RAG Evaluation: The Good, The Bad, and The Tricky

leboncoin tech

Reka HALMAI

Machine Learning Engineer @ leboncoin

leboncoin

Anis ZAKARI

Machine Learning Engineer @Hymaïa x @leboncoin

hymaïa

Index

01
Introduction

Large Language Models (LLMs)
What are they ?

Key Characteristics:

- Massive deep learning models using transformer architecture
- Billions of parameters
- Require multiple GPUs for high computational power
- ChatGPT popularized LLMs

Training:

- Pre-trained on vast amounts of text data
- Training corpus often unknown (IP concerns)
- Learn language patterns, facts, and reasoning

Functionality:

- Predict next word/token based on previous context
- Generate text sequentially until stop condition

How can we ensure they work?

By evaluating!

Evaluation is about **trust** to ensure the **accuracy** and **relevance** of the generated outputs and avoid **regression.**

Validates the model's performance in high-stakes applications (e.g., healthcare, customer support).

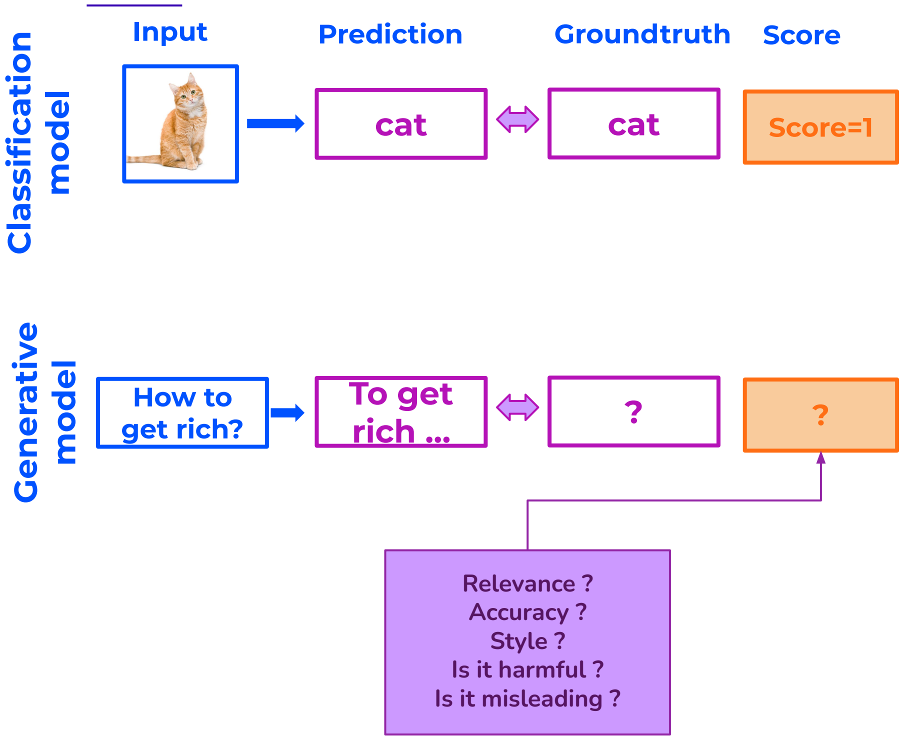
What we need is **good metrics. But what is a good metric?**

02

Why LLM evaluation is tricky?

Evaluate something that is generated

How would you do that ?



Other Difficulties with LLM to consider:

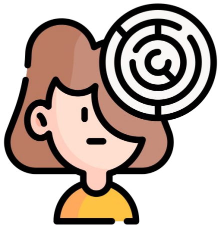
- One input, many possible answers
- Subjectivity
- Sensitivity: slight changes might affect the output
- Non-determinism
- Continuous Evaluation

Evaluation

Types of evaluation

Text similarity metrics
(ex ROUGE, BertScore...)

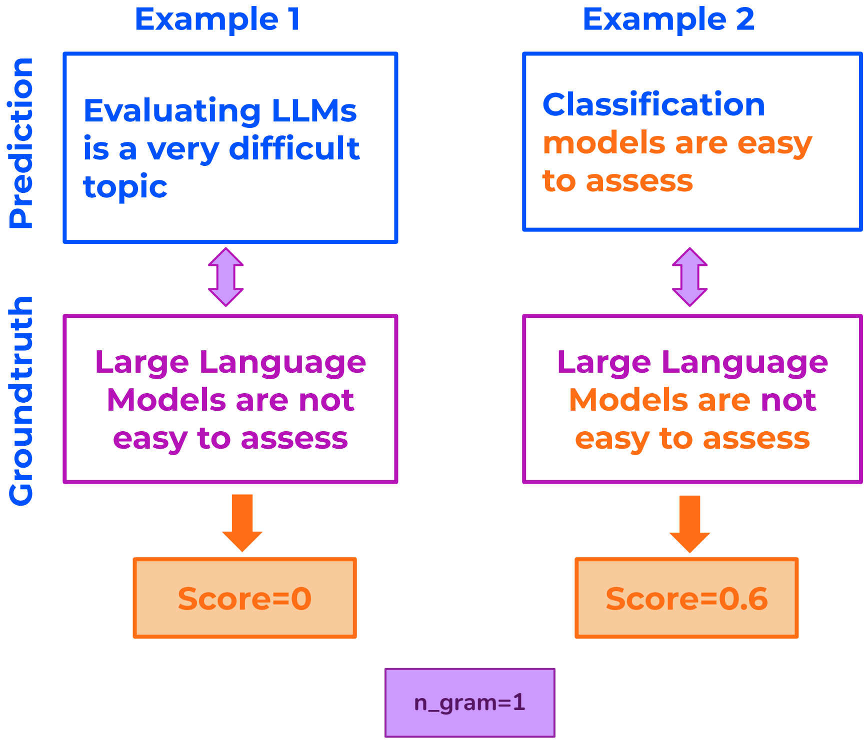
Human Evaluation



Model evaluation/ LLM as a judge

Evaluation with text similarity scores

ROUGE: Naive Text Similarity Metric with Inherent Limitations



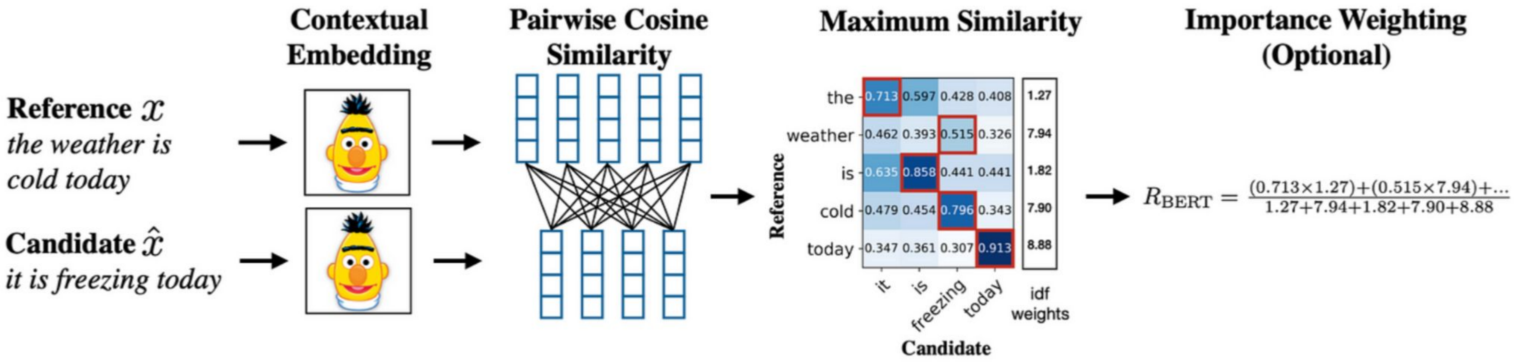
Text similarity scores are:

- **Limited** to **surface-level text similarity**
  - ignores context and semantics
  - low correlation with human judgments
- Unable to measure **factual accuracy** or **user satisfaction**
- **Biased** toward rigid matching of reference texts

