RAGAS: Faithfulness Metric

What is it?

Measures if the chatbot's answer is factually correct based on the retrieved context.
Score: 0-1 (higher = more faithful to context)

How it Works

1. Breaks the answer into individual statements
2. Checks if each statement is supported by the context
3. Score = (supported statements) / (total statements)

What You Provide

response: Chatbot's answer
retrieved_contexts: Documents the RAG retrieved

Note: You need backend access to get retrieved_contexts!

Step 1: Capture Data from Your RAG Pipeline

Modify your chatbot code to log the retrieved contexts:

```
# In your RAG chatbot code
docs = vector_db.similarity_search(query)
retrieved_contexts = [doc.page_content for doc in docs]

# Save to database/logs for later testing
save_to_logs({
    "question": query,
    "response": llm_response,
    "retrieved_contexts": retrieved_contexts
})
```

Step 2: Test with RAGAS (Using Real Data)

```
from ragas.metrics import Faithfulness
from ragas.llms import llm_factory
from openai import AsyncOpenAI

client = AsyncOpenAI()
llm = llm_factory("gpt-4o-mini", client=client)
scorer = Faithfulness(llm=llm)

# Get real data from your logs
test_case = get_from_logs()

result = await scorer.ascore(
    response=test_case["response"],
    retrieved_contexts=test_case["retrieved_contexts"]
)
print(result)
```

Note: Do NOT hardcode values. Use actual data captured from your chatbot.

Example 1: HIGH Score (Faithful Answer)

Context: Flynas allows free cancellation within 24 hours of booking.

Answer: You can cancel for free within 24 hours of booking.

Statements: 'Free cancellation' supported, 'Within 24 hours' supported

Result: 2/2 statements supported → Score: 1.0 (HIGH)

Example 2: LOW Score (Hallucinated Answer)

Context: Flynas allows free cancellation within 24 hours of booking.

Answer: You can cancel for free anytime. Full refund guaranteed.

Statements: 'Cancel anytime' NOT supported, 'Full refund' NOT supported

Result: 0/2 statements supported → Score: 0.0 (LOW)

Key Insight

Faithfulness catches hallucinations - when the LLM makes up facts not in the retrieved documents. You must capture real retrieved_contexts from your RAG pipeline to use this metric.