CASCON 2025

35th IEEE International Conference on Collaborative Advances in Software and Computing

# ChatAcadien: A RAG-LLM-Based Chatbot for Exploring Acadian Genealogy

Rayen Ghali[1], Sid Ahmed Selouani[1]

[1]Laboratoire de Recherche en Interaction Humain-Système (LARIHS), Université de Moncton, Canada

UNIVERSITÉ DE MONCTON
CAMPUS DE SHIPPAGAN

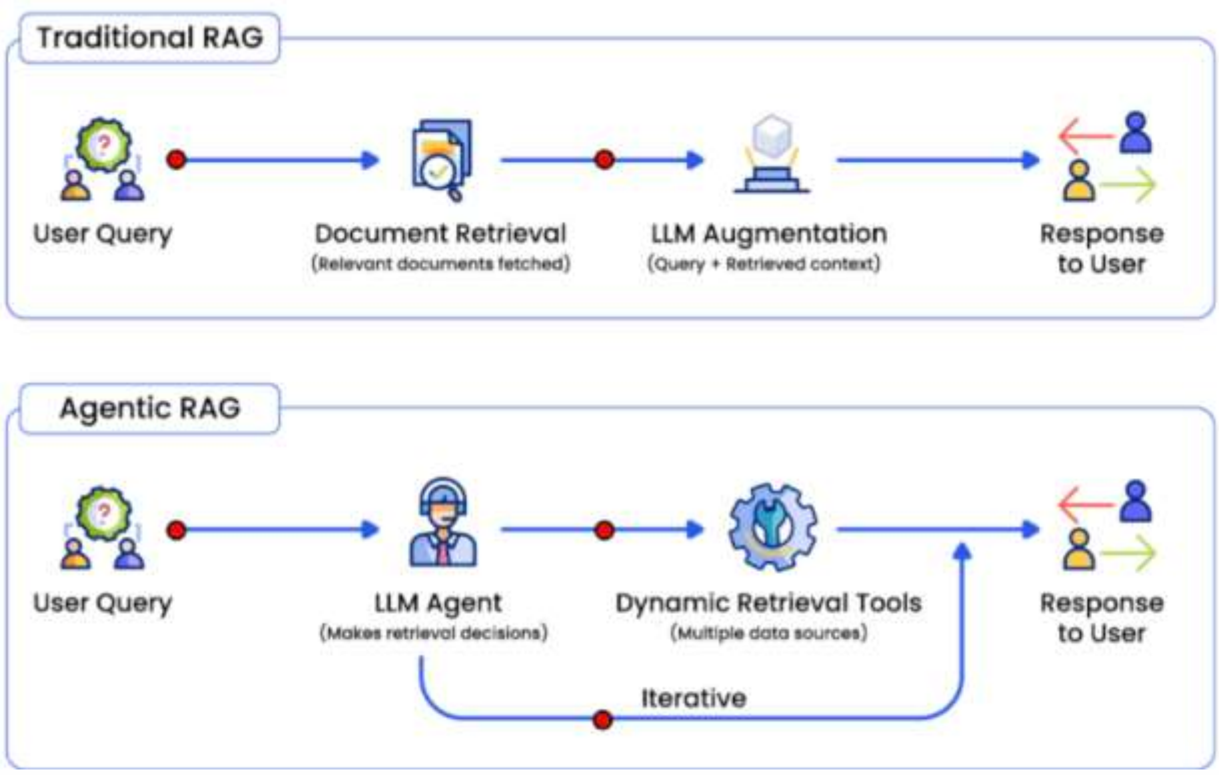
York University, Toronto, ON, Canada November 10 - 13, 2025

# CEAAC Current Limitations

- Must visit center in person (travel/physical constraints)
- 825 inquiries (April 2022-2023), 45% taking 15+ minutes each
- Manual searches by archivists (time consuming, limits capacity)
- Limited scalability for serving the public

**Centre d’études acadiennes Anselme-Chiasson - Université de Moncton**

# Agentic RAG



Anil Inamdar, via LinkedIn (August 2025)

# Genealogical Documents Challenges

- **Namesake Resolution :** Multiple individuals share identical names across generations (e.g., "Charles to Charles to Charles")
- **Temporal Reasoning :** Interpreting dates and chronological family timelines
- **Large Context Requirements :** Family trees often exceed 40k+ tokens per document
- **Domain Abbreviations :** **m** = marié/married, **n** = né/born, **s** = séparée/separated, **d** = décédé/died, **v** = vers/circa
- **Parentage Ambiguity:** "Anne SURETTE (Joseph & Isabelle Babineau)" could mean Anne's parents OR Anne's children—LLMs may misinterpret the generational direction.