* Source: Blagec, Dorffner, Moradi: A global analysis of metrics used for me
* ROUGE: Recall-Oriented Understudy for Gisting Evaluation

Evaluation with semantic similarity scores

BERTScore - Capturing Meaning Beyond Surface Matching



Advantages:

- Captures semantics beyond surface matching
- Handles paraphrases and linguistic variations
- Correlates better with human judgments

Limitations:

- Biased toward reference text
- Semantic similarity =/= relevance
- Computationally expensive

Human Evaluation Challenges

The Human Factor

User feedbacks

Rate answers on a scale

Advantages

- Captures **nuanced aspects**
- Identifies **safety risks** and biases
- **Qualitative insights** for improvement

Disadvantages

- **Cost and Time - difficult to scale**
- **Need for Expertise:** Requires domain experts/ specific knowledge
- **Subjectivity and Bias:**
    - Evaluators' differing opinions and biases can skew results.
    - Inconsistent judgments reduce reliability and reproducibility

LLM as a judge
Challenges of AI-Based Evaluation

Advantages

- **Cost-effective** alternative to human evaluation
- **Scalable** for large-scale assessments
- **Consistent scoring (more or less)** across multiple samples
- Empirically **good results**

Disadvantages

- Potential **biases**
  - ➢ Inherited from training data
  - ➢ Prefer a single score
  - ➢ Prefer own model prediction
  - ➢ Order counts
- Requires **careful prompt engineering** and fine-tuning
- **Lack of determinism** in iterations

Best used in conjunction with human oversight

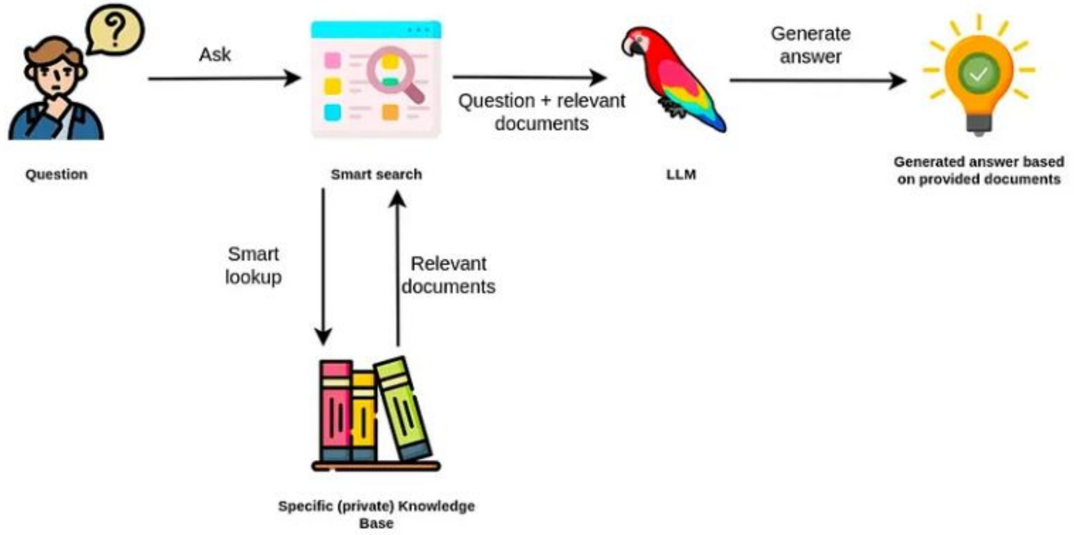
03

Why RAG evaluation is even trickier?

What is RAG?

Understanding Retrieval Augmented Generation (RAG)

RAG combines **retrieval** (locating relevant data from a knowledge base) with text **generation** (leveraging the retrieved information as context for generation).



Key benefits:

- Adapts the LLM to **rare or unseen data**
- **Enhances factual accuracy and reduces hallucinations.**

RAG Evaluation

The metrics should answer the questions...

Did we find the right information?

- Information retrieval / Ranking metrics (Precision, recall, F1 score, MRR, nDCG)
- Reference context via LLM as a judge

Given the retrieved information, did we generate the correct answer?

- Rule based metrics
- Groundedness via LLM as a judge

Is the prediction the correct answer?

- Standard scores (ROUGE, METEOR)
- Semantic based scores (BertScore)
- Human evaluation
- Answer relevance via LLM as a judge

RAG-Specific Challenges 1.
Duality

Combination of **retrieval** and **generation**, each requiring different evaluation metrics.

➡ **Challenges related to the retriever**

- Irrelevant information “creates noise”
- Lack of relevant information makes it impossible to reply well

➡ **Challenges related to the generation**

- Preserving relevant information
- Mitigating hallucinations/incorrect generations
- Handling query diversity effectively

Interdependence between them makes it hard to isolate their impact on performance

➡ **Interdependence Traps**

RAG Interdependence Traps 1.

Lost in the middle

When the prompt is very long, important information in the middle might be lost.

Even if you retrieve the correct document you are not guaranteed to generate the good answer

This one is really relevant

PROMPT

You are a useful assistant, here is your task...

Here are the relevant documents:

- A
- B
- C
- D
- E

<More instructions>

<user question>

LLM Answer

According to my knowledge there's no relevant document ...

RAG Interdependence Traps 2.
Context Truncation

We've correctly retrieved documents A, B, C but needed D for the whole context

PROMPT

You are a useful assistant, here is your task...

Here are the relevant documents:

- **A**
- **B**
- **C**

<More instructions>

<user question>

LLM Answer

According to my knowledge, the process starts with ...

The generation component fails to add the proper project ending.

Even if you retrieved the correct documents, you might miss some information.

RAG Interdependence Traps 3.

Hallucination Due to Weak Retrieval

The generation component hallucinates because of the irrelevant context retrieved.

Even if you retrieve the correct document, you might need to control for not retrieving the incorrect ones.

This doc is relevant

These docs are not relevant

PROMPT

You are a useful assistant, here is your task...

Here are the relevant documents:

- **A**
- **B**
- **C**

<More instructions>

<user question>

LLM Answer

According to my knowledge, ...

RAG-Specific Challenges 2.

Complexity related to the knowledge base

Knowledge base quality:

- Incompleteness, biases, and inaccuracies
- Outdated or stale information
- Duplicates and redundant information
- Inconsistencies and contradictions within the knowledge base

Dynamic External Knowledge:

- Frequently updated knowledge bases
- Difficulty in consistent evaluation

Multilingual bias in LLM/Embedding models

04
Our use cases

Use case 1: Markus

leboncoin's Customer care Assistant

What:

- Reply to users tickets automatically

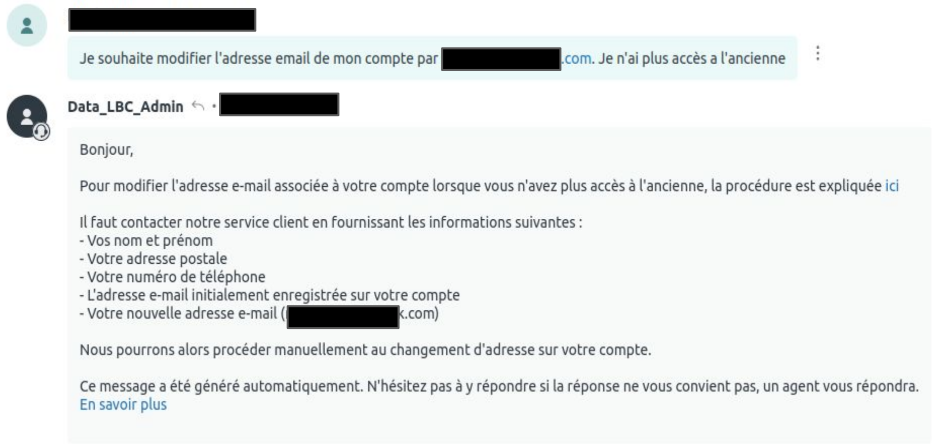
Why? :

- ease the burden on customer service.
- ~200k tickets /month

LangChain

FAISS

Amazon Bedrock



[redacted]

Je souhaite modifier l'adresse email de mon compte par [redacted].com. Je n'ai plus accès a l'ancienne

Data_LBC_Admin · [redacted]

Bonjour,

Pour modifier l'adresse e-mail associée à votre compte lorsque vous n'avez plus accès à l'ancienne, la procédure est expliquée ici

Il faut contacter notre service client en fournissant les informations suivantes :
- Vos nom et prénom
- Votre adresse postale
- Votre numéro de téléphone
- L'adresse e-mail initialement enregistrée sur votre compte
- Votre nouvelle adresse e-mail ([redacted]k.com)

Nous pourrons alors procéder manuellement au changement d'adresse sur votre compte.

