Measuring the GenAI Quality. KPI, Metrics

Aleksei Kolesnikov

Staff Software Engineer

Recently I faced an interesting challenge.

How to properly estimate the application quality which uses Generative AI under the hood?

How to properly formulate the question?

How to address this question?

What metrics to use?

What metrics are achievable?

How much will optimization cost?

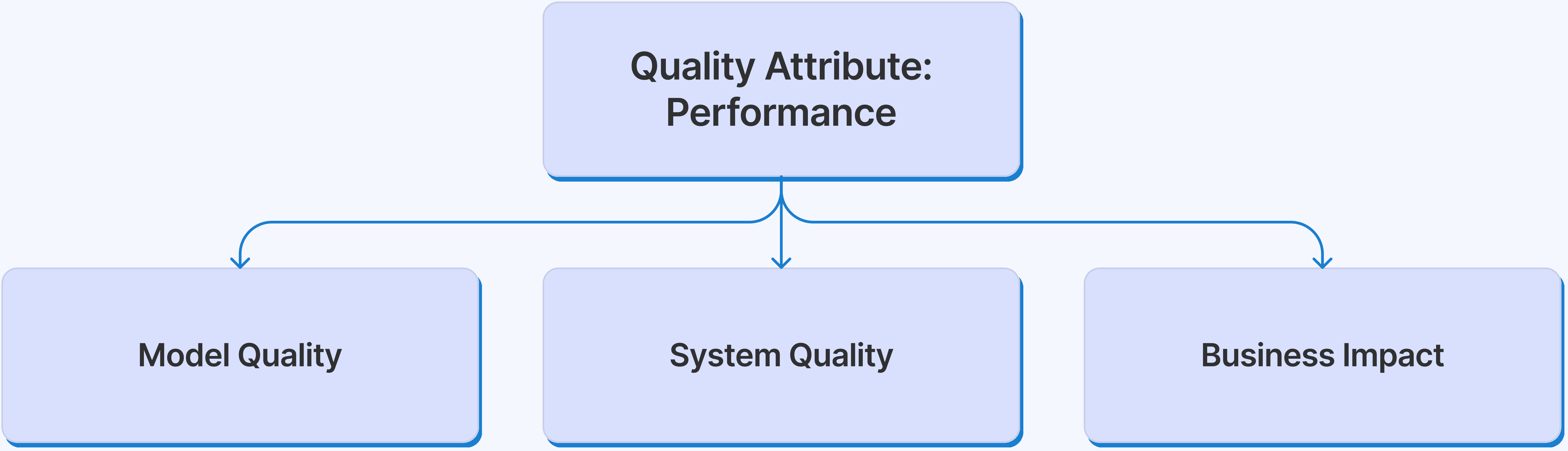
Addressing such a vast subject in a single brief post is a challenge, so let's focus on the "Performance Quality Attribute."

Indeed, GenAI comes with its own peculiarities, but established best practices in architecture can still guide you.

Begin by identifying which quality attributes of your existing system need enhancement. It's crucial to understand the limitations of your system. What are the constraints and sub-attributes we are dealing with?

In this question let's follow Google AI best practices.

Measuring the GenAI Quality.



Quality Attribute: Performance

Model Quality

System Quality

Business Impact

In general, **Model Quality** refers to how well the model can generate responses that are accurate, relevant and contextually sound.

In general, **System Quality** refers to the overall efficiency, effectiveness, and performance of the entire AI system and to the quality of **integrated components:**

→ data acquisition

→ pre-processor

→ post-processor

→ context gathering

→ prompt generation

→ flow orchestration

In general, **Business Impact** of an LLM is its contribution towards achieving the broader goals and objectives of the business

Model Quality

When we talk about Model Quality, we're really talking about how “intelligent” the model is in understanding and responding to its inputs

Model Quality Metrics

Our attention is primarily on the metrics highlighted in bold:

→ Quality index

→ Error Rate

→ **Latency**

→ Accuracy

→ Safety

Business Impact

When we talk about Business impact in the context of LLM, it refers to Applications & Models adoption, how it improves operational efficiency, enhances user experience or drives innovation.

Business Impact Metrics.

Our attention is primarily on the metrics highlighted in bold:

→ **Frequency of use**

→ **Session length**

→ **Queries per session**

→ **Query length**

→ Abandonment rate

→ **User satisfaction**

System Quality

When we talk about System Quality in the context of LLMs, we're discussing the overall performance and reliability of the whole AI system

System Quality Metrics

Our attention is primarily on the metrics highlighted in bold:

→ Data relevance

→ **Data and AI reusability**

→ **Throughput**

→ **System latency**

→ Integration and backward compatibility

Model Quality

Lets touch the simplest scenario.

We figured out that we have performance issues with model itself.

Which model to use then?
How to start Evaluating this?

