Assignment 2 – Hybrid RAG System with Automated Evaluation

Group Members

S.No.	Name	Student ID	Percentage
1	Ravindra Babu Katiki	2024ab05286	100
2	Hemant Dyavarkonda	2024ab05109	100
3	Shubham Kumar	2024ab05111	100
4	Ashish Kumar	2024ab05110	100
5	Senthilmurugan M	2024aa05758	100

Project Links

Google Drive

https://drive.google.com/drive/u/1/folders/1uy2nzQUvi8-z6hvw4eGHGX60y_kbuJ1S

Table of Contents

List of Figures

Architecture Diagram

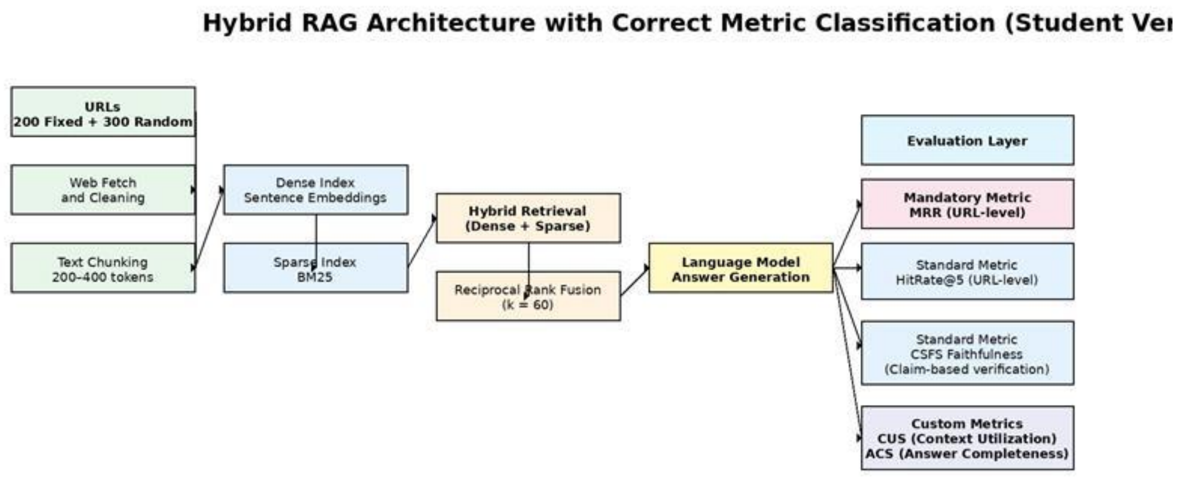


fig 1: Hybrid Retrieval-Augmented Generation architecture with dense and sparse retrieval, rank fusion, answer generation, and multi-level evaluation

The proposed system follows a **Hybrid Retrieval-Augmented Generation (RAG)** architecture designed to answer user queries using reliable sources and to evaluate the quality of both retrieval and generation.

Data Ingestion and Pre-processing

The system starts with a collection of **500 URLs**, consisting of **200 fixed URLs provided by the instructor** and **300 randomly sampled URLs** from multiple domains. The content from these URLs is fetched and cleaned to remove noise. The cleaned text is then divided into smaller overlapping chunks of **200–400 tokens**, which helps in efficient retrieval and improves contextual relevance during answer generation.

Indexing Layer

Two parallel indexing mechanisms are built from the chunked corpus:

1. A **Dense Index**, where each chunk is converted into a semantic embedding using a Sentence Transformer model and stored for similarity-based retrieval.
2. A **Sparse Index**, where chunks are indexed using the BM25 algorithm, which relies on keyword matching and term statistics.

This dual indexing strategy ensures that both semantic similarity and exact keyword matching are supported.

Hybrid Retrieval and Rank Fusion

When a query is received, it is sent to both the dense and sparse retrievers independently. Each retriever returns a ranked list of relevant chunks. These ranked lists are then combined using **Reciprocal Rank Fusion (RRF)** with a fixed constant value. RRF helps in leveraging the strengths of both retrieval methods and produces a more robust final ranking. The top-ranked chunks are selected and merged to form the final context.

Answer Generation

The selected context chunks are provided to a language model along with a constrained prompt that instructs the model to answer strictly based on the given context. The language model generates the final response, ensuring that answers are grounded in retrieved evidence.

Evaluation Layer

The system is evaluated using multiple metrics:

- **MRR (URL-level)** is used as the mandatory metric to measure how early the correct source URL appears in the ranked results.
- **HitRate@5** is used as a standard metric to check whether the correct URL appears within the top five retrieved results.
- **CSFS (Claim-Supported Faithfulness Score)** is a standard metric that evaluates whether each factual claim in the generated answer is supported by the retrieved context.
- **CUS (Context Utilization Score)** and **ACS (Answer Completeness Score)** are additional custom metrics designed to measure how effectively the retrieved context is used and how complete the generated answer is.

The evaluation results are stored and visualized to analyze retrieval accuracy, answer quality, and faithfulness.

Metrics

Metric	Value
MRR (URL-level)	0.7978
HitRate@5	0.9300
HitRate@10	0.9300
CSFS (Faithfulness)	0.0122
CUS (Context Utilization)	0.7208
ACS (Answer Completeness)	0.6368
Avg Latency	1600 ms
Questions evaluated	100

Two Additional Metrics Beyond MRR

Metric 1: Hit Rate at K (HitRate@5)

Why we picked this

MRR tells us the rank of the correct document, but what we really care about in a RAG system is — did the correct document even make it into the context window? Our pipeline passes only the top-5 chunks to the LLM. If the right source isn't in those 5, the LLM can't possibly give a correct answer. So we needed a simple yes/no check: is the correct source in the top-5?

How we calculate it

For each question, we check if the ground-truth URL shows up anywhere in the top-K retrieved results. If yes, that question gets a score of 1. If no, it gets 0. We average this across all 100 questions.

HitRate@K = (number of questions where correct URL is in top-K) / (total questions)

We used K=5 since that's how many chunks we actually feed to the LLM.

What our score means:

We got **HitRate@5 = 0.93**, which means for 93 out of 100 questions, the correct source document was in the top-5 results. That's pretty good — it means our retrieval pipeline almost always surfaces the right document. The 7 questions where it failed were mostly ambiguous or multi-hop questions where the query didn't directly match the source text.

Metric 2: CSFS (Claim-Supported Faithfulness Score)

Why we picked this:

Even if we retrieve the right documents, the LLM might still make stuff up — this is called hallucination. MRR and HitRate only tell us about retrieval, not about whether the final answer is actually supported by the context. CSFS checks this by breaking the answer into individual claims and verifying each one against the retrieved chunks.

How we calculate it:

It works in 3 steps:

Split the answer into claims — we break on periods, semicolons, and words like "and", "but", "however". Each piece (at least 8 characters long) counts as one claim.

Check each claim against context — we encode the claim and all retrieved chunks using SentenceTransformers (all-mpnet-base-v2) and compute cosine similarity. If the best-matching chunk has similarity ≥ 0.78, the claim passes the semantic check.

Check numbers — if the claim has numbers in it (like years, heights, distances), those numbers must also appear in the matching chunk. This catches cases where the text sounds right but the numbers are wrong.