**MELANSON**

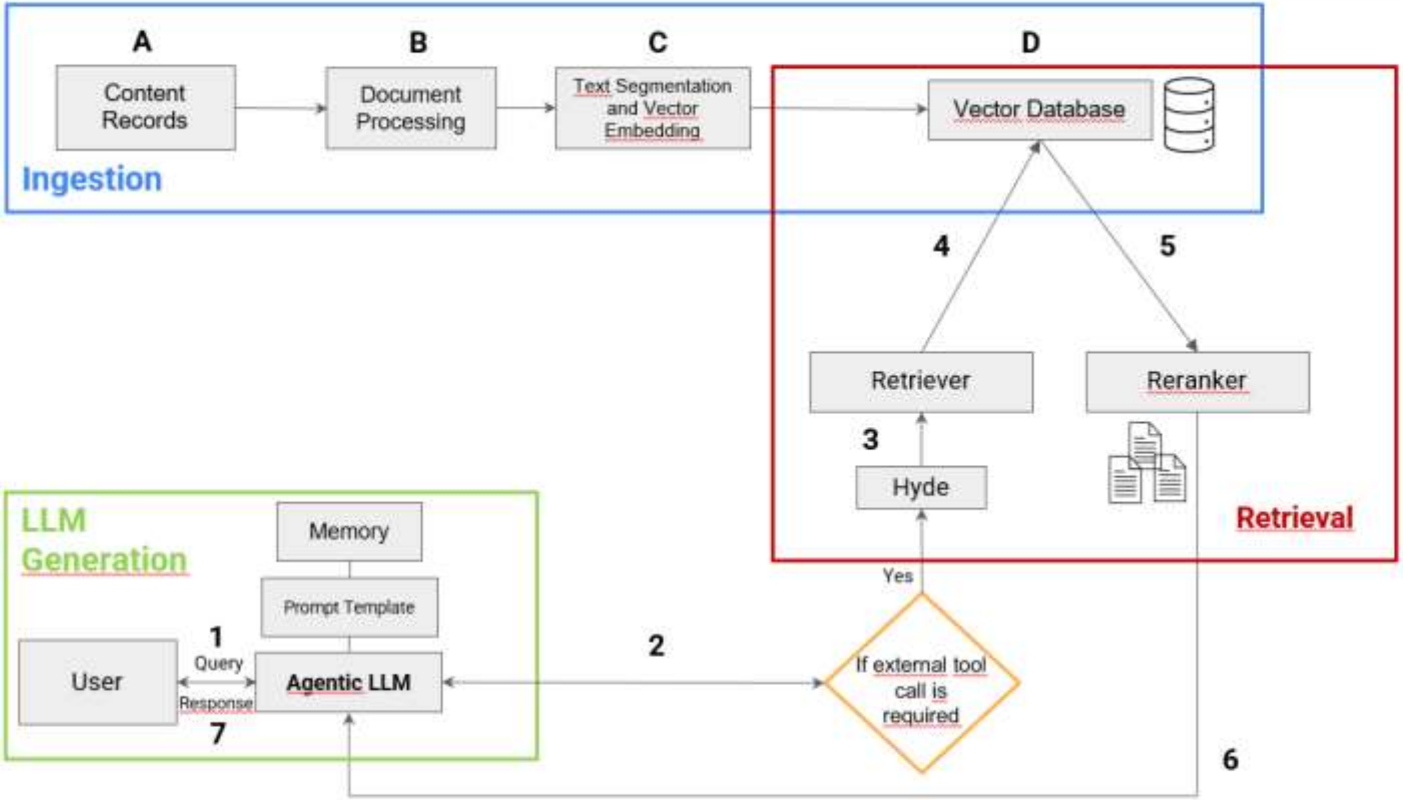
***1. CHARLES MELANSON (to Charles to Charles to Charles), m ANNE LÉGER. Children:***
*i. Anne dite Nanon, b Pisiquit – July 1766; m v 1787 Pierre dit Pître BRUN (Pierre & Théodore Boudreaux); m Menoudie 7 Nov 1852.*
*2. ii. Casimir, b Pisiquit 13 Sept 1768; m v 1790 Anne SURETTE (Joseph & Isabelle Babineau); d (according to P. Gaudet) v 1791, drowned.*
*3. iii. Joseph dit Tabaîme, b Baie-Ste-Marie 14 Nov 1771; m v 1796 Apolline FOREST (Paul & Anne Bourque).*
*iv. Jean-Baptiste, b Baie-Ste-Marie 25 Feb 1774; d Memramcook May 15, 1814, single.*
*…*
***2. CASIMIR MELANSON (to Charles to Charles to Charles to Charles), m. ANNE SURETTE. Children:***
*i. Marie, b. 1791; m. Grande-Digue September 10, 1810 Paul LÉGER (Charles & Marie Bourque); d. Cap-Pelé December 27, 1888.*
***3. JOSEPH MELANSON (to Charles to Charles to Charles to Charles), m. APOLLINE FOREST. Children:***

