Una prospettiva aziendale su come valutare l'IA nei prodotti

Lorenzo Pozzi
Data Scientist
PCO R&D

IMPORTANZA DELLE METRICHE DI VALUTAZIONE

Una metrica di valutazione deve informarci su quanto vicino un prodotto si avvicina alle nostre aspettative ideali.

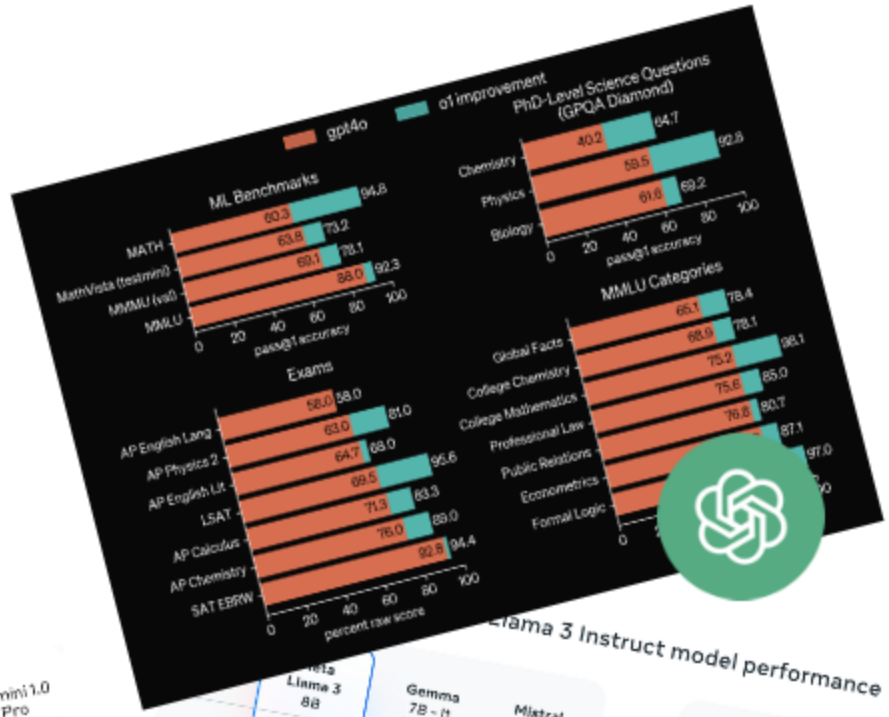
Anche nello sviluppo di prodotti commerciali applichiamo il metodo sicentifico: abbiamo una idea di un prodotto a che livello di qualità deve farlo; quinidi per stabilirne la qualità facciamo degli esperimenti e rappresentiamo il risultato con delle metriche.

Avere delle metriche significa segnare il nostro progresso.

LA SCELTA DI METRICA

Scegliere le guste metriche non è sempre scontato, specialmente su task e problemi complessi, o modelli con più componenti. Capiamo con degli esempi...

Prendiamo gli LLM per esempio. Per ogni nuovo modello, vengono pubblicati i risultati usando dei benchmark "standard".



Llama 3 Instruct model performance

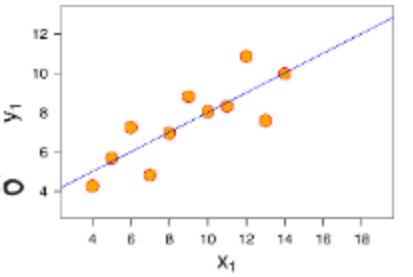
	Meta Llama 3 8B	Gemma 7B - it Measured	Mistral 7B Instruct Measured
MMLU 5-shot	68.4	53.3	58.4
GPQA 0-shot	34.2	21.4	26.3
HumanEval 0-shot	62.2	30.5	36.6
GSM-8K 8-shot, CoT	79.6	30.6	39.9
MATH 4-shot, CoT	30.0	12.2	11.0

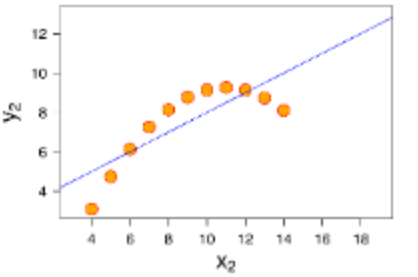
	Meta Llama 3 70B	Gemini Pro 1.5 Published	Claude 3 Sonnet Published
MMLU 5-shot	82.0	81.9	79.0
GPQA 0-shot	39.5	41.5 CoT	38.5 CoT
HumanEval 0-shot	81.7	71.9	73.0
GSM-8K 8-shot, CoT	93.0	91.7 11-shot	92.3 0-shot
MATH 4-shot, CoT	50.4	58.5 Minerva prompt	40.5