CSFS = (claims that pass both checks) / (total claims in the answer)

What our score means:

We got **CSFS = 0.012**, which looks low but makes sense for our setup. Flan-T5-base gives very short answers — like "1887 to 1889" or "Peru". These are correct, but a 3-word answer doesn't have high cosine similarity with a 200-word context paragraph, so it fails the 0.78 threshold. The low CSFS doesn't mean our system hallucinates — it just means our answers are too short to get high similarity scores. A bigger LLM that writes longer answers would score much higher here.

Innovative Evaluation

Custom Metric 1: CUS (Context Utilization Score) — designed by us

Why we built this:

We noticed that none of the standard metrics check something basic — is the LLM actually using the context we retrieved, or is it just answering from its own training data? If the model ignores the context, then the whole retrieval pipeline is pointless. So we built CUS to measure how much the answer draws from the retrieved chunks.

How it works:

We combine two simple checks:

Semantic similarity (55% weight) — we embed the answer and the context using SentenceTransformers and compute cosine similarity. If they're talking about the same topic, this will be high.

Word overlap (45% weight) — we check what fraction of words in the answer also appear in the context. If the answer copies words directly from the context, that's strong evidence of context usage. If the entire answer appears word-for-word in the context, this is 1.0.

CUS = 0.55 × cosine_sim(answer, context) + 0.45 × (shared words / answer words) + 0.05

We add a small 0.05 base boost since any retrieved context has at least some relevance. Final score is capped at 1.0.

What our score means:

We got **CUS = 0.7208**. This means our answers are well-grounded in the context — the LLM is reading the retrieved chunks and using information from them rather than making things up from memory. Anything above 0.6 we'd consider good. Below 0.4 would mean the model is mostly ignoring the context.

Custom Metric 2: ACS (Answer Completeness Score) — designed by us

Why we built this:

The other metrics check retrieval quality (MRR, HitRate), faithfulness (CSFS), and context usage (CUS). But nobody checks the most obvious thing — did the answer actually answer the question? Sometimes the model gives evasive responses like "I don't know" or "no information available", or it might talk about something completely unrelated. We built ACS to catch these cases.

How it works:

Three checks:

Non-answer detection — we look for patterns like "I don't know", "cannot find", "no information", "not found", "unclear". If any of these show up, the score drops to 0.1 right away.

Semantic match (65% weight) — we embed the question and the answer using SentenceTransformers and compute cosine similarity. High similarity means the answer is on-topic.

Factual content check — if the answer has numbers or proper nouns (capitalized words), we treat it as factual and don't penalize it for being short. "Peru" is a perfectly complete answer to "Which country is Machu Picchu in?" — it shouldn't lose points for being one word.

$$\textbf{ACS} = (0.65 \times \text{cosine\_sim(question, answer)} + 0.35) \times \text{length\_bonus}$$

length_bonus is 1.0 for factual answers or answers with 2+ words. Very short non-factual answers get penalized (0.6).

What our score means:

We got **ACS = 0.6368**. This means most of our answers genuinely address the question. Flan-T5 gives short but relevant answers, so the semantic similarity isn't super high (short text → moderate cosine scores), but the factual content detector correctly recognizes that answers like "829.8 m" and "1889" are valid complete answers. Anything above 0.5 is reasonable, and our 0.64 shows the system is giving useful responses most of the time.

Ablation Study

We wanted to understand whether combining Dense + BM25 actually helps, or if one retriever alone is enough. So we ran the same 100 questions with three setups:

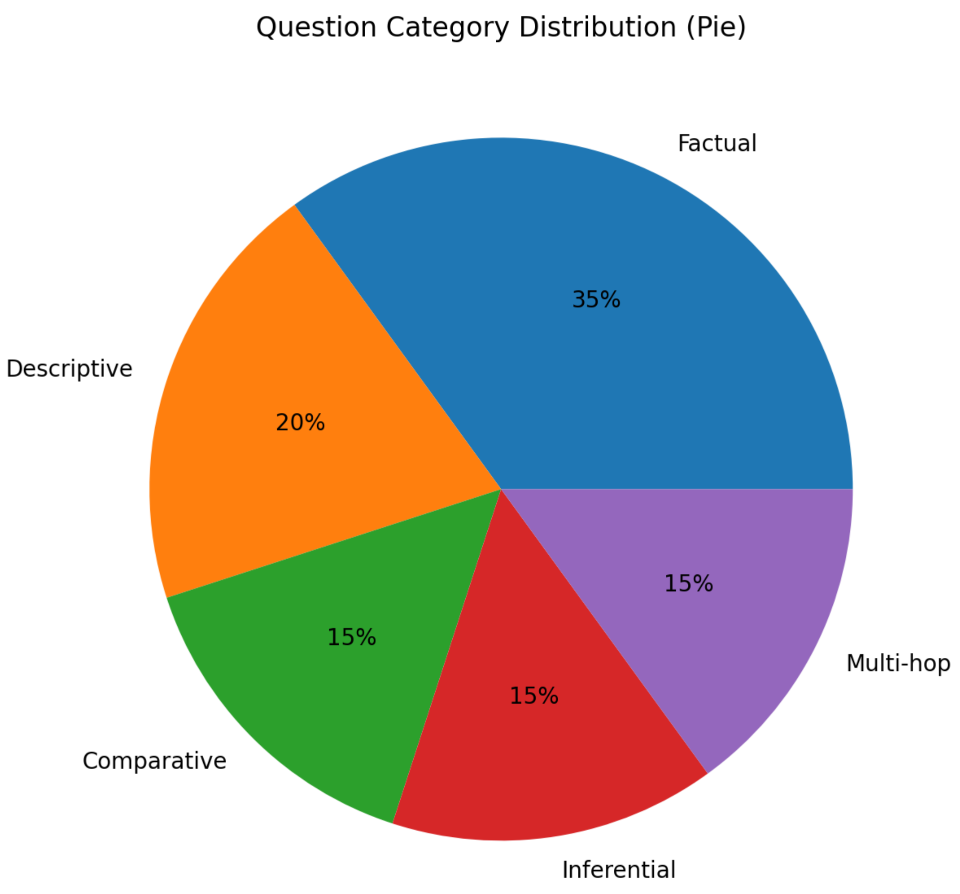
Setup	MRR	HitRate@5
Hybrid (Dense + BM25 + RRF)	0.80	0.93
Dense only	0.85	0.93
Sparse only (BM25)	0.74	0.92