Ce message a été généré automatiquement. N'hésitez pas à y répondre si la réponse ne vous convient pas, un agent vous répondra.
En savoir plus

Evaluating Markus

Different evaluation steps

Pre-production evaluation:

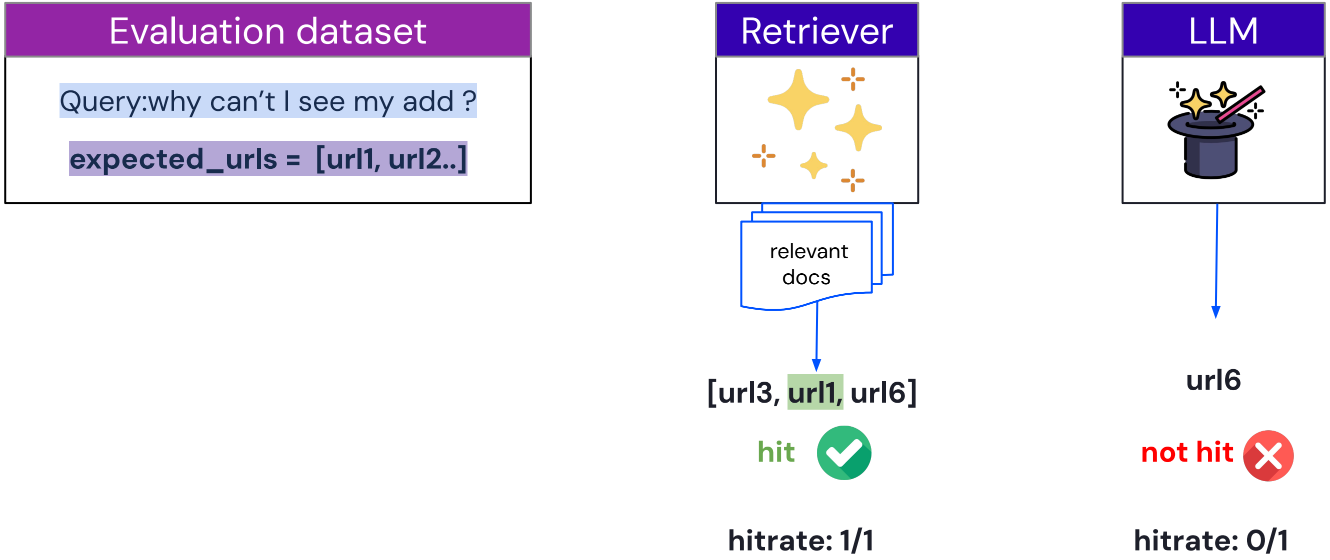
- Retriever evaluation
- LLM's ability to choose the right documents
- LLM as a judge to assess generated answers on specific scenarios
- Customer care support to assess if answers are acceptable

Post-production evaluation:

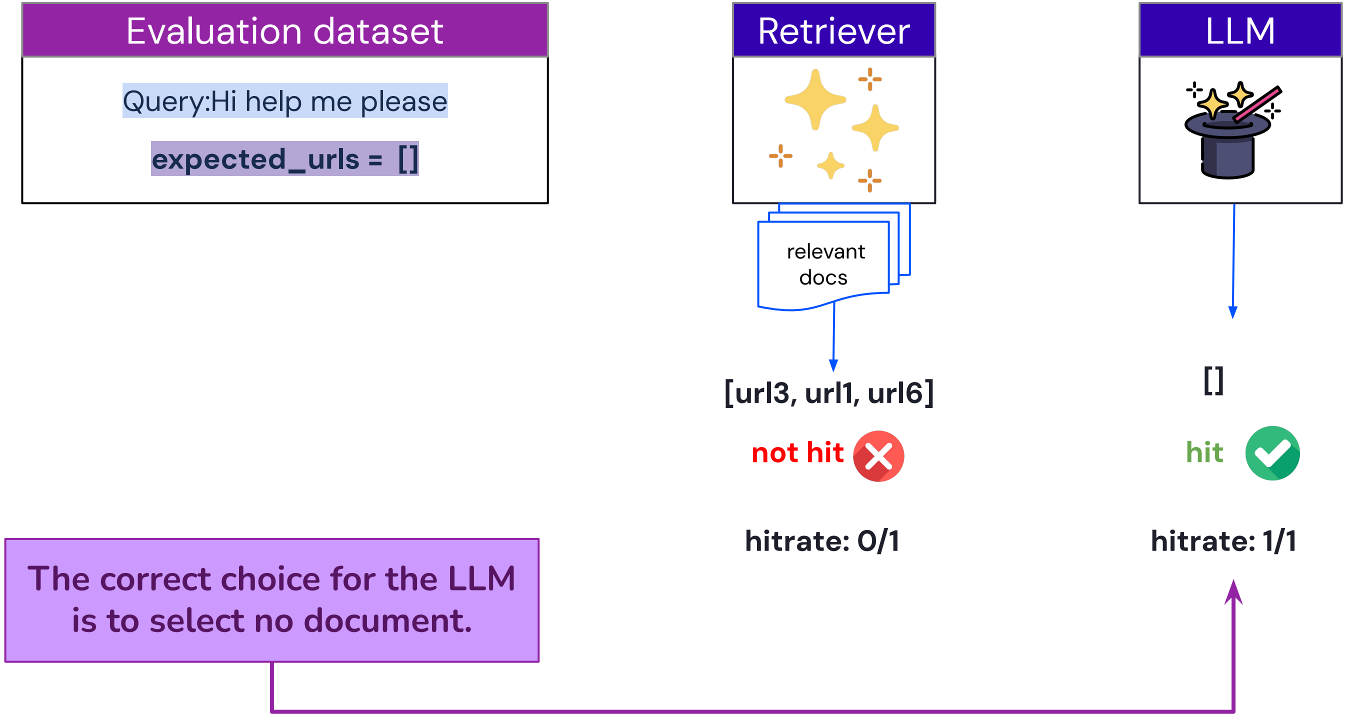
- Guardrails
- Feedback from customer care