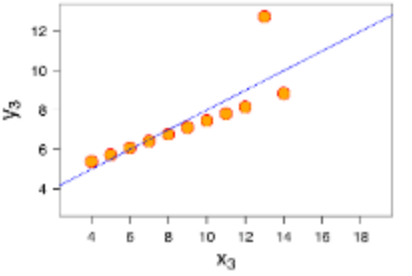
	Claude 3 Opus	Claude 3 Sonnet	Claude 3 Haiku	GPT-4	GPT-3.5	Gemini 1.0 Ultra	Gemini 1.0 Pro
Undergraduate level knowledge MMLU	86.8% 5-shot	79.0% 5-shot	75.2% 5-shot	86.4% 5-shot	70.0% 5-shot	83.7% 5-shot	71.8% 5-shot
Graduate level expertise GPQA Diamond	50.4% 0-shot CoT	40.4% 0-shot CoT	33.3% 0-shot CoT	35.7% 0-shot CoT	28.1% 0-shot CoT	—	—
Grade school math GSM8K	95.0% 0-shot CoT	92.3% 0-shot CoT	88.9% 0-shot CoT	92.0% 5-shot CoT	57.1% 5-shot	94.4% Maj1@32	86.5% Maj1@32
Math problem solving MATH	60.1% 0-shot CoT	43.1% 0-shot CoT	38.9% 0-shot CoT	52.9% 4-shot	34.1% 4-shot	53.2% 4-shot	32.6% 4-shot
Multilingual math MGSM	90.7% 0-shot	83.5% 0-shot	75.1% 0-shot	74.5% 8-shot	—	79.0% 8-shot	63.5% 8-shot
Code HumanEval	84.9% 0-shot	73.0% 0-shot	75.9% 0-shot	67.0% 0-shot	48.1% 0-shot	74.4% 0-shot	67.7 0-s
Reasoning over text DROP, F1 score	83.1 3-shot	78.9 3-shot	78.4 3-shot	80.9 3-shot	64.1 3-shot	82.4 Variable shots	74.1 Variable shots
Mixed evaluations BIG-Bench-Hard	86.8% 3-shot CoT	82.9% 3-shot CoT	73.7% 3-shot CoT	83.1% 3-shot CoT	66.6% 3-shot CoT	83.6% 3-shot CoT	75.0% 3-shot CoT
Knowledge Q&A ARC-Challenge	96.4% 25-shot	93.2% 25-shot	89.2% 25-shot	96.3% 25-shot	85.2% 25-shot	—	—
Common Knowledge HellaSwag	95.4% 10-shot	89.0% 10-shot	85.9% 10-shot	95.3% 10-shot	85.5% 10-shot	87.8% 10-shot	

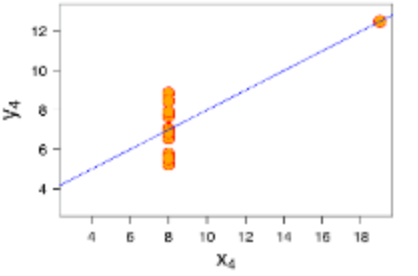
IL QUARTETTO DI AMBSCOBE

Risciamo di cadere in uno scenario descritto dal quartetto di Ambscobe, in cui un modello complesso viene descritto da poche statistiche numeriche, che però non rispecchiano molte altre capacità (ed errori).
La classifica OpenLLM è un esempio dell'enfasi che poniamo su questo tipo di parametri.