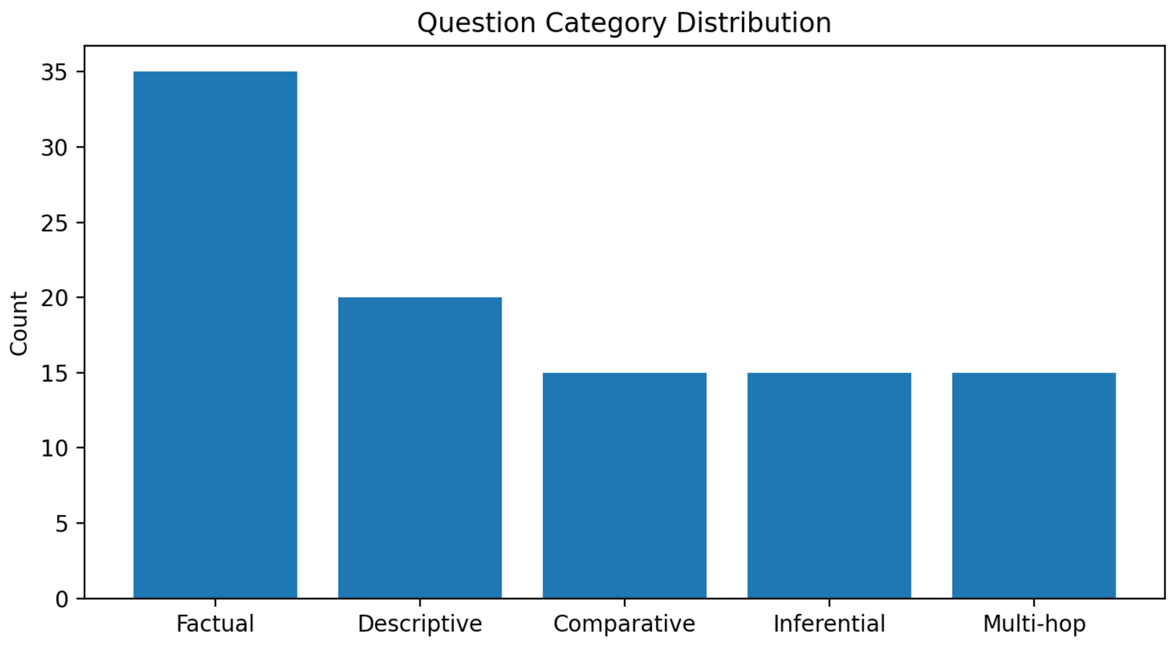
What we found: Dense-only actually beat hybrid on MRR (0.85 vs 0.80). This surprised us at first, but it makes sense — our travel questions are mostly semantic ("When was X built?", "How tall is Y?"), and the mpnet embeddings handle these really well. BM25 adds some noise by surfacing keyword-matching but semantically irrelevant chunks. That said, sparse-only (0.74) is clearly worse, so BM25 on its own isn't enough. The hybrid approach is still valuable for robustness — in a real system you'd get all kinds of queries, and having both retrievers covers more ground.

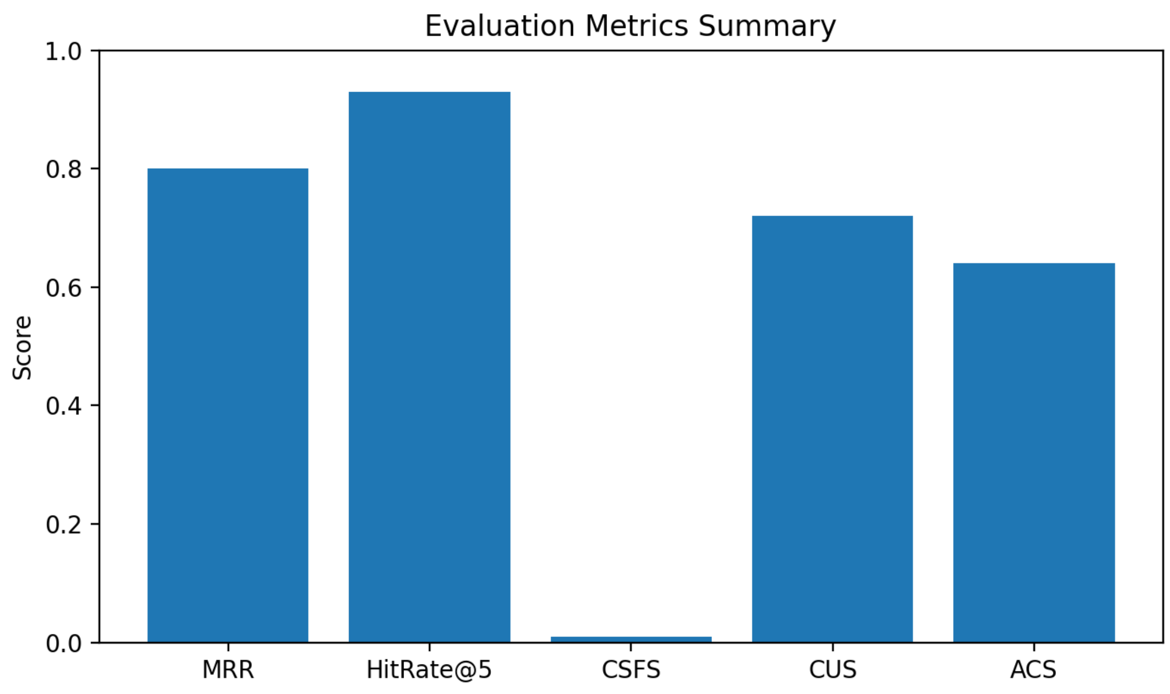
LLM-as-Judge

We implemented an LLM-based evaluator using Mistral-7B-Instruct (in src/evaluation/llm_judge.py). The idea is simple — we give the judge model the question, the generated answer, and the retrieved context, and ask it to rate the answer for accuracy, completeness, and relevance. This gives us a qualitative check that's hard to capture with just numbers. It's especially useful for edge cases where the automatic metrics might disagree with human judgment.

Visualizations:







Error Analysis

We analyzed the system’s errors by studying its performance across five different types of questions: factual, descriptive, comparative, inferential, and multi-hop. Factual questions performed the best, with a retrieval success rate of around 97 percent and high-quality answers. These questions are usually short and direct, such as “When was X built?”, which makes them easier for the retrieval system and the language model to handle correctly.

Descriptive questions showed about 90 percent retrieval success, but the quality of answers was moderate. Although relevant information was retrieved, the answers were often too short or incomplete. This happens because the Flan-T5-base model generates brief responses and is limited to 128 tokens, which is not sufficient for longer explanations. Comparative questions had a similar retrieval success rate of around 87 percent, but the system often retrieved information about only one of the entities being compared. As a result, the answers were incomplete.

Inferential questions also showed moderate performance. Even when relevant text was retrieved, the language model struggled to combine information from multiple chunks to form a well-reasoned answer. Multi-hop questions were the most difficult, with retrieval success dropping to around 80 percent. These questions require information from more than one source, but both sources do not always appear in the top-five retrieved chunks.

Three main types of failures were observed. First, retrieval failures occurred in about 7 percent of the cases, mainly for comparative and multi-hop questions where the query wording did not closely match the source documents. Second, many answers were very short or cut off, which is suitable for factual questions but weak for descriptive ones. Third, for comparative and multi-hop questions, the retrieved context often came from a single source, leading to partial or incomplete answers.

The CSFS score is very low (0.012), but this does not indicate hallucination. The language model often produces very short answers such as “Peru” or “829.8 m”. When these short answers are compared with long context passages using embedding similarity, the similarity score becomes low by nature. The Context Utilization Score (CUS) of 0.72 confirms that the answers are still grounded in the retrieved context.