Small dataset on specific scenarios to be covered

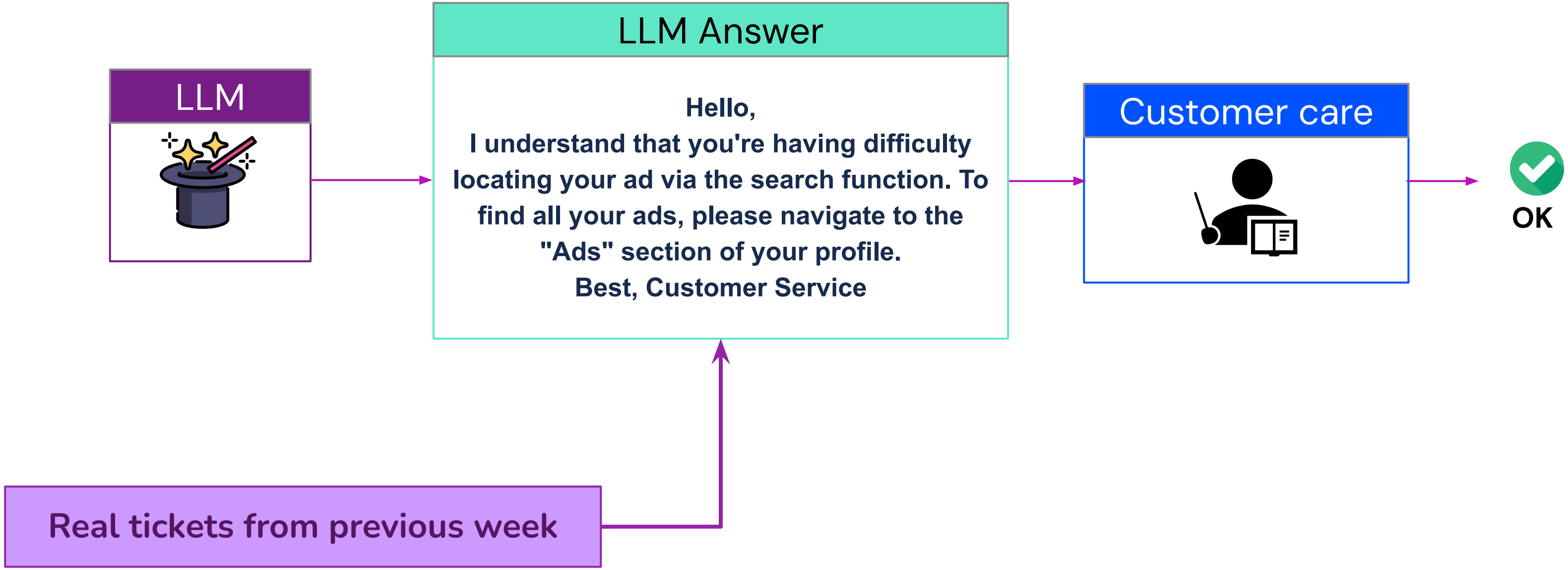
Positive examples



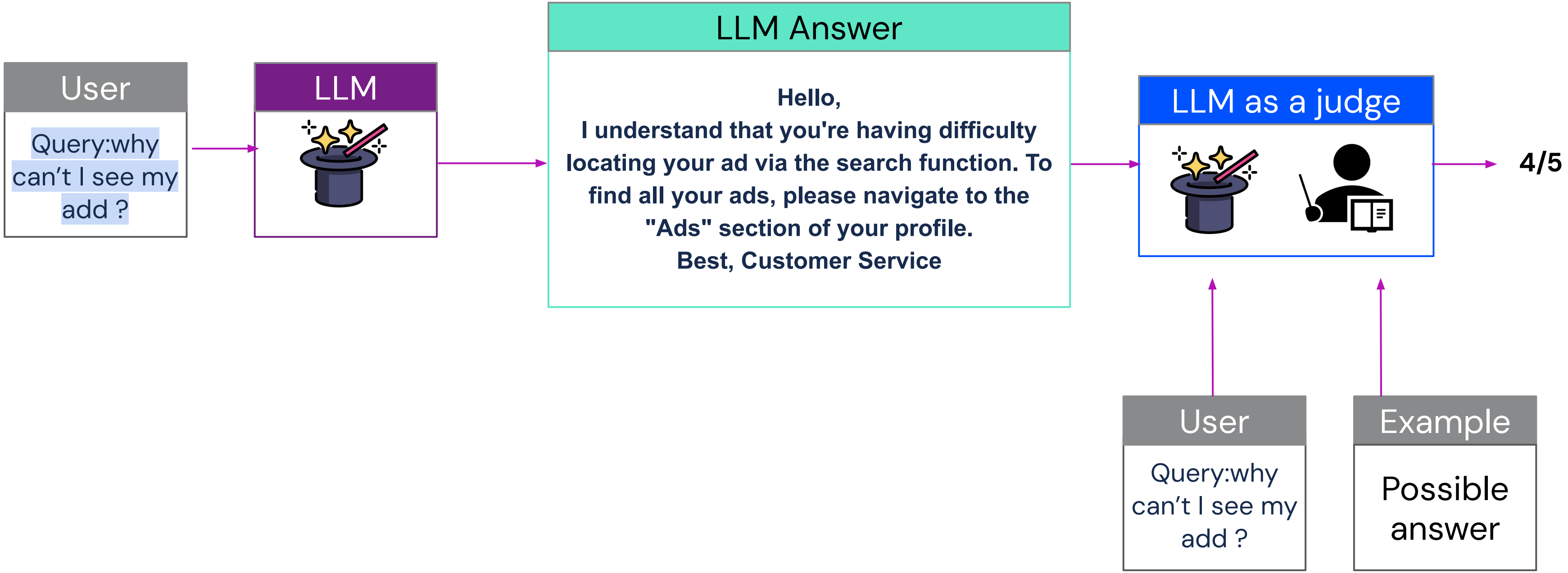
Small dataset on specific scenarios to be covered
Negative examples



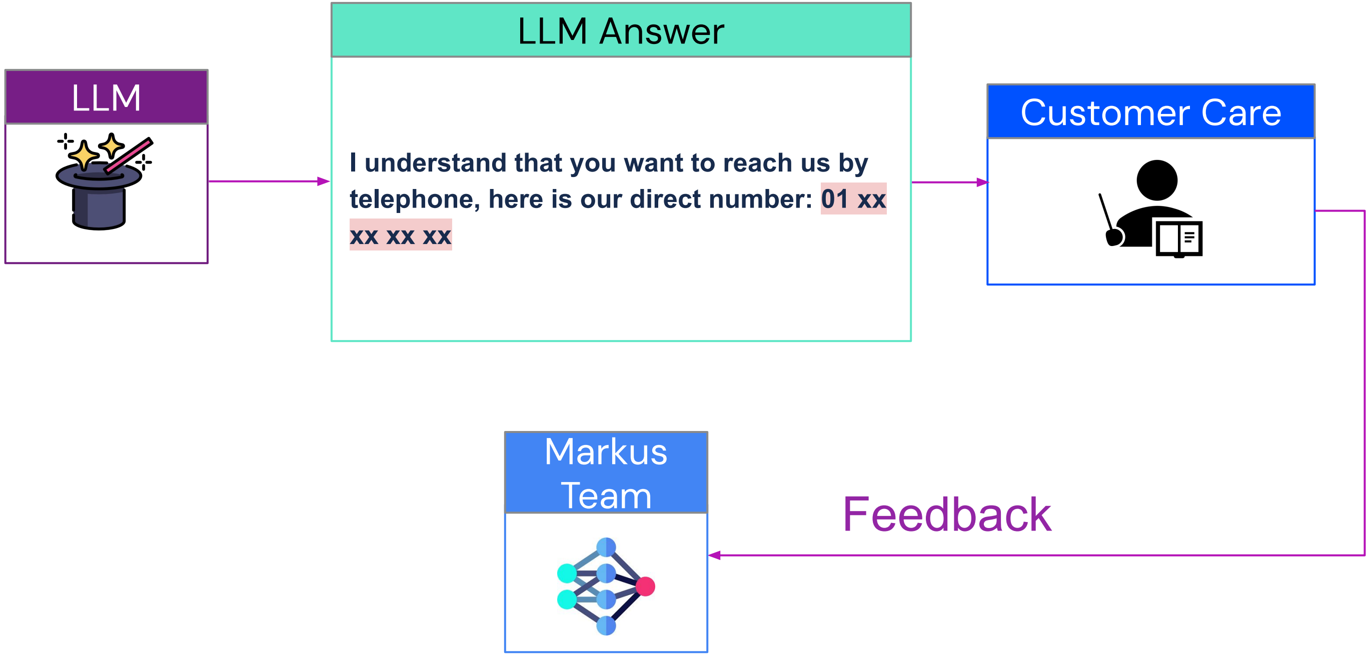
Customer care support
Validate behaviour and accuracy