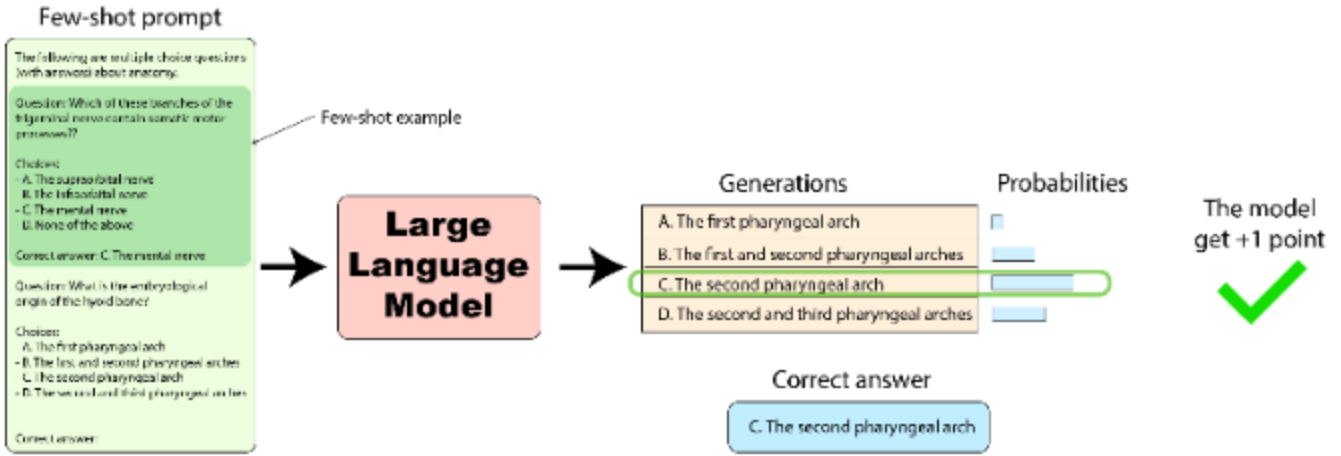






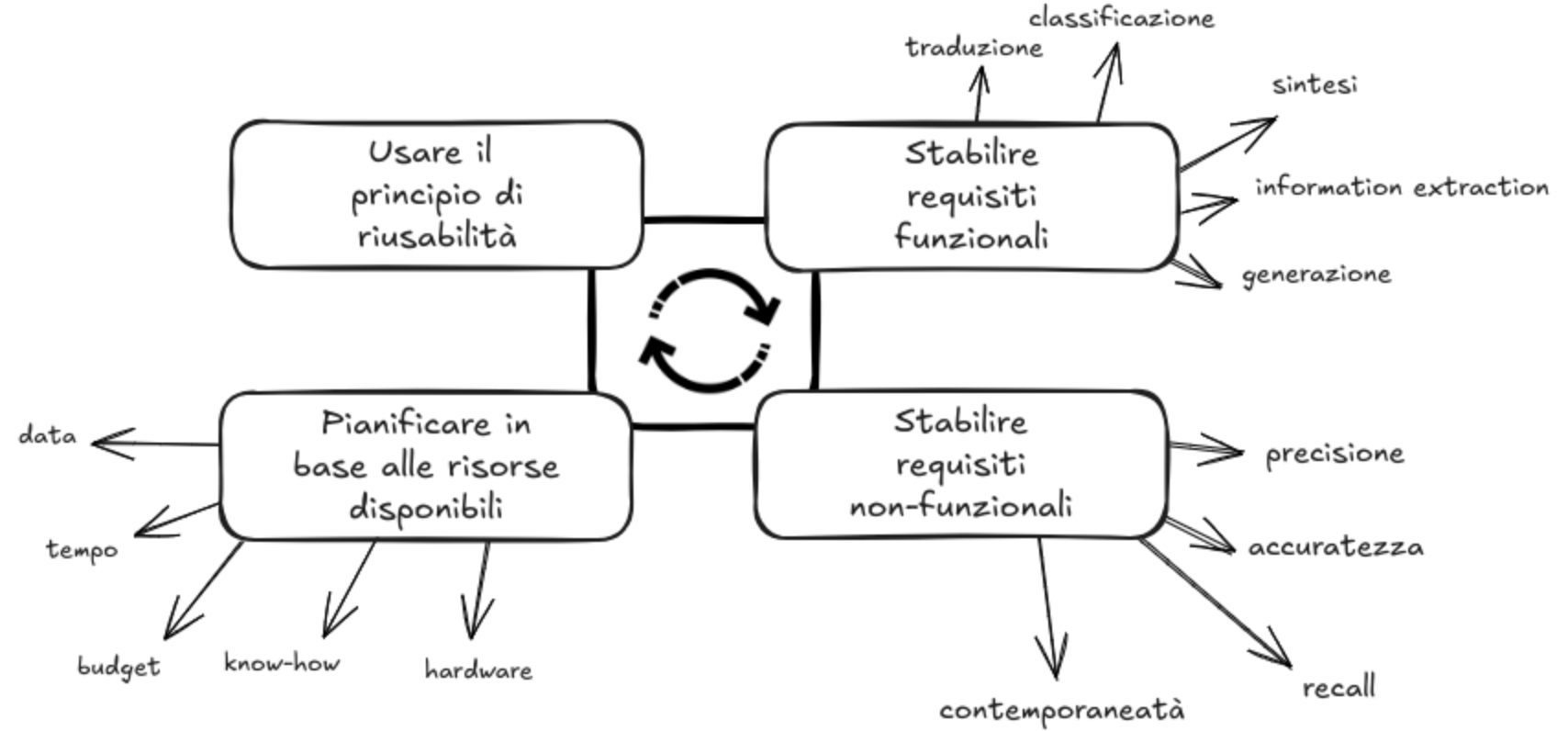
MASSIVE MULTITASKING LANGUAGE UNDERSTANDING

Prendiamo ad esempio MMLU...



Dovrebbe essere chiaro ora che MMLU fornisce poche o nessuna informazione utile su quale modello sia migliore per le applicazioni del mondo reale. Molto raramente abbiamo bisogno di utilizzare LLM per domande a scelta multipla.

TIPI DI METRICHE



IMPORTANZA DELLE METRICHE IN CICD

CI/CD significa integrazione e la distribuzione continue.

L'integrazione continua (CI) consiste nell'integrazione frequente e automatica delle modifiche al codice in un repository condiviso del codice sorgente. La distribuzione continua e/o il deployment continuo (CD) è un processo in due parti durante il quale le modifiche al codice vengono integrate, testate e distribuite. La distribuzione automatica non corrisponde completamente al deployment continuo, che rilascia in automatico gli aggiornamenti nell'ambiente di produzione.

CASO D'USO - RAG

La RAG è un sistema composito. In particolare si divide Retriever e LLM.