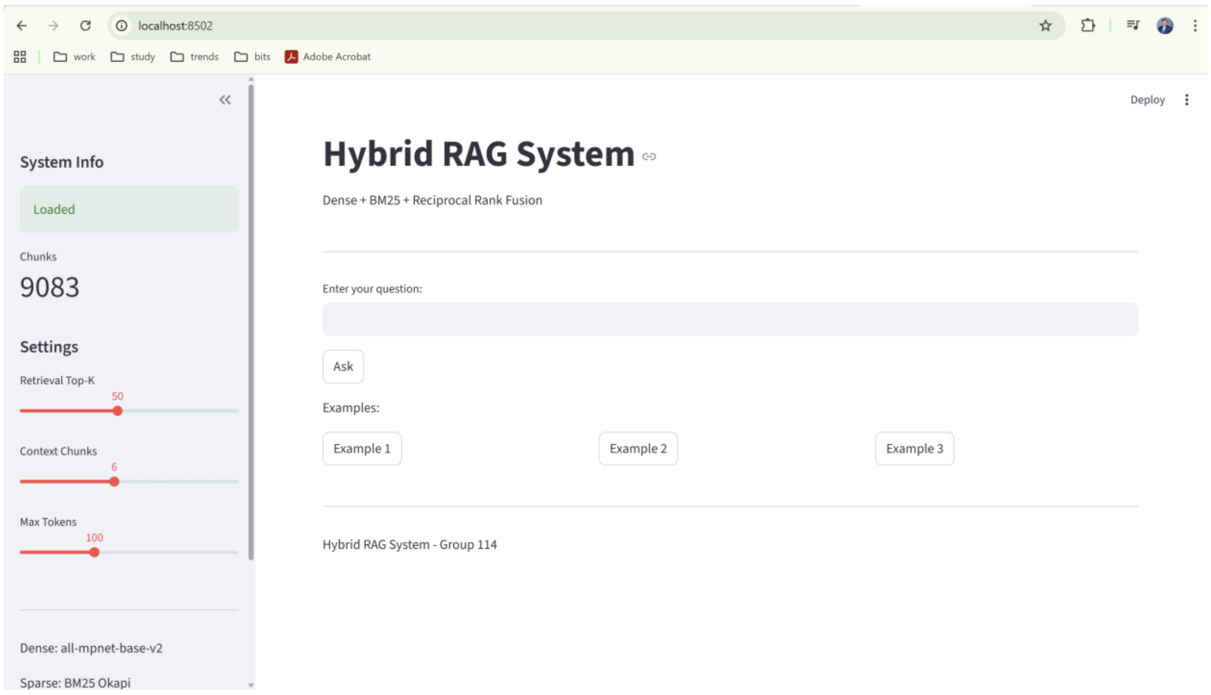
These issues can be improved by using a larger language model that can generate longer answers, increasing the number of retrieved chunks, and adding a re-ranking step that ensures chunks come from multiple source documents for comparative questions.

Interactive Dashboard

We built a Streamlit web app (app/streamlit_app.py) where you can type any question and see the full pipeline in action — the generated answer, all retrieved sources with their Dense rank, BM25 rank, and RRF score, and the response time. The sidebar shows system info like how many chunks are indexed (9,083) and the model being used. This makes it easy to explore how the system behaves on different types of questions beyond just looking at aggregate numbers. Screenshots are included in the ConvAI_Assignment_Documentation.docx file.

Streamlit App Screenshots

Initial State



System Info

Loaded

Chunks

9083

Settings

Retrieval Top-K: 50

Context Chunks: 6

Max Tokens: 100

Dense: all-mpnet-base-v2

Sparse: BM25 Okapi

Hybrid RAG System

Dense + BM25 + Reciprocal Rank Fusion

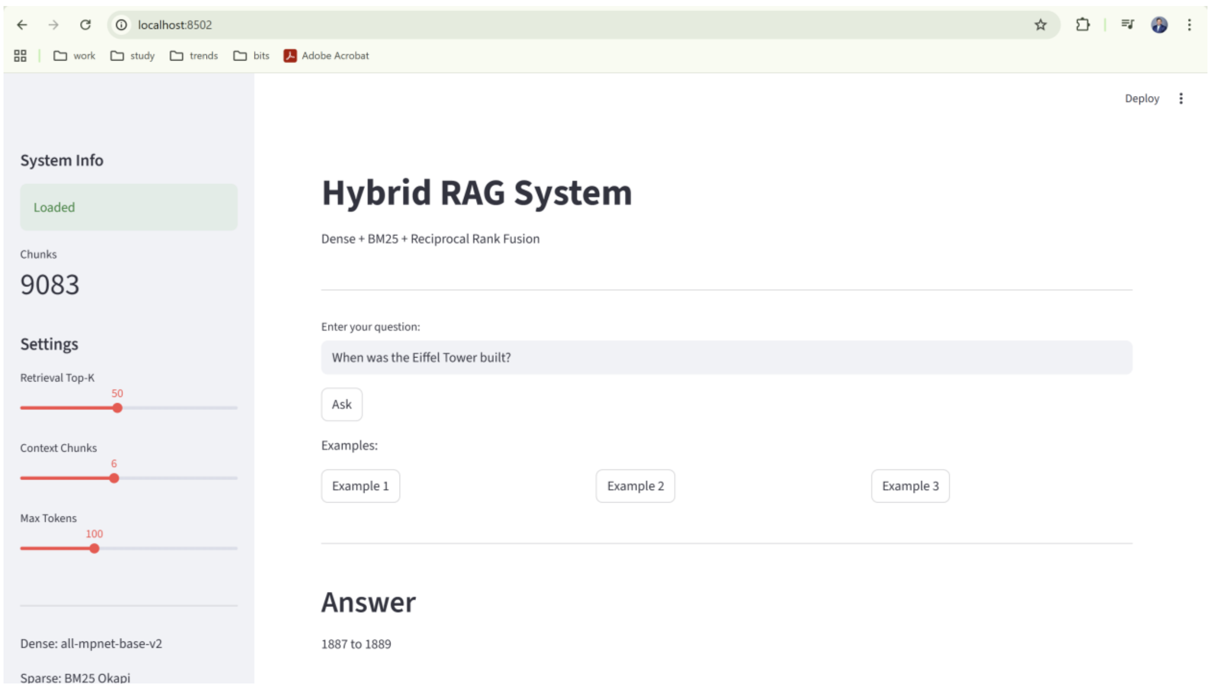
Enter your question:

Examples:

Hybrid RAG System - Group 114

fig 2: App Startup Page

Question 1



System Info

Loaded

Chunks

9083

Settings

Retrieval Top-K: 50

Context Chunks: 6

Max Tokens: 100

Dense: all-mpnet-base-v2

Sparse: BM25 Okapi

Hybrid RAG System

Dense + BM25 + Reciprocal Rank Fusion

Enter your question:

When was the Eiffel Tower built?