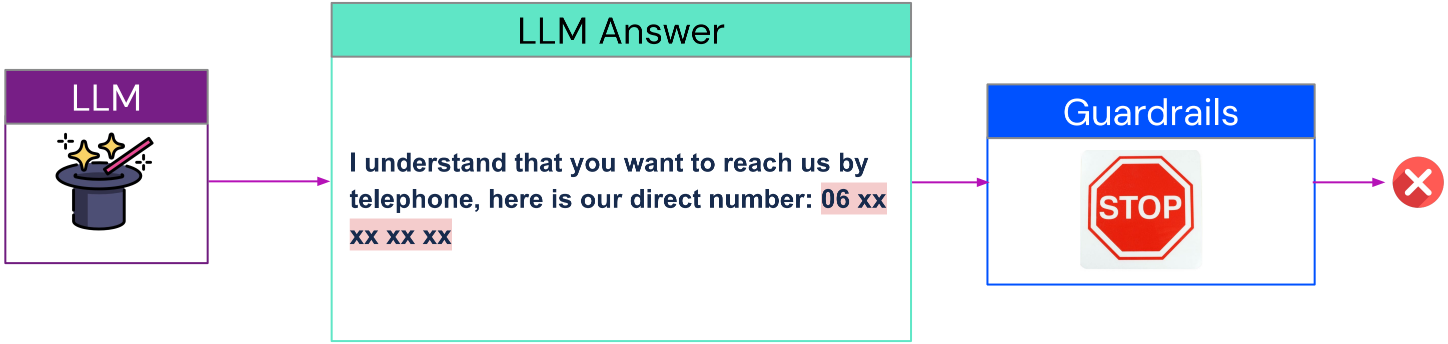
LLM as a judge
Validate behaviour and accuracy



Feedback from Customer Service
Human in the loop



Post Production
Guardrails



Use case 2: Ada - AI Assistant

leboncoin’s internal chatbot

leboncoin’s internal chatbot developed as a result of privacy concerns.

Key Services:

- General chatbot (UI and Slack)
- Slack thread and channel summarization
- Document upload and processing

RAG Integration for Internal Knowledge:

- Company policies (expenses, procurement, etc...)
- Projects, onboarding, and team information (Confluence)

Evaluating Ada RAG?

Process

Off-line evaluation	Release in dev, ask for beta user feedback	Deploy

Evaluate iterations of the RAG system with **metrics** and store them in **Langfuse**

We evaluate each major iteration on an evaluation set:

- FAQs for policies
- Confluence: Encourage users to contribute **Q&As**.
- **User-generated data**: collect questions during user testing, feedback or support tickets.

1. **Duration:** typically a week

2. **Purpose:**
   - Allow users to explore new features/behaviors
   - Gather feedback
   - Collect evaluation data

Presently, **no online evaluation** because we don't monitor what users ask about.

We plan to add **likes** with option to reuse the example to improve the performance.

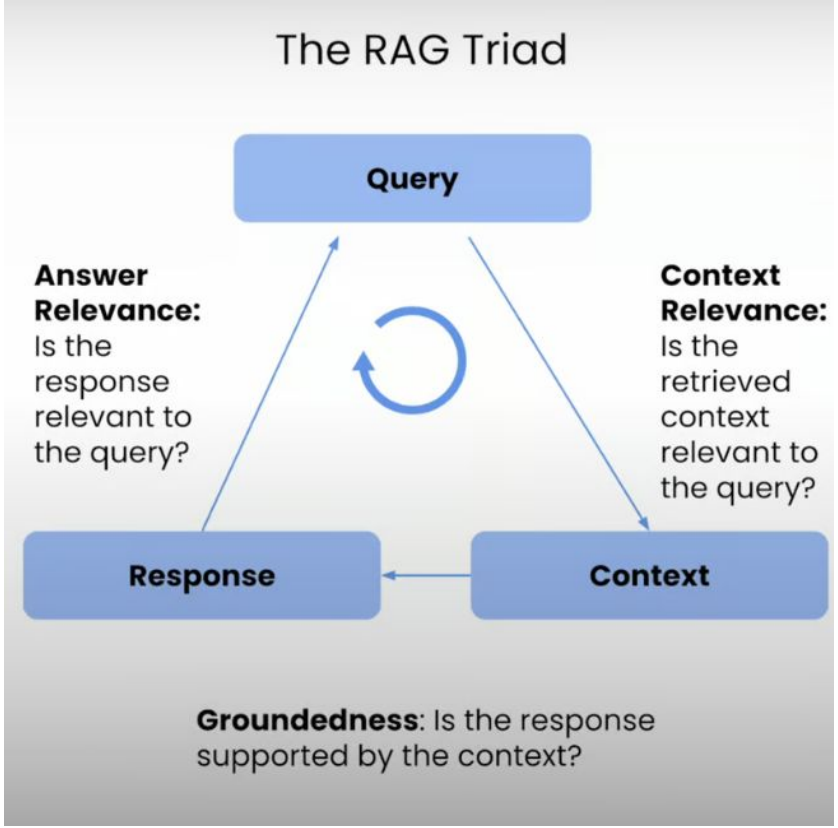
Evaluating Ada

Scores cont'- LLM as a judge

LLM as a judge scores

- Context relevancy
- Answer relevancy
- Groundedness/ Faithfulness

These scores are normalised between 0 and 1.



Evaluating Ada continued

LLM as Judge: How the Metrics Work

These metrics are based on **prompting** and an **LLM call.**

These metrics are **sensitive to**:

- small changes in the prompt
- the order of the inputted information
- the length of inputted information
- the model used

Some python packages (Trulens, ragas) have them implemented.

```
context_relevance_system_prompt = """You are a RELEVANCE grader; providing the
relevance of the given CONTEXT to the given QUESTION.
       Respond only as a number from 0 to 10 where 0 is the least relevant and 10 is
the most relevant.
       A few additional scoring guidelines:
       - Long CONTEXTS should score equally well as short CONTEXTS.
       - RELEVANCE score should increase as the CONTEXTS provides more RELEVANT
context to the QUESTION.
       - RELEVANCE score should increase as the CONTEXTS provides RELEVANT context to
more parts of the QUESTION.
       - CONTEXT that is RELEVANT to some of the QUESTION should score of 2, 3 or 4.
Higher score indicates more RELEVANCE.
       - CONTEXT that is RELEVANT to most of the QUESTION should get a score of 5, 6,
7 or 8. Higher score indicates more RELEVANCE.
       - CONTEXT that is RELEVANT to the entire QUESTION should get a score of 9 or
10. Higher score indicates more RELEVANCE.
       - CONTEXT must be relevant and helpful for answering the entire QUESTION to get
a score of 10.
       - Never elaborate."""

context_relevance_user_prompt = """QUESTION: {question}
      CONTEXT: {context}
      Please answer with the template below for all statement sentences:
      Supporting Evidence: <Identify the sentences from the CONTEXT where the
information is most relevant to the QUESTION.>
      Score: <Output a number between 0-10 where 0 is no information overlap and 10 is
all information is overlapping>
   """
```

RAG evaluation UI

Langfuse

Open-source observability platform for LLM applications. Helps track, analyze, and optimize AI interactions

Features:

- Tracing and logging datasets and metrics via experiments
- Prompt management
- Performance monitoring
- Cost tracking

Benefits:

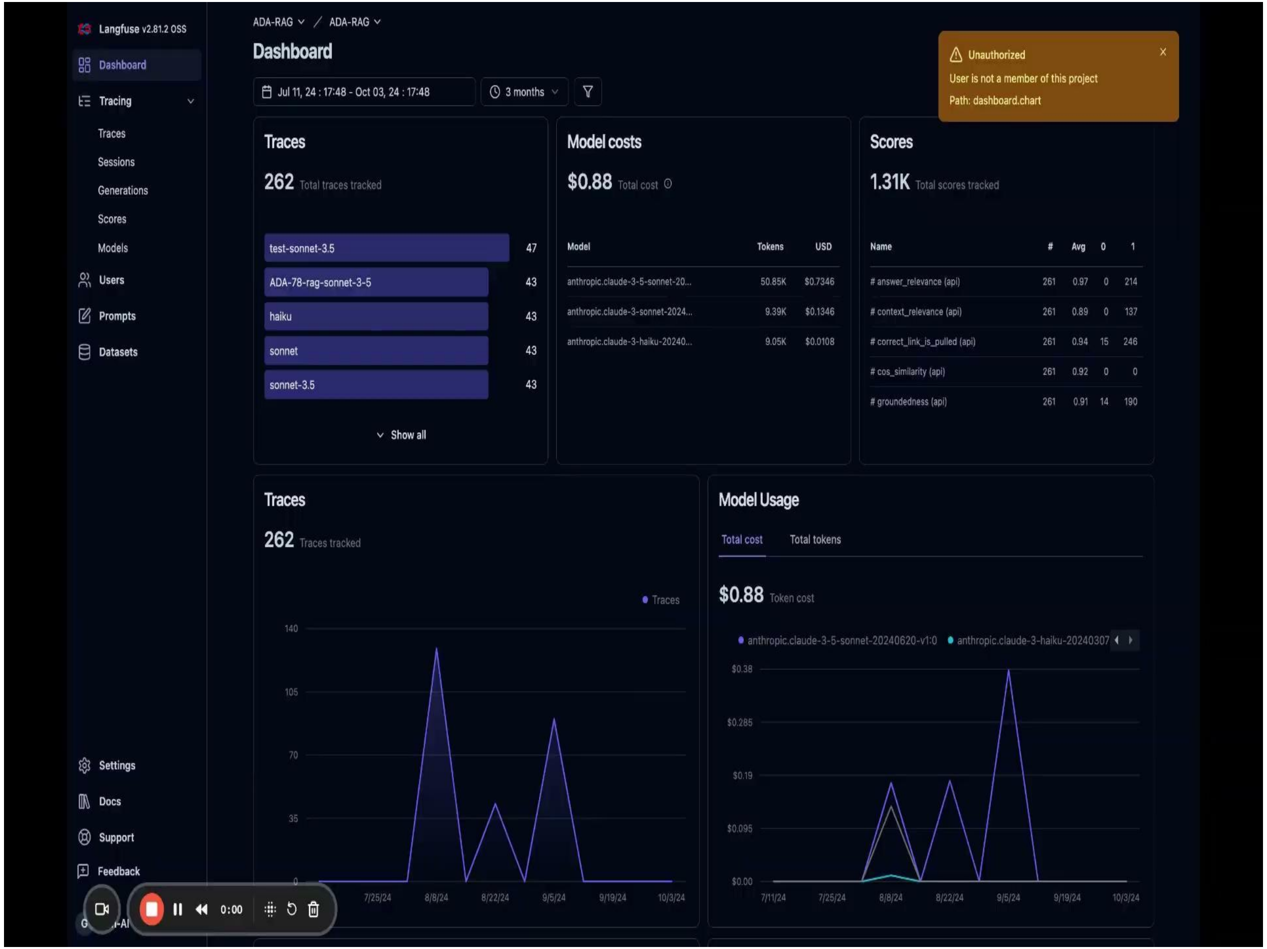
- Easy integration to the code.
- Centralized location for tracking performance metrics.
- Enhances transparency and reproducibility in AI evaluation workflows.

RAG evaluation UI
Langfuse

Disadvantages of Langfuse:

- Limited Customization
- No native filtering capacity
- Limited analytical features
- Not easy to analyse the results

Alternatives to Langfuse:


Langfuse v2.81.2 OSS
Dashboard
Tracing
Traces
Sessions
Generations
Scores
Models
Users
Prompts
Datasets
Settings
Docs
Support
Feedback
ADA-RAG / ADA-RAG
Dashboard
Jul 11, 24 : 17:48 - Oct 03, 24 : 17:48
3 months
Unauthorized
User is not a member of this project
Path: dashboard.chart
Traces
262 Total traces tracked
test-sonnet-3.5 47
ADA-78-rag-sonnet-3-5 43
haiku 43
sonnet 43
sonnet-3.5 43
Show all
Model costs
$0.88 Total cost
Model Tokens USD
anthropic.claude-3-5-sonnet-20... 50.85K $0.7346
anthropic.claude-3-sonnet-2024... 9.39K $0.1346
anthropic.claude-3-haiku-20240... 9.05K $0.0108
Scores
1.31K Total scores tracked
Name # Avg 0 1
answer_relevance (api) 261 0.97 0 214
context_relevance (api) 261 0.89 0 137
correct_link_is_pulled (api) 261 0.94 15 246
cos_similarity (api) 261 0.92 0 0
groundedness (api) 261 0.91 14 190
Traces
262 Traces tracked
Traces
140
105
70
35
0
7/25/24
8/8/24
8/22/24
9/5/24
9/19/24
10/3/24
Model Usage
Total cost
Total tokens
$0.88 Token cost
anthropic.claude-3-5-sonnet-20240620-v1:0
anthropic.claude-3-haiku-20240307
$0.38
$0.285
$0.19
$0.095
$0.00
7/11/24
7/25/24
8/8/24
8/22/24
9/5/24
9/19/24
10/3/24
0:00

04
Wrap up

Evaluation ensures RAG:

• Improves response accuracy, style and behaviour.

• Handles domain-specific queries effectively

• Reduces hallucinations and factual errors.

and identifies where we can improve the system

But, evaluation is difficult because of:

•Growing knowledge bases complicate assessment

• Time-consuming manual review is often necessary

• Challenge to evaluate constantly

And it's even more tricky because of:

• Defining good metrics and aggregating them is challenging

Non-reproducibility and biases

• Difficulties to pinpoint where an error comes from due to interdependence

Key takeaways
What we have learned

- Text-similarity metrics are often inadequate for comprehensive evaluation.
- distinguishing between information retrieved and hallucinated by the model can be unclear.

- Usually, best is to **mix a variety of metrics**:
  - Rule based
  - Semantic based
  - LLM as a judge

- Effective strategies need to consider real-world contexts and use cases but getting good evaluation data can be challenging.
- Guardrails are essential

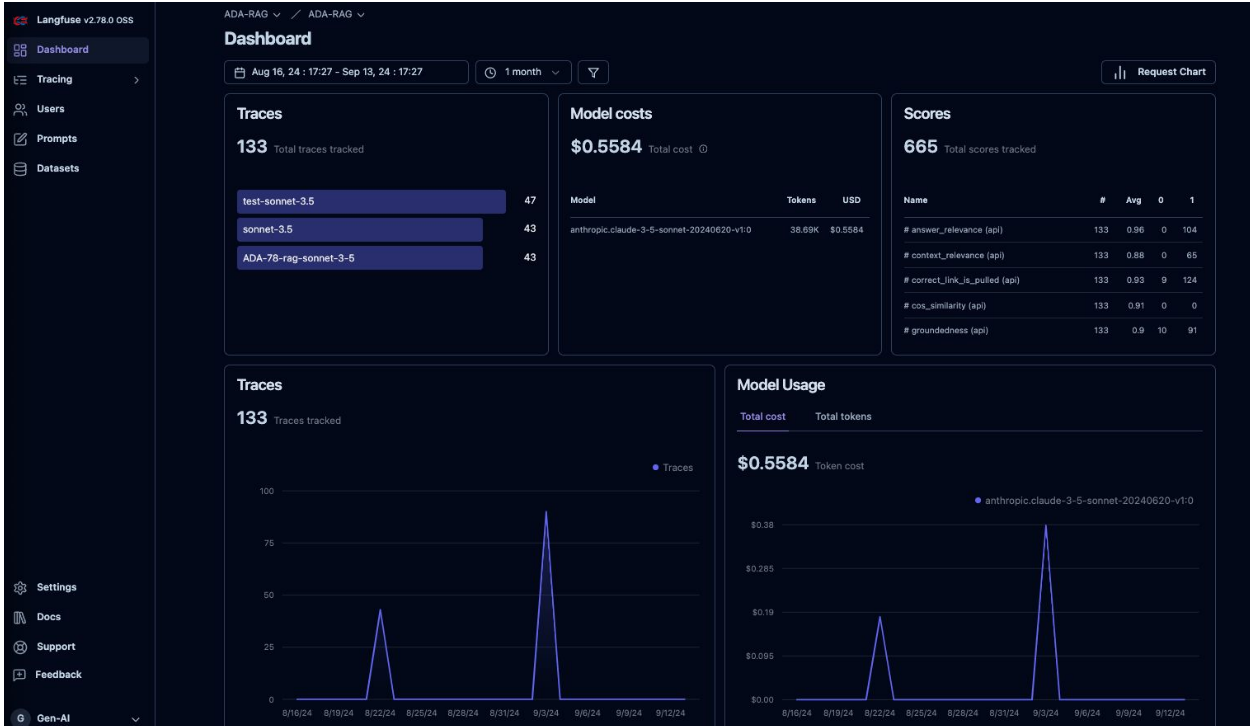
Thank you!
Questions?