Need to understand to which scenarios your business flow relates:

- Core scenarios
- NarrativeQA
- NaturalQuestions (open-book)
- NaturalQuestions (closed-book)
- OpenbookQA
- MMLU (Massive Multitask Language Understanding)
- MATH
- GSM8K (Grade School Math)
- LegalBench
- MedQA
- WMT 2014

Holistic Framework for Evaluating Foundation Models (HELM)

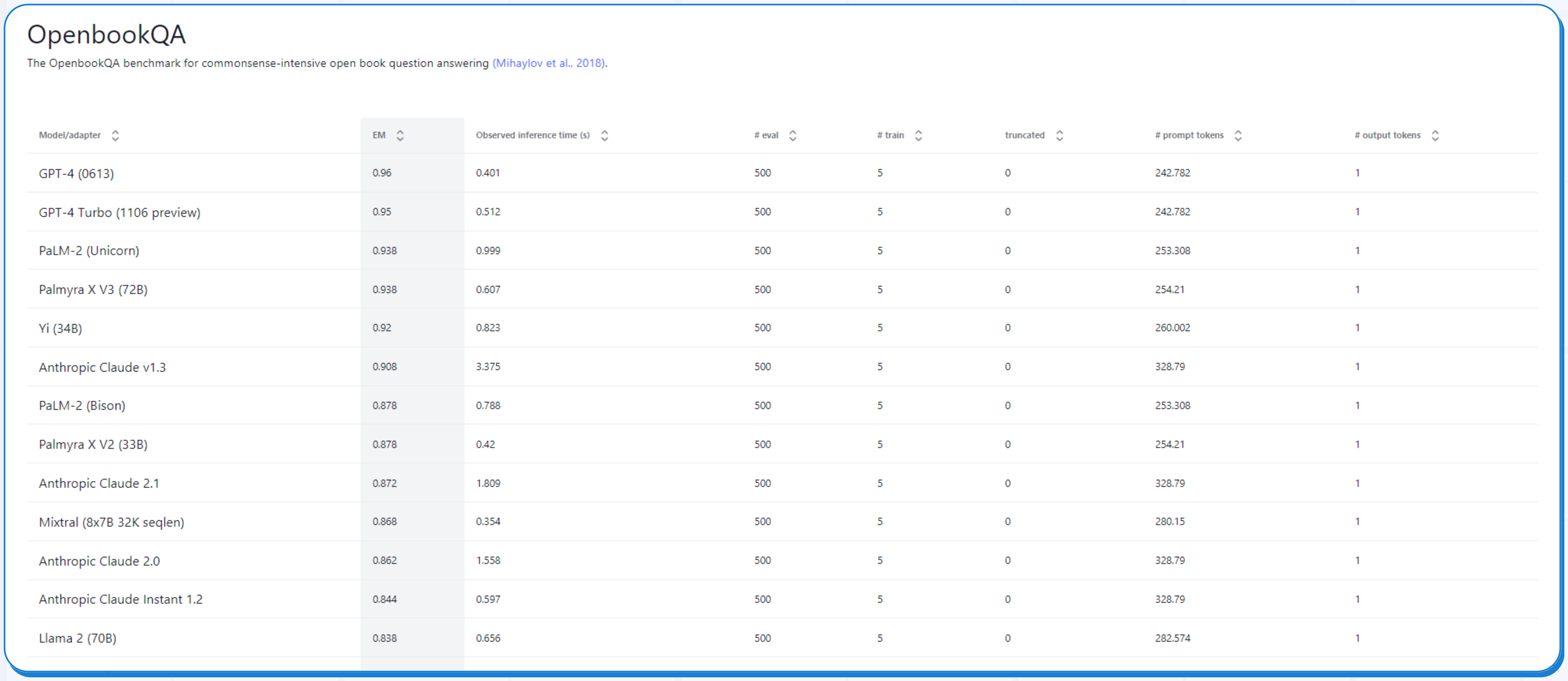
A holistic framework for evaluating foundation models takes into account a broad range of factors that contribute to the overall performance, impact, and value of the model.

1. Technical Evaluation: This includes assessing the model's performance metrics, robustness, reliability, and efficiency.
2. Use-Case Evaluation: This involves examining how effectively the model performs in its intended application or use case, and its adaptability to different scenarios.

Aleksei Kolesnikov
Staff Software Engineer

Holistic Framework for Evaluating Foundation Models (HELM)

There are case studies regarding HELM, for example at Stanford where you can find metrics for all core scenarios with numbers for all most popular models https://crfm.stanford.edu/helm/lite/latest/



OpenbookQA

The OpenbookQA benchmark for commonsense-intensive open book question answering (Mihaylov et al., 2018).

Model/adapter	EM	Observed inference time (s)	# eval	# train	truncated	# prompt tokens	# output tokens
GPT-4 (0613)	0.96	0.401	500	5	0	242.782	1
GPT-4 Turbo (1106 preview)	0.95	0.512	500	5	0	242.782	1
PaLM-2 (Unicorn)	0.938	0.999	500	5	0	253.308	1
Palmyra X V3 (72B)	0.938	0.607	500	5	0	254.21	1
Yi (34B)	0.92	0.823	500	5	0	260.002	1
Anthropic Claude v1.3	0.908	3.375	500	5	0	328.79	1
PaLM-2 (Bison)	0.878	0.788	500	5	0	253.308	1
Palmyra X V2 (33B)	0.878	0.42	500	5	0	254.21	1
Anthropic Claude 2.1	0.872	1.809	500	5	0	328.79	1
Mixtral (8x7B 32K seqlen)	0.868	0.354	500	5	0	280.15	1
Anthropic Claude 2.0	0.862	1.558	500	5	0	328.79	1
Anthropic Claude Instant 1.2	0.844	0.597	500	5	0	328.79	1
Llama 2 (70B)	0.838	0.656	500	5	0	282.574	1

Aleksei Kolesnikov
Staff Software Engineer