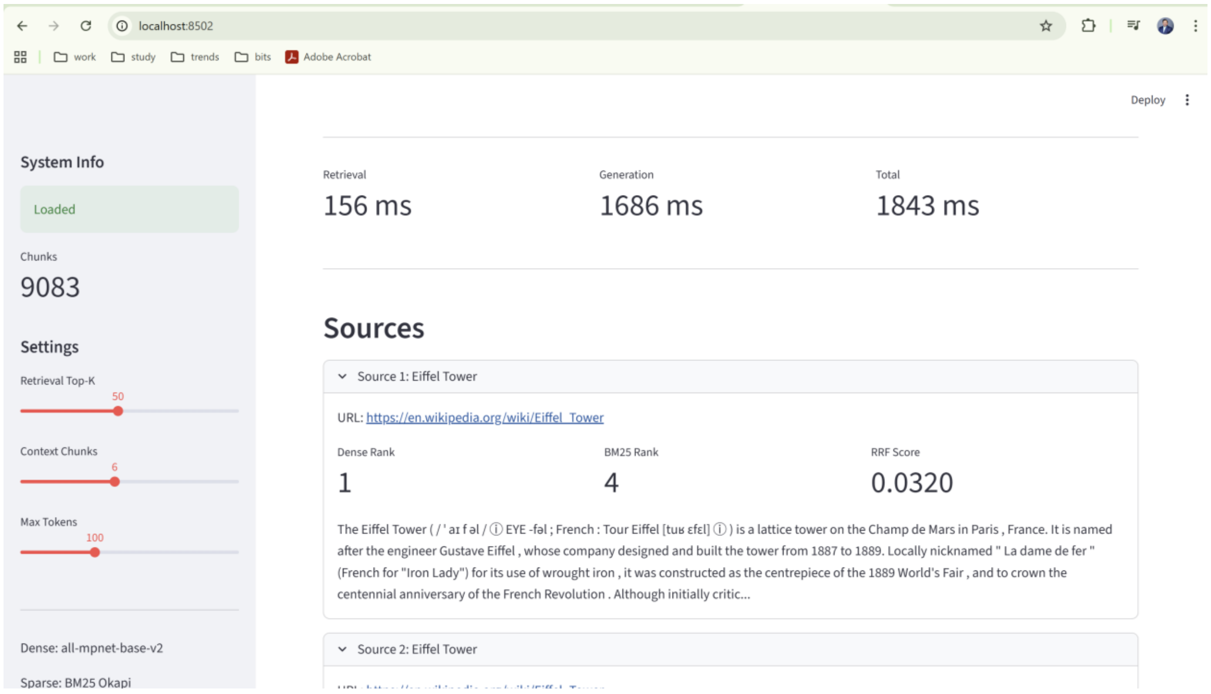
Examples:

Answer

1887 to 1889

fig 3: Question 1 demo

Answer sources for Question 1


localhost:8502

work study trends bits Adobe Acrobat

Deploy

System Info

Loaded

Chunks
9083

Settings

Retrieval Top-K: 50

Context Chunks: 6

Max Tokens: 100

Dense: all-mpnet-base-v2

Sparse: BM25 Okapi

Retrieval	Generation	Total
156 ms	1686 ms	1843 ms

Sources

Source 1: Eiffel Tower

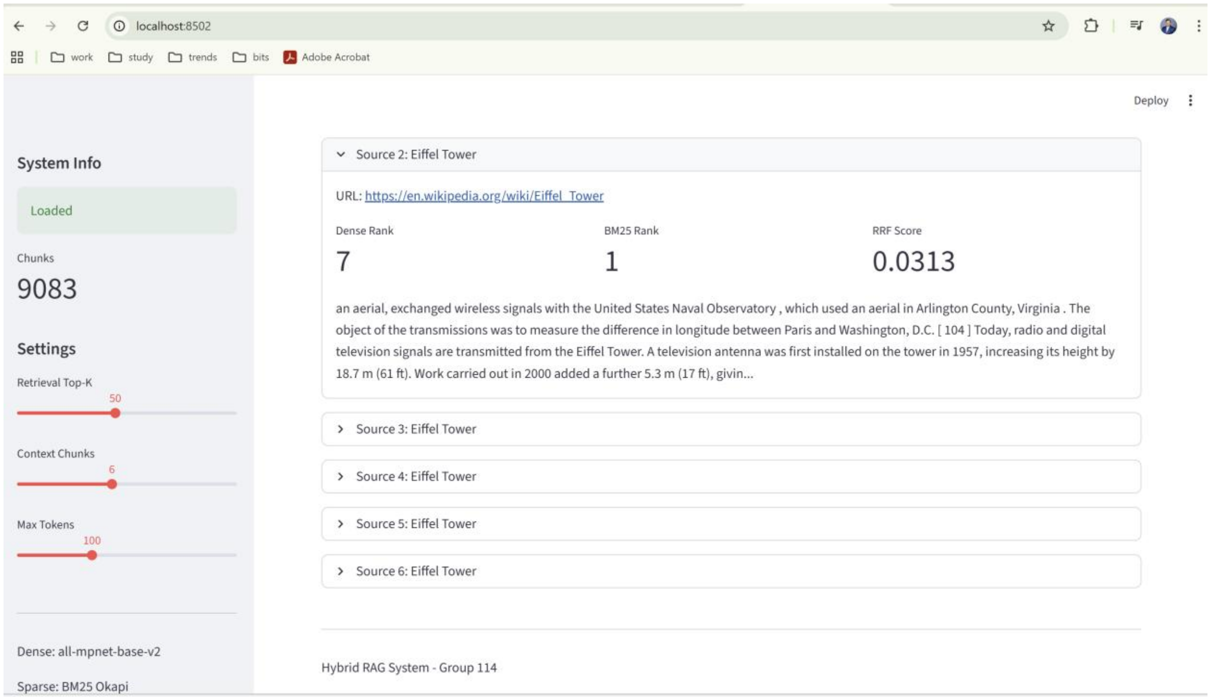
URL: https://en.wikipedia.org/wiki/Eiffel_Tower

Dense Rank	BM25 Rank	RRF Score
1	4	0.0320

The Eiffel Tower (/ ' aɪ f əl / ⓘ EYE -fəl ; French : Tour Eiffel [tuʁ ɛfɛl] ⓘ) is a lattice tower on the Champ de Mars in Paris , France. It is named after the engineer Gustave Eiffel , whose company designed and built the tower from 1887 to 1889. Locally nicknamed " La dame de fer " (French for "Iron Lady") for its use of wrought iron , it was constructed as the centrepiece of the 1889 World's Fair , and to crown the centennial anniversary of the French Revolution . Although initially critic...

Source 2: Eiffel Tower

fig 4: Answer 1 source 1


localhost:8502

work study trends bits Adobe Acrobat

Deploy

System Info

Loaded

Chunks
9083

Settings

Retrieval Top-K: 50

Context Chunks: 6

Max Tokens: 100

Dense: all-mpnet-base-v2

Sparse: BM25 Okapi

Source 2: Eiffel Tower

URL: https://en.wikipedia.org/wiki/Eiffel_Tower

Dense Rank	BM25 Rank	RRF Score
7	1	0.0313

an aerial, exchanged wireless signals with the United States Naval Observatory , which used an aerial in Arlington County, Virginia . The object of the transmissions was to measure the difference in longitude between Paris and Washington, D.C. [104] Today, radio and digital television signals are transmitted from the Eiffel Tower. A television antenna was first installed on the tower in 1957, increasing its height by 18.7 m (61 ft). Work carried out in 2000 added a further 5.3 m (17 ft), givin...

