query

Higher Answer Relevance shows the LLM Generator is able to generate better quality relevant responses for queries

Answer Relevancy - Similarity-Based

Formula:

$$\text{Answer Relevancy} = \frac{1}{N}\sum_{i=1}^{N}\cos(E_{gi}, E_o) \quad \text{or equivalently,} \quad \text{Answer Relevancy} = \frac{1}{N}\sum_{i=1}^{N}\frac{E_{gi}\cdot E_o}{\|E_{gi}\|\|E_o\|}$$

Explanation of Key Variables:

- E_{gi}: Embedding of the ith generated question (reverse-engineered based on the response).
- E_o: Embedding of the original question (input query).
- N: Number of generated questions (typically defaults to 3).

How It Works:

- **Generate Variants**: Use an LLM to generate variants of the original question based on the answer (reverse engineering).
- **Calculate Cosine Similarity**: Compute the cosine similarity between each generated question's embedding and the original question's embedding.
- **Average Similarity**: Take the mean of these cosine similarity scores as the final **Answer Relevancy** score.

Answer Relevancy - Similarity-Based

- Input Variables
  - **Input Query**: "What is AI?"
- **Generated Response:**
  - "AI refers to machines mimicking human intelligence, such as problem-solving and learning, and includes applications like virtual assistants, robotics, and autonomous vehicles."

Answer Relevancy - Similarity-Based

Computation:

Using the **Answer Relevancy - Similarity-Based** formula:

$$\text{Answer Relevancy} = \frac{1}{N} \sum_{i=1}^{N} \cos(E_{g_i}, E_o)$$

1) **Generate Question Variants** (reverse-engineered from the generated response):

- Question 1: "What is the meaning of AI in terms of human-like intelligence?"
- Question 2: "What applications are included under AI technology?"
- Question 3: "How does AI mimic human intelligence?"

Answer Relevancy - Similarity-Based

Computation:

Using the **Answer Relevancy - Similarity-Based** formula:

$$\text{Answer Relevancy} = \frac{1}{N} \sum_{i=1}^{N} \cos(E_{g_i}, E_o)$$

2) **Calculate Cosine Similarity** between each generated question's embedding and the original question's embedding:

- Cosine similarity for Question 1 and Input Query = 0.92
- Cosine similarity for Question 2 and Input Query = 0.91
- Cosine similarity for Question 3 and Input Query = 0.93

3) **Average Similarity Score**:

- Answer Relevancy = (0.92 + 0.91 + 0.93)/3 = 0.9207

Faithfulness

Measures if the information in the **generated response** factually aligns with the contents of the **retrieved context** document chunks (nodes)

Higher Faithfulness means the generated response is more grounded with regard to the retrieved context reducing contradictions

Faithfulness

$$\text{Faithfulness} = \frac{\text{Number of Truthful Claims}}{\text{Total Number of Claims}}$$

Explanation of Key Variables:

- **Number of Truthful Claims:**
  - The count of claims in the actual output (generated response) that are verified as truthful based on the retrieval context.
- **Total Number of Claims:**
  - The total count of claims made in the actual output (generated response).

How it Works:

- **Identify Claims**:
  - The evaluation system, using an LLM, extracts all claims from the generated response.
- **Verify Truthfulness**:
  - Each claim is checked against the retrieval context to determine if it is consistent with the facts provided.
- **Calculate Faithfulness Score**:
  - The ratio of truthful claims to the total number of claims provides the **Faithfulness** score.

A claim is considered **truthful** if it does not contradict any facts in the retrieval context.

Faithfulness

- Input Variables
  - **Input Query**: "What is AI?"
- Generated Response (Claims):
  - ["AI refers to machines mimicking human intelligence, including problem-solving and learning.", "AI has applications like virtual assistants, robotics, and autonomous vehicles."]
- Retrieved Context:
  - ["Artificial intelligence refers to machines mimicking human intelligence, like problem-solving and learning. AI includes applications like virtual assistants, robotics, and autonomous vehicles. It's evolving rapidly with advancements in machine learning and deep learning.", "NLP is a branch of AI that enables computers to understand, interpret, and generate human language. Techniques include tokenization, stemming, and sentiment analysis. Applications range from chatbots to language translation services.", "Machine learning is a field of artificial intelligence focused on enabling systems to learn patterns from data. Algorithms analyze past data to make predictions or classify information. Popular applications include recommendation systems and image recognition."]