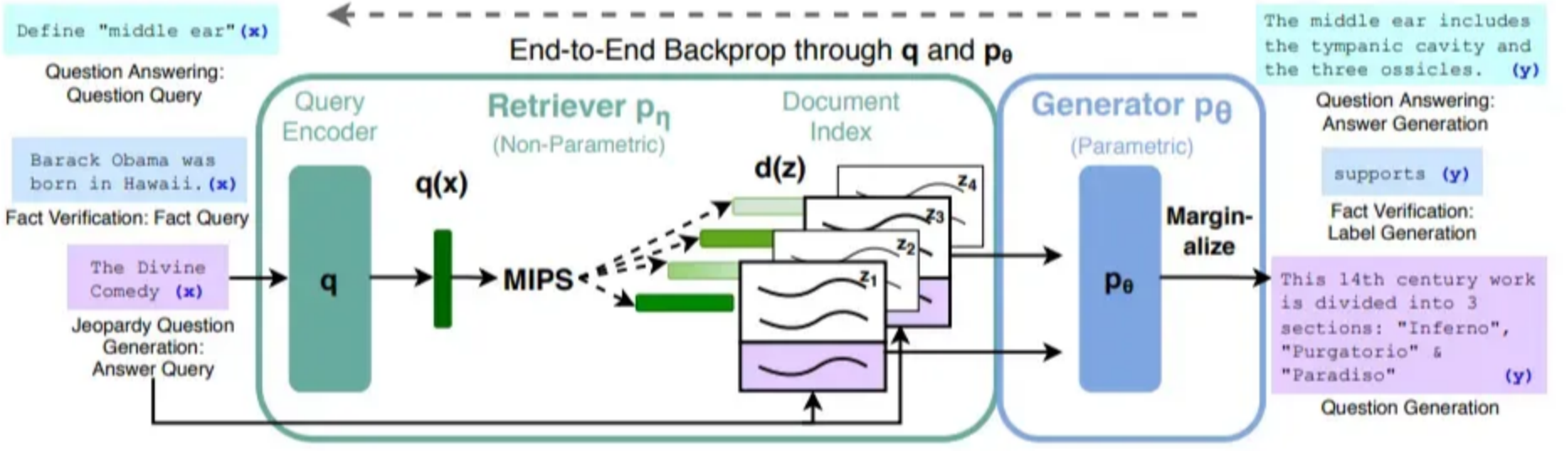
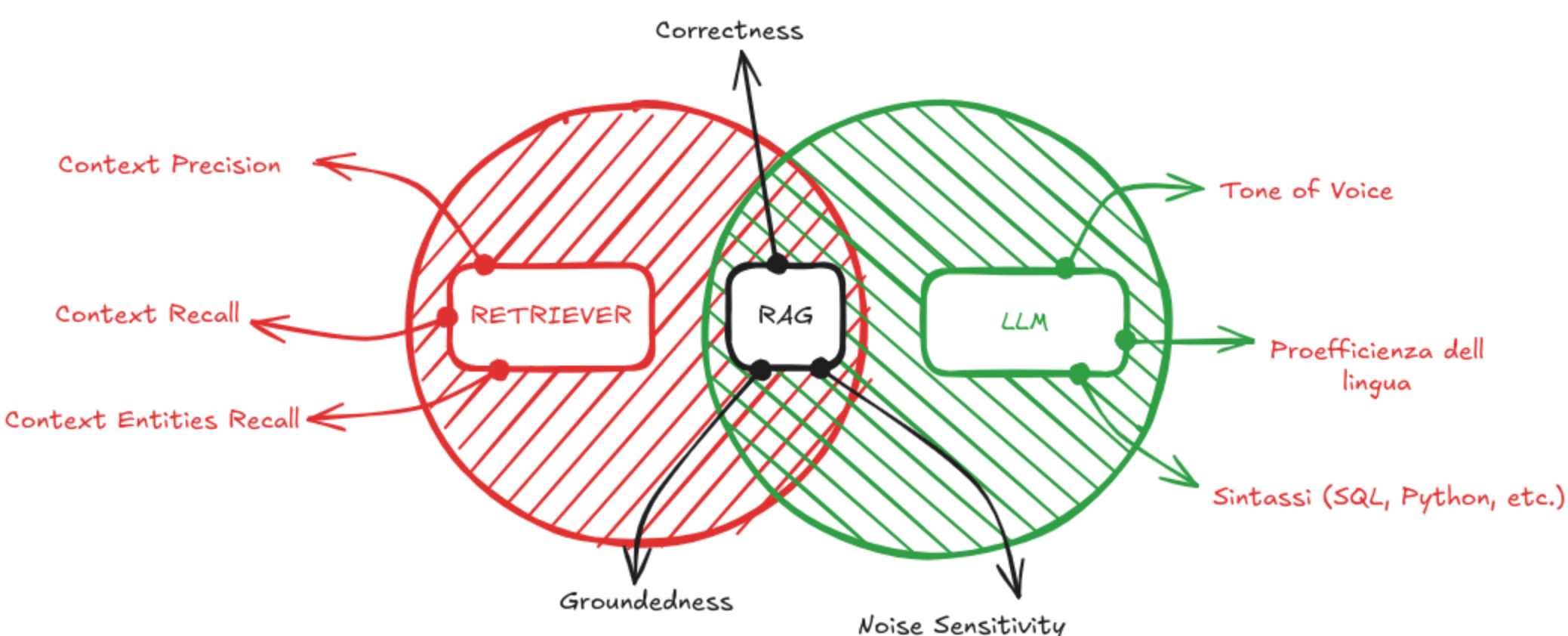


Figure 1: Overview of our approach. We combine a pre-trained retriever (*Query Encoder* + *Document Index*) with a pre-trained seq2seq model (*Generator*) and fine-tune end-to-end. For query x, we use Maximum Inner Product Search (MIPS) to find the top-K documents z_i. For final prediction y, we treat z as a latent variable and marginalize over seq2seq predictions given different documents.

Una valutazione coerente deve tenere conto di questa complessità e isolare i sistemi dove possibile.


Correctness
Context Precision
Context Recall
Context Entities Recall
RETRIEVER
RAG
LLM
Tone of Voice
Proefficienza dell lingua
Sintassi (SQL, Python, etc.)
Groundedness
Noise Sensitivity