***Source: Stephen Adrian White's registers (37 families, 1700-1900)***

# Paper Contributions

- Introduced ChatAcadien.ca platform democratizing access to Acadian genealogical archives remotely
- Applied document preprocessing to clarify family relationships and reduce hallucinations
- Compared three chunking strategies to address context window limitations while maintaining lineage integrity
- Evaluated system performance using RAGAS metrics to ensure factual consistency

# ChatAcadien Architecture Overview



# Ingestion Stage : Document processing

- **Converted from pdf to markdown**
- **Expanded abbreviations and spelled out the relationships previously implied with parentheses.**

CHARLES MELANSON **(to** Charles**, to** Charles**, to** Charles**), m.** ANNE LÉGER. Children :

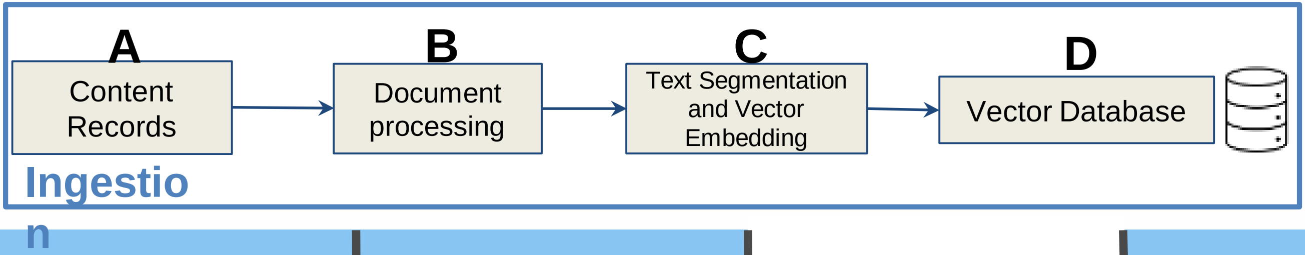
i. Anne, called *Nanon*, **b.** Pisiguit – July 1766; **m. c.** 1787 Pierre, called *Pitre BRUN* **(Pierre & Théodore Boudreau); d.** Menoudie, 7 Nov 1852.

ii…

CHARLES MELANSON (**son of** Charles Melanson**, grandson of** Charles Melanson**, great-grandson of** Charles Melanson), **married** ANNE LÉGER. Children:

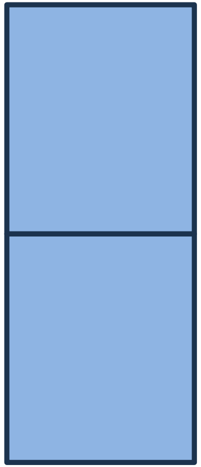
i. Anne, called Nanon MELANSON, **born at** Pisiguit in July 1766; Anne called Nanon MELANSON **married about** 1787 to Pierre, called Pitre **BRUN (Pierre BRUN, son of Pierre BRUN & Théodore Boudreau);** Anne called Nanon MELANSON **died at** Menoudie on 7 November 1852.