Faithfulness

$$\text{Faithfulness} = \frac{\text{Number of Truthful Claims}}{\text{Total Number of Claims}}$$

1) Identify Claims in the Generated Response:

- Claim 1: "AI refers to machines mimicking human intelligence, including problem-solving and learning."
- Claim 2: "AI has applications like virtual assistants, robotics, and autonomous vehicles."

2) Verify Truthfulness by Checking Against Retrieved Context:

- Claim 1: Verified as truthful (matches context: "Artificial intelligence refers to machines mimicking human intelligence, like problem-solving and learning.")
- Claim 2: Verified as truthful (matches context: "AI includes applications like virtual assistants, robotics, and autonomous vehicles.")

3) Count Truthful Claims:

- Number of Truthful Claims: 2
- Total Number of Claims: 2

4) Calculate Faithfulness Score:

- Faithfulness = 2/2 = 1.0

Hallucination Check

Measures the proportion of contradictory statements by comparing the **generated response** to the **expected context** document chunks (ground truth reference).

Lower the hallucination score, lower the proportion of contradictory statements making the response more grounded and relevant

Hallucination Check

$$\text{Hallucination} = \frac{\text{Number of Contradicted Contexts}}{\text{Total Number of Contexts}}$$

Explanation of Key Variables:

- **Number of Contradicted Contexts:**
  - The count of contexts in the ground truth where the actual output (generated response) presents information that directly contradicts the context.
- **Total Number of Contexts:**
  - The total count of contexts provided as ground truth for comparison.

How It Works:

- **Identify Contradictions:**
  - The evaluation system, using an LLM, examines each context to determine if any part of the generated response contradicts the information provided.
- **Calculate Hallucination Score:**
  - The ratio of contradicted contexts to the total number of contexts provides the **Hallucination** score.

A lower **Hallucination** score (closer to 0) indicates that the generated response aligns well with the provided contexts, with no contradictions.

Hallucination Check

- Input Variables
  - **Input Query**: "What is AI?"
- Generated Response:
  - "AI refers to machines mimicking human intelligence, such as problem-solving and learning, and includes applications like virtual assistants, robotics, and autonomous vehicles."
- Retrieved Context:
  - ["Artificial intelligence refers to machines mimicking human intelligence, like problem-solving and learning. AI includes applications like virtual assistants, robotics, and autonomous vehicles. It's evolving rapidly with advancements in machine learning and deep learning.", "Machine learning is a field of artificial intelligence focused on enabling systems to learn patterns from data. Algorithms analyze past data to make predictions or classify information. Popular applications include recommendation systems and image recognition."]

Hallucination Check

Computation:

Using the Hallucination formula:

$$\text{Hallucination} = \frac{\text{Number of Contradicted Contexts}}{\text{Total Number of Contexts}}$$

1) Check for Contradictions in Each Context:

- **Context 1:** "Artificial intelligence refers to machines mimicking human intelligence, like problem-solving and learning. AI includes applications like virtual assistants, robotics, and autonomous vehicles."
  - **Verdict:** No contradiction (The generated response aligns with this context)
- **Context 2:** "Machine learning is a field of artificial intelligence focused on enabling systems to learn patterns from data. Algorithms analyze past data to make predictions or classify information."
  - **Verdict:** No contradiction (The generated response does not contradict this context and focuses on AI in general, which is consistent)

2) Count Contradicted Contexts:

- Number of Contradicted Contexts: 0
- Total Number of Contexts: 2

3) Calculate Hallucination Score:

- Hallucination = 0/2 = 0.0

Custom LLM as a Judge (G-Eval)

G-Eval is a framework that uses LLMs with chain-of-thoughts (CoT) to evaluate LLM responses and retrieved contexts based on **ANY** custom criteria.

You can decide the judging criteria and give detailed evaluation steps using prompt instructions

Custom LLM as a Judge - GEval

How Is It Calculated?

G-Eval is a two-step algorithm that first generates a series of `evaluation_steps` using chain of thoughts (CoTs) based on the given `criteria`, before using the generated steps to determine the final score using the parameters presented in an `LLMTestCase`.

When you provide `evaluation_steps`, the `GEval` metric skips the first step and uses the provided steps to determine the final score instead.