leboncoin
tech

leboncoin
tech

RAG evaluation UI

Langfuse - Dashboard



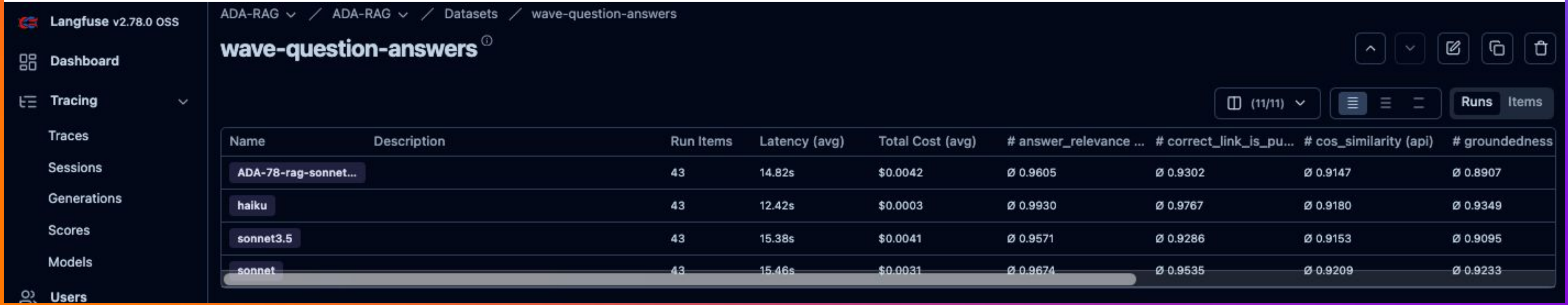
You can see the traces/ model usage for an interval…

But these are evaluation tests, filtering and interacting with the results could be more useful!

RAG evaluation UI

Langfuse

- You can access **Traces**/ **Sessions**/ **Generations**/ **Scores**/ **Prompts** and **Datasets**
- I find **Datasets** the most useful - here I can run many different versions of the same QA pairs and compare different solutions.
- But when you want to compare the results, it's not that easy…



RAG evaluation UI

Langfuse - comparing experiments

- For an individual example it's easy to compare different experiments
- But you can't group experiments together in an easy way, can't order them by increasing scores…
- It stays a very human exercise to translate the findings. To understand why a score was low you'll need to access each generation individually.

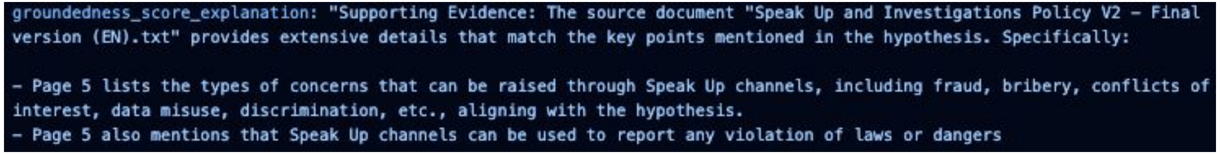
Run At	Dataset Item	Trace	Latency	Cost	# answer_relevance ...	# correct_link_is_pu...	# cos_similarity (api)	# groundedness (api)	# context_relevance...
22/08/2024, 17:49:45	21c9e4a1-7d...		16.73s	$0.0046	1.0000	1.0000	0.9491	1.0000	1.0000
08/08/2024, 12:13:48	21c9e4a1-7d...		10.15s	$0.0002	1.0000	1.0000	0.9295	1.0000	1.0000

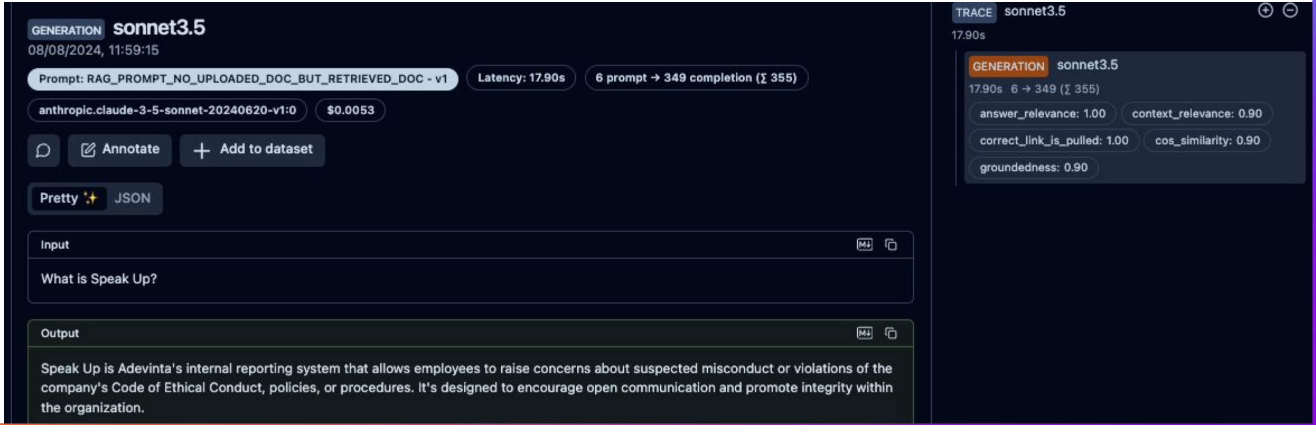
RAG evaluation UI

Langfuse - Individual trace

A trace contains the **question**, the **expected answer**, the **llm generated response** and anything else that you want to add to it:

- the retrieved documents
- the explanation for LLM generated scores
- Prompt





Evaluation with scores

BertScore

ROUGE score measures how similar are the prediction and a possible ground truth but in a naive setup it can easily go wrong:

prediction	groundtruth	
Evaluating LLMs is a very difficult topic	Large Language Models are not easy to assess	Score: 0
Classification models are easy to assess	Large Language Models are not easy to assess	Score: 0.6 (n_gram=1)

wut?

Issues with standard metrics

Why Traditional Metrics Fall Short

Limited Scope, focus on surface-level text similarity:

- Metrics like BLEU and ROUGE **ignore context and meaning.**
- Metrics like BertScore takes into account the semantics but is biased toward a “groundtruth”.
- They tend to be **biased**, correlation between human judgement and some benchmark scores are low.*

Lack of Factual Accuracy Measurement:

- Unable to assess if the generated content is **factually correct or misleading**.
- No Consideration for **User Satisfaction.**
- Doesn’t capture qualitative aspects like **coherence, fluency, or helpfulness**.

Bias Toward Rigid Matching

* Source: Blagec, Dorffner, Moradi: A global analysis of metrics used for measuring performance in natural language processing

RAG Evaluation

The metrics should answer...

Does the prediction answer the question correctly?

Human evaluation

LLM as a judge
(Answer relevance)

Evaluation set with reference answers

Standard scores
(ROUGE, METEOR)

Semantic based scores
(BertScore)

Models/ LLM as a judge

RAG Evaluation

The metrics should answer...

Did we find the right information? (Retrieval metrics)

No “reference context”: LLM as a judge

Pairwise comparison of “reference context” and retrieved context (Precision, recall, F1 score, MRR, nDCG)

Given the retrieved information, did we generate the correct answer? (Generation metrics)

Generation based metrics - LLM metrics

Rule based metrics

LLM as a judge

RAG-Specific Challenges 4.