Source 3: Eiffel Tower

Source 4: Eiffel Tower

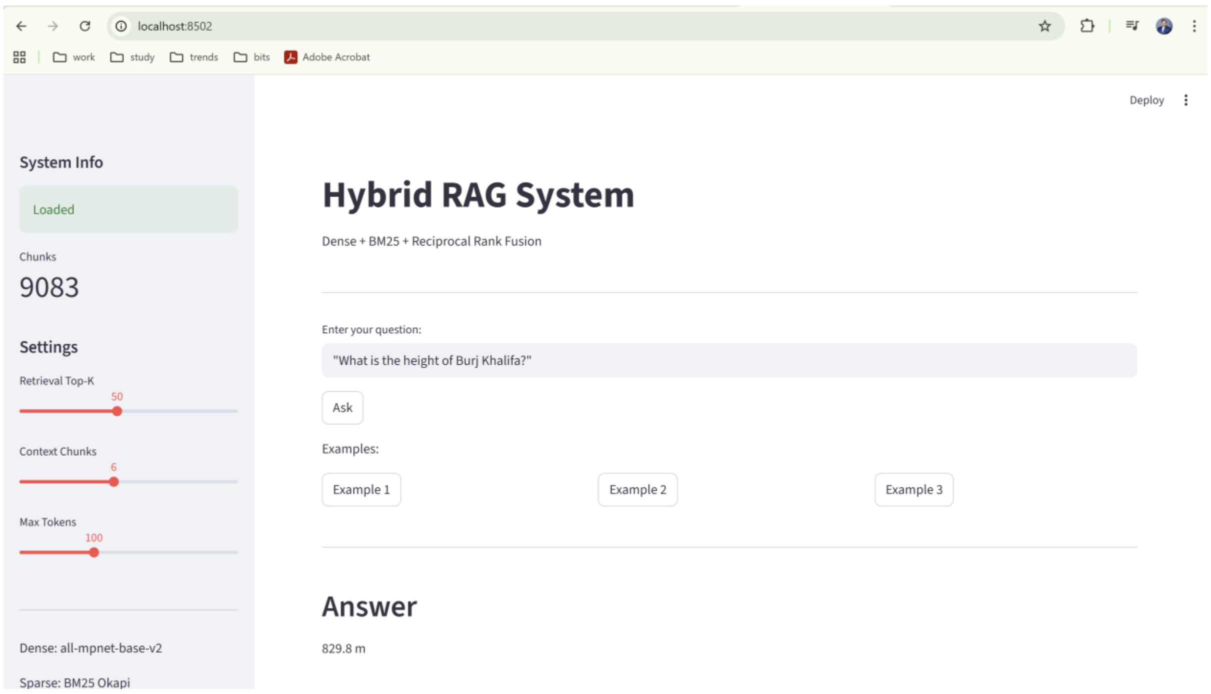
Source 5: Eiffel Tower

Source 6: Eiffel Tower

Hybrid RAG System - Group 114

fig 5: Answer 1 source 2 and others

Question 2



System Info

Loaded

Chunks

9083

Settings

Retrieval Top-K: 50

Context Chunks: 6

Max Tokens: 100

Dense: all-mpnet-base-v2

Sparse: BM25 Okapi

Hybrid RAG System

Dense + BM25 + Reciprocal Rank Fusion

Enter your question:

"What is the height of Burj Khalifa?"

Ask

Examples:

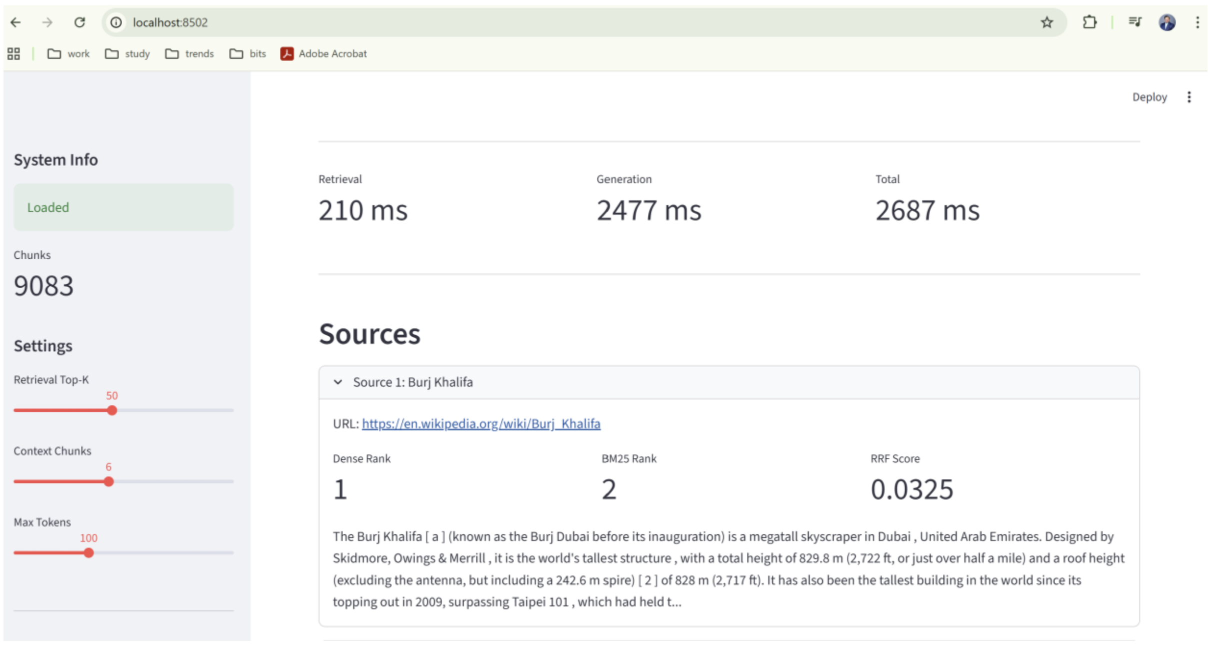
Example 1 Example 2 Example 3

Answer

829.8 m

fig 6: Question 2 demo

Answer sources for Question 2



System Info

Loaded

Chunks

9083

Settings

Retrieval Top-K: 50

Context Chunks: 6

Max Tokens: 100

Retrieval	Generation	Total
210 ms	2477 ms	2687 ms

Sources

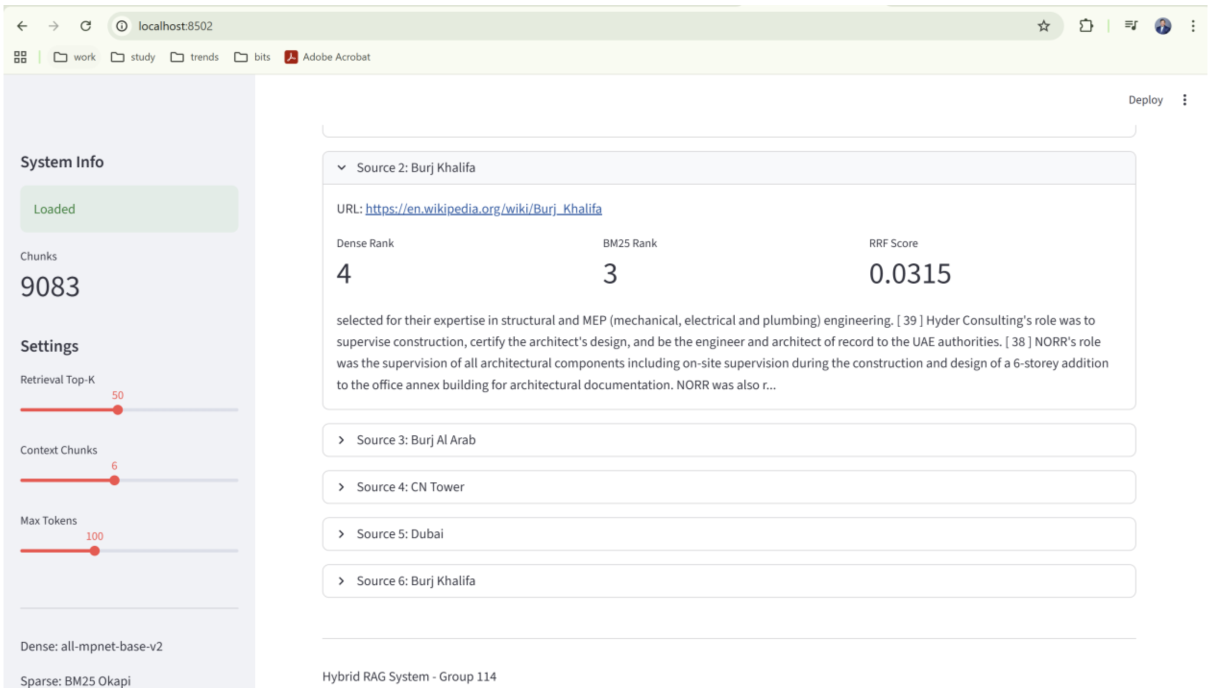
Source 1: Burj Khalifa

URL: https://en.wikipedia.org/wiki/Burj_Khalifa

Dense Rank	BM25 Rank	RRF Score
1	2	0.0325

The Burj Khalifa [a] (known as the Burj Dubai before its inauguration) is a megatall skyscraper in Dubai , United Arab Emirates. Designed by Skidmore, Owings & Merrill , it is the world's tallest structure , with a total height of 829.8 m (2,722 ft, or just over half a mile) and a roof height (excluding the antenna, but including a 242.6 m spire) [2] of 828 m (2,717 ft). It has also been the tallest building in the world since its topping out in 2009, surpassing Taipei 101 , which had held t...

fig 7: Question 2 source 1



System Info

Loaded

Chunks

9083

Settings

Retrieval Top-K: 50

Context Chunks: 6

Max Tokens: 100

Dense: all-mpnet-base-v2

Sparse: BM25 Okapi

Source 2: Burj Khalifa

URL: https://en.wikipedia.org/wiki/Burj_Khalifa

Dense Rank	BM25 Rank	RRF Score
4	3	0.0315

selected for their expertise in structural and MEP (mechanical, electrical and plumbing) engineering. [39] Hyder Consulting's role was to supervise construction, certify the architect's design, and be the engineer and architect of record to the UAE authorities. [38] NORR's role was the supervision of all architectural components including on-site supervision during the construction and design of a 6-storey addition to the office annex building for architectural documentation. NORR was also r...

Source 3: Burj Al Arab

Source 4: CN Tower

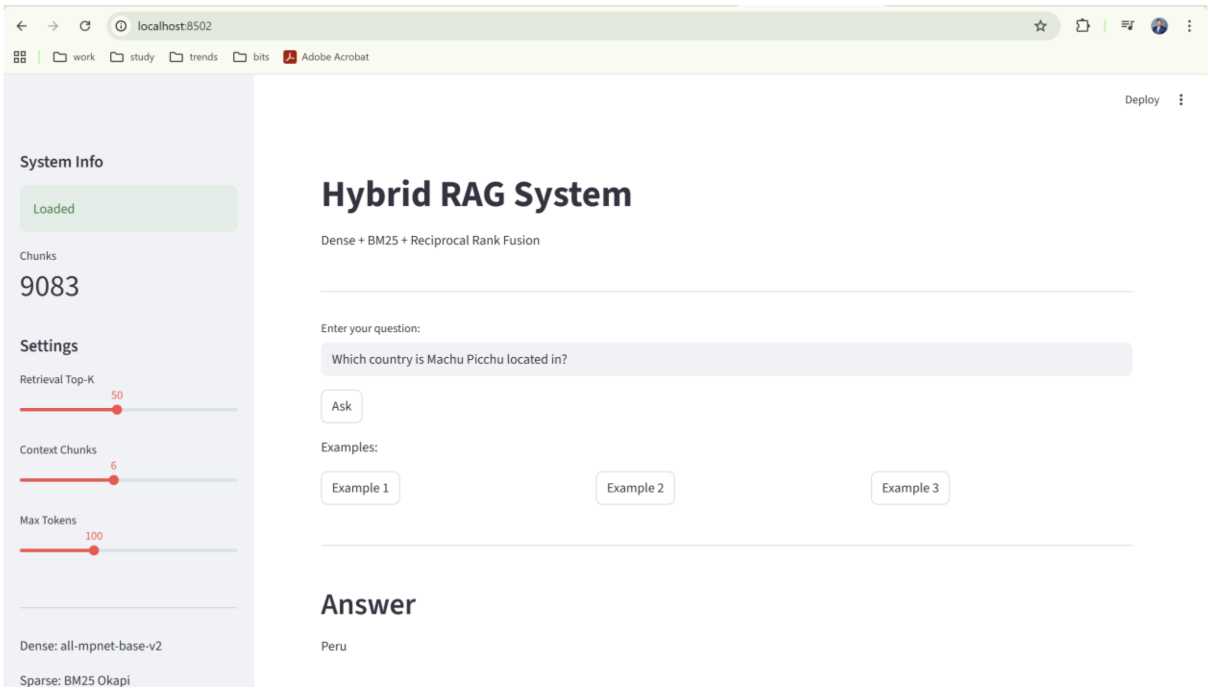
Source 5: Dubai

Source 6: Burj Khalifa

Hybrid RAG System - Group 114

fig 8: Question 2 source 2 and others

Question 3



System Info

Loaded

Chunks

9083

Settings

Retrieval Top-K: 50

Context Chunks: 6

Max Tokens: 100

Dense: all-mpnet-base-v2

Sparse: BM25 Okapi

Hybrid RAG System

Dense + BM25 + Reciprocal Rank Fusion

Enter your question:

Which country is Machu Picchu located in?