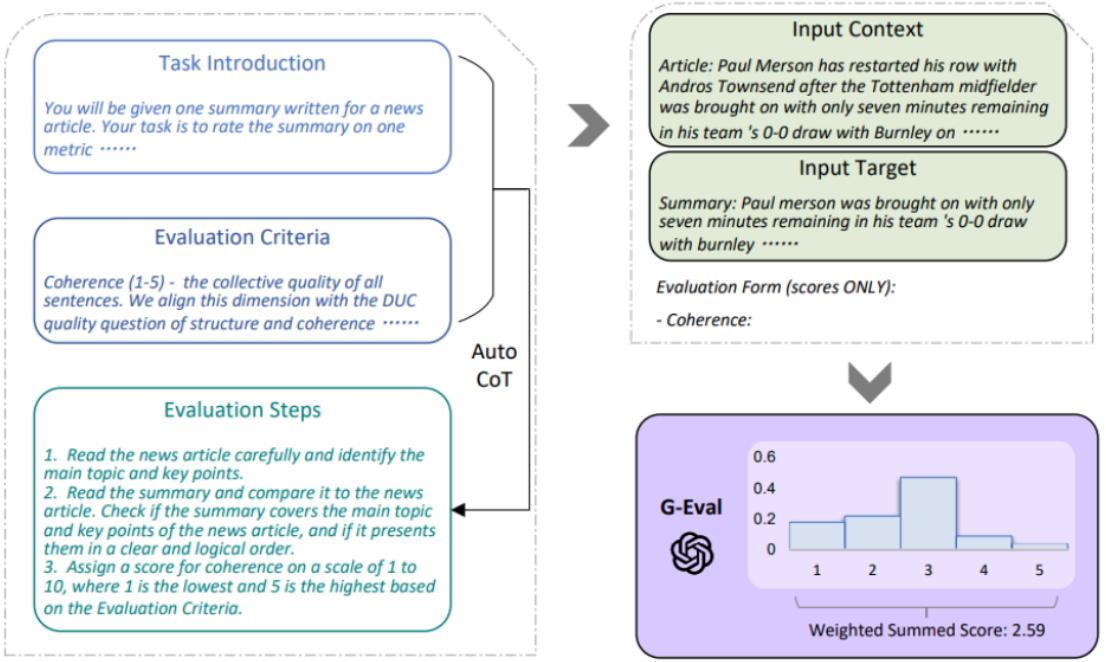


Figure 1: The overall framework of G-EVAL. We first input Task Introduction and Evaluation Criteria to the LLM, and ask it to generate a CoT of detailed Evaluation Steps. Then we use the prompt along with the generated CoT to evaluate the NLG outputs in a form-filling paradigm. Finally, we use the probability-weighted summation of the output scores as the final score.

Analytics Vidhya

Custom LLM as a Judge - GEval

```python
test_case = LLMTestCase(
    input='What is AI?',
    actual_output='AI refers to machines mimicking human intelligence, such as problem-solving and learning, and includes applications like electric sheep and cyborg kittens',
    retrieval_context=["Artificial intelligence refers to machines mimicking human intelligence, like problem-solving and learning. AI includes applications like virtual assistants, robotics, and autonomous vehicles. It's evolving rapidly with advancements in machine learning and deep learning.",.....]
)

metric = GEval(
    threshold=0.5,
    model="gpt-4o",
    name="RAG Fact Checker",
    evaluation_steps=[
        "Create a list of statements from 'actual output'",
        "Validate if they are relevant and answers the given question in 'input', penalize if any statements are irrelevant",
        "Also Validate if they exist in 'expected output', penalize if any statements are missing or factually wrong",
        "Also validate if these statements are grounded in the 'retrieval context' and penalize if they are missing or factually wrong",
        "Finally also penalize if any statements seem to be invented or made up and do not make sense factually given the 'input' and 'retrieval context'"
    ],
    evaluation_params=[LLMTestCaseParams.INPUT,
                       LLMTestCaseParams.ACTUAL_OUTPUT,
                       LLMTestCaseParams.RETRIEVAL_CONTEXT],
    verbose_mode=True
)

result = evaluate([test_case], [metric])
```

```
RAG Fact Checker (GEval) Verbose Logs
**********************************************

Criteria:
None

Evaluation Steps:
[
    "Create a list of statements from 'actual output'",
    "Validate if they are relevant and answers the given question in 'input', penalize if any statements are irrelevant",
    "Also Validate if they exist in 'expected output', penalize if any statements are missing or factually wrong",
    "Also validate if these statements are grounded in the 'retrieval context' and penalize if they are missing or factually wrong",
    "Finally also penalize if any statements seem to be invented or made up and do not make sense factually given the 'input' and 'retrieval context'"
]

Score: 0.5326435311680616
Reason: The actual output partially aligns with the input and retrieval context. It correctly defines AI as machines mimicking human intelligence, which is consistent with the retrieval context. However, it includes irrelevant and potentially made-up examples like 'electric sheep and cyborg kittens' not supported by the context or expected output.
```

Thank You

Analytics Vidhya