Hallucinations due to the LLM and or The Knowledge base

The question isn't whether it will come, but rather when.

Therefore a guardrail bloc has to be included in the architecture to avoid harmful or misleading answers

<add schema here>

Use case 2: Markus - ..
...

- Retrieval part is important
- Prompt is very sensitive –
- *If possible fix the problems without changing the prompt.*
- Hallucinations are still a pain
- Non deterministic answers are still a challenge for evaluation and reproducibility
- Documents that are close semantically to the input are not always necessarily the most relevant.
- Constant “surprise” <-> continuous debugging
- You must have guardrails to mitigate those surprises (add scripts to detect hallucinations for example).
- Evaluation frameworks are new and not yet matures

Takeaways and learnings

Evaluating CCA RAG

Process

Off-line evaluation	User testing process	Deploy
Blabla CCA	**Blabla** CCA	**Blabla** CCA

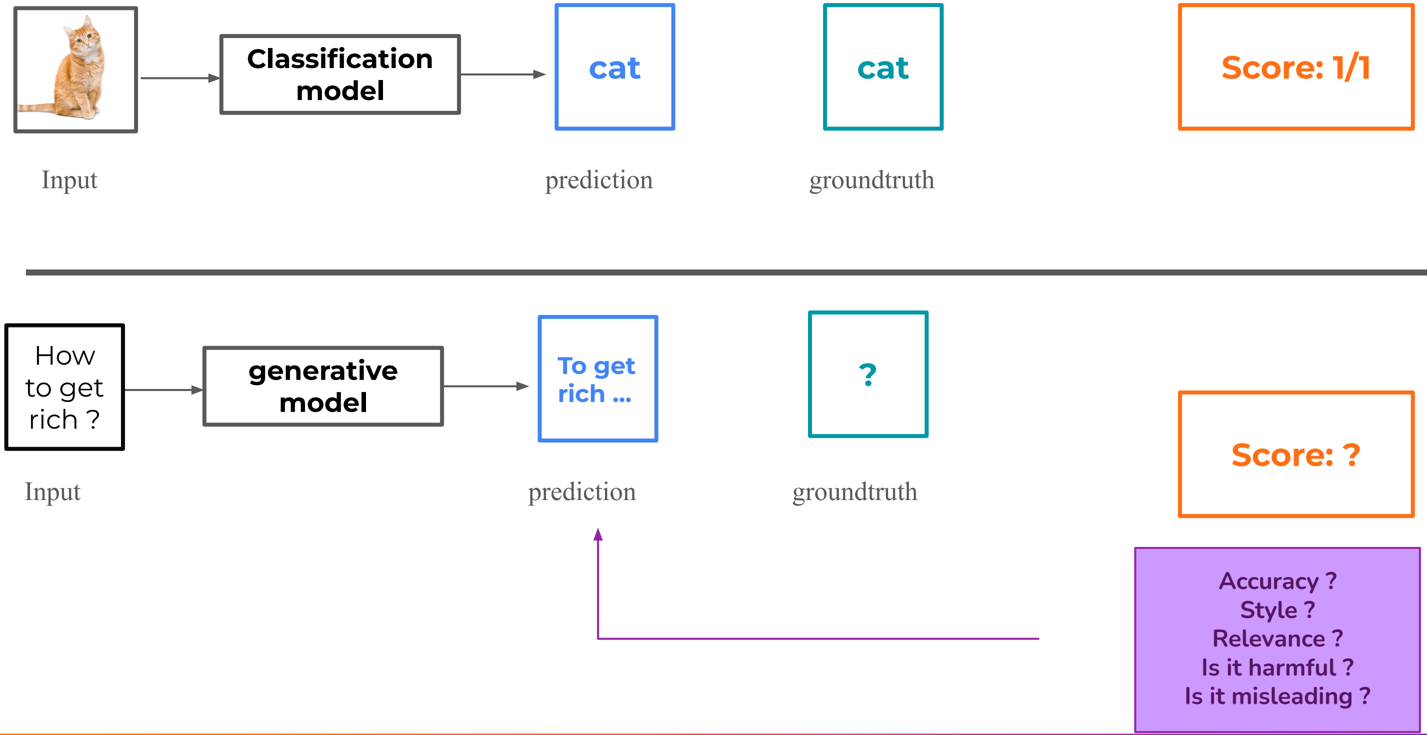
Challenges of LLM Evaluation

Complexities in Evaluating LLMs

- **Output Variability:** Multiple valid responses possible for one input
- **Subjectivity:** Human and model judgments can be inconsistent and biased
- **Context Sensitivity:** Subtle input changes may significantly alter outputs
- **Non-determinism:** Outputs may vary across runs for the same input
- **Continuous Evaluation:** Needed due to constant model improvements

Evaluate something that is generated

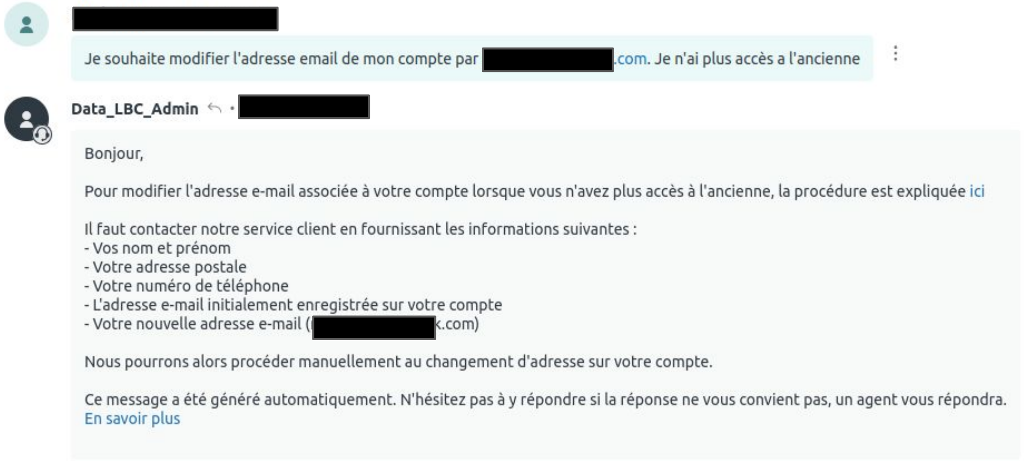
How would you do that ?



Use case 1: Markus - Customer Care Assistant

leboncoin customer service total tickets : ~200k/month

Many of these tickets could be automated using LLMs and leboncoin's FAQ to ease the burden on customer service.


[redacted]

Je souhaite modifier l'adresse email de mon compte par [redacted].com. Je n'ai plus accès a l'ancienne

Data_LBC_Admin · [redacted]

Bonjour,

Pour modifier l'adresse e-mail associée à votre compte lorsque vous n'avez plus accès à l'ancienne, la procédure est expliquée ici

Il faut contacter notre service client en fournissant les informations suivantes :
- Vos nom et prénom
- Votre adresse postale
- Votre numéro de téléphone
- L'adresse e-mail initialement enregistrée sur votre compte
- Votre nouvelle adresse e-mail ([redacted]k.com)

Nous pourrons alors procéder manuellement au changement d'adresse sur votre compte.

Ce message a été généré automatiquement. N'hésitez pas à y répondre si la réponse ne vous convient pas, un agent vous répondra.
En savoir plus

LangChain

FAISS

Amazon Bedrock

Evaluating Ada
Scores used

Rule-based scores: make sure that the answer is in a certain way (dependant of the use case)

The end message should always finish with a caution *"Be aware that I can make mistake. I found the source here…"*

Check if source is in response

BertScore: use reference responses, embed prediction and reference response, compare the similarity

Compare the stability of a response in several runs by comparing their embeddings

Retrieval metrics: check whether correct page is pulled:
precision/ recall/ F1 scores

LLM as a judge scores