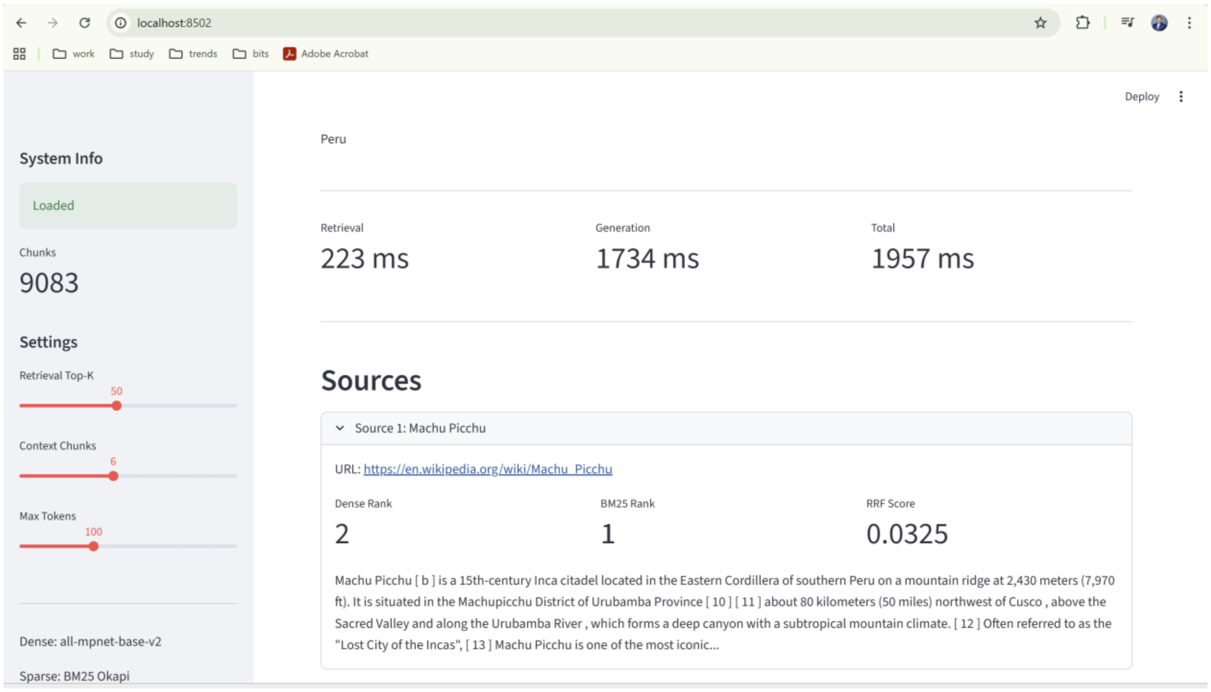
Examples:

Answer

Peru

fig 9: Question 3 demo

Answer sources for Question 3


localhost:8502

work study trends bits Adobe Acrobat

Deploy

System Info

Loaded

Chunks

9083

Settings

Retrieval Top-K: 50

Context Chunks: 6

Max Tokens: 100

Dense: all-mpnet-base-v2

Sparse: BM25 Okapi

Peru

Retrieval	Generation	Total
223 ms	1734 ms	1957 ms

Sources

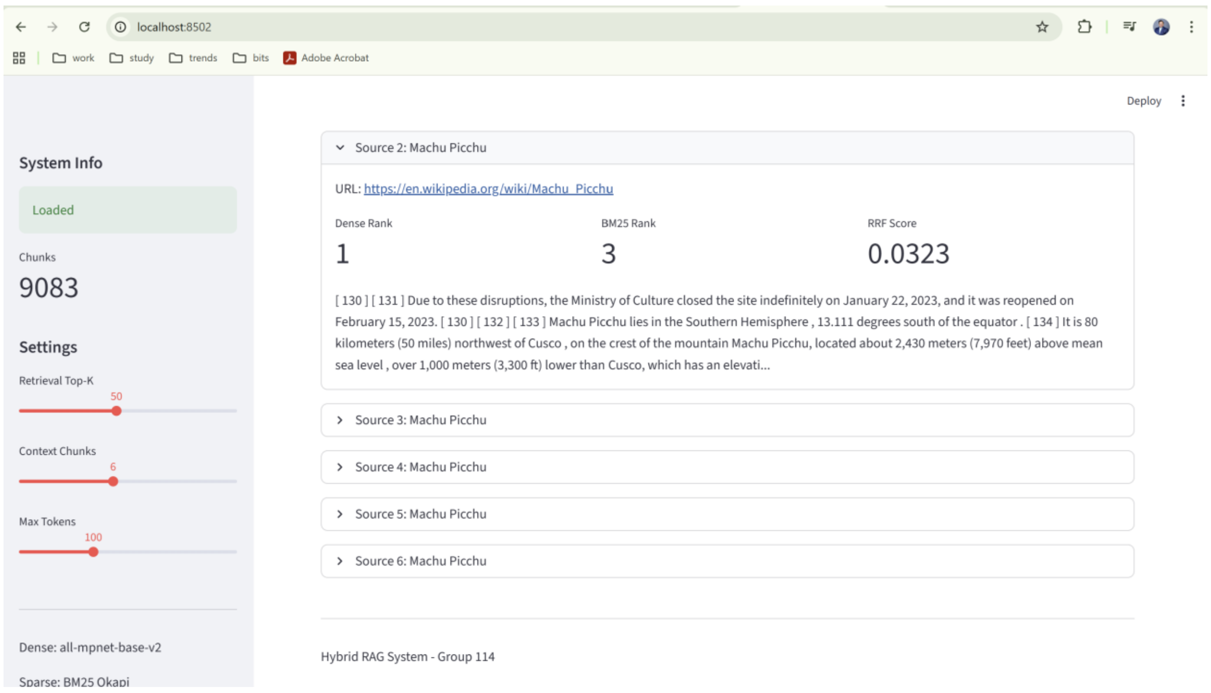
Source 1: Machu Picchu

URL: https://en.wikipedia.org/wiki/Machu_Picchu

Dense Rank	BM25 Rank	RRF Score
2	1	0.0325

Machu Picchu [b] is a 15th-century Inca citadel located in the Eastern Cordillera of southern Peru on a mountain ridge at 2,430 meters (7,970 ft). It is situated in the Machupicchu District of Urubamba Province [10] [11] about 80 kilometers (50 miles) northwest of Cusco , above the Sacred Valley and along the Urubamba River , which forms a deep canyon with a subtropical mountain climate. [12] Often referred to as the "Lost City of the Incas", [13] Machu Picchu is one of the most iconic...

fig 10: Question 3 source 1


localhost:8502

work study trends bits Adobe Acrobat

Deploy

System Info

Loaded

Chunks

9083

Settings

Retrieval Top-K: 50

Context Chunks: 6

Max Tokens: 100

Dense: all-mpnet-base-v2

Sparse: BM25 Okapi

Source 2: Machu Picchu

URL: https://en.wikipedia.org/wiki/Machu_Picchu

Dense Rank	BM25 Rank	RRF Score
1	3	0.0323

[130] [131] Due to these disruptions, the Ministry of Culture closed the site indefinitely on January 22, 2023, and it was reopened on February 15, 2023. [130] [132] [133] Machu Picchu lies in the Southern Hemisphere , 13.111 degrees south of the equator . [134] It is 80 kilometers (50 miles) northwest of Cusco , on the crest of the mountain Machu Picchu, located about 2,430 meters (7,970 feet) above mean sea level , over 1,000 meters (3,300 ft) lower than Cusco, which has an elevati...

Source 3: Machu Picchu

Source 4: Machu Picchu

Source 5: Machu Picchu

Source 6: Machu Picchu

Hybrid RAG System - Group 114

fig 11: Question 3 source 2 and others