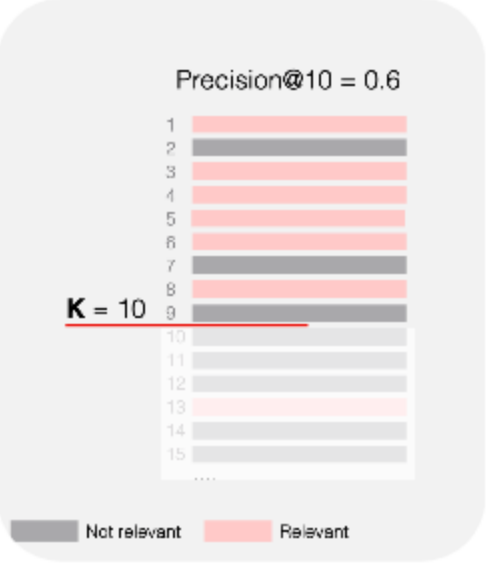
VALUTARE UN RETRIEVER

La CONTEXT PRECISION è una metrica che misura la proporzione di documenti rilevanti nel contesto trovato da retriever. Viene calcolato come media della precision@k per ogni documento nel contesto. Precision@k è il rapporto tra il numero di documenti rilevanti al rango k e il numero totale di documenti al rango k.

$$\text{Context Precision@K} = \frac{\sum_{k=1}^{K} (\text{Precision@k} \times v_k)}{\text{Total number of relevant items in the top } K \text{ results}}$$

$$\text{Precision@k} = \frac{\text{true positives@k}}{(\text{true positives@k} + \text{false positives@k})}$$

Dove K è il numero totale di documenti nel contesto estratto e v_k è l'indicatore di importanza al rango.

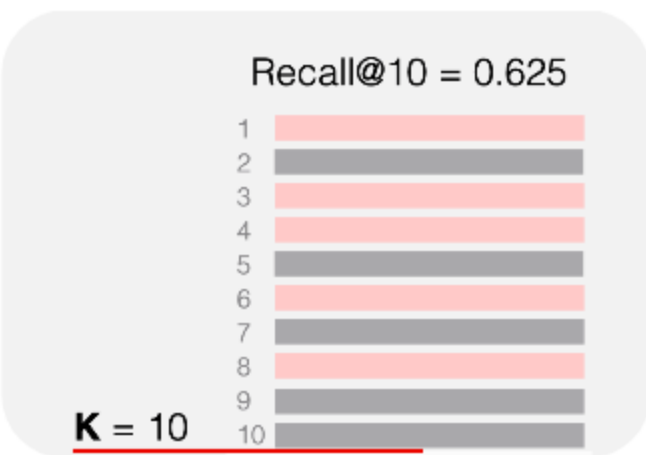


$$CP@K = \frac{1 + 1/2 + 2/3 + \ldots}{10} = 0.66$$

v_k = 1 per ogni k

La CONTEXT RECALL misura quanti documenti rilevanti (o informazioni) sono stati recuperati con successo. Si concentra sul verificare se sono stati persi risultati importanti.

$$\text{context recall} = \frac{|\text{GT claims that can be attributed to context}|}{|\text{Number of claims in GT}|}$$



$$CR = \frac{5}{10} = 0.5$$

VALUTARE UNA RAG

Hallucination-Free? Assessing the Reliability of Leading AI Legal Research Tools

Varun Magesh[*] Stanford University

Faiz Surani[*] Stanford University

Matthew Dahl Yale University

Mirac Suzgun Stanford University

Christopher D. Manning Stanford University

Daniel E. Ho[†] Stanford University

Correttezza: diciamo che una risposta è corretta se è sia fattualmente corretta che pertinente alla domanda. Una risposta è errato se contiene informazioni fattualmente inesatte.

Fondatezza: per le risposte corrette, valutiamo inoltre la fondatezza di ciascuna risposta. La risposta è "fondata" ("grounded") se le affermazioni fattuali chiave nella sua risposta fanno riferimenti validi a documenti pertinenti.

Una risposta è considerato allucinato se è errata o infondato.

COSTI HARDWARE & USER EXPERIENCE

Memory

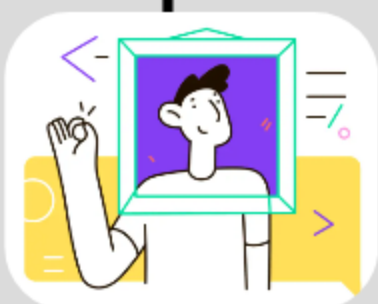
KV Cache Size

Peak Memory

Model Size

First Token Latency

Per-output Token Latency



Latency

Generation Latency

GRAZIE!

+

= open-projects

Pubblichiamo le nostre presentazioni e codice su un repositori open e visibile a tutti.
Siamo sempre alla ricerca di nuovi progetti e idee innovative. Veniteci a trovare!