ii…



# Ingestion Stage : Chunking Methods

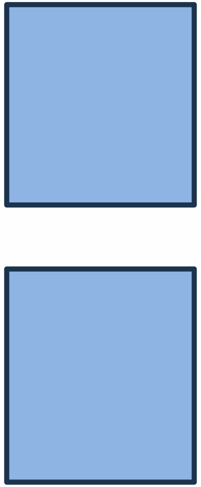
## Full Lineage Retention (FLR)



Full documents
**5k - 47k**
tokens/doc

- + Preserves complete family lineages intact
- + Provides holistic view of connections
- – Inefficient for targeted specific searches
- – Challenges distinguishing homonyms across generations (high hallucination risk)
- – Embeddings less representative of documents

## Per-Family Segmentation (PFS)

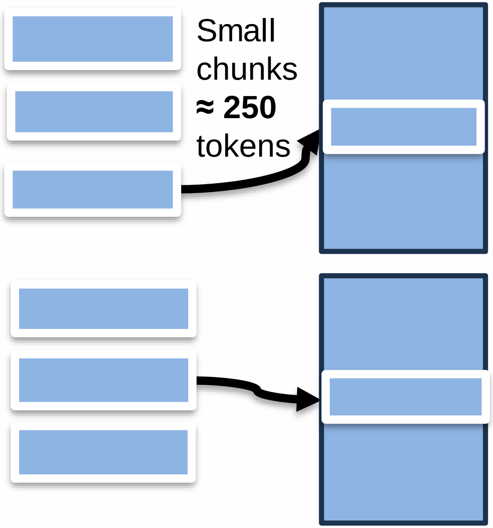


Family
block
**≈ 2k**
tokens

Family
block
**≈ 2k**
tokens

- + Enables targeted and efficient searches
- + Improves relevant information assimilation
- – Loses original ordering of families and intergenerational links

## Parent-Child Chunking (PCC)



Small
chunks
**≈ 250**
tokens

Surrounding
family
blocks :
**≈ 4k** tokens

Surrounding
family
blocks :
**≈ 4k** tokens

- + Offers balance between precision and context
- + Retains family order and lineage structure while allowing for targeted searches
- – Processes more tokens than PFS

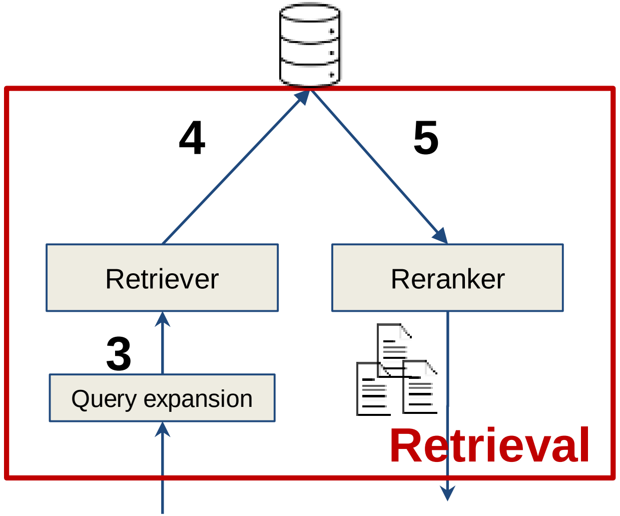
# Retrieval Stage

- **Query Expansion :** Enriches user queries with additional context and reformulations to improve retrieval coverage (e.g., " Anne Surette’s husband" → expanded to include variations like " Anne Surette mariée à", "époux de Anne Surette", relevant dates)

- **Retriever :** Performs vector cosine-similarity search across documents chunks to identify top-k candidates based on semantic proximity between query emeddings and chunk embeddings

- **Reranker :** Cross-encoder model re-scores retrieved chunks by computing contextual relevance for each query-document pair.
  Reranker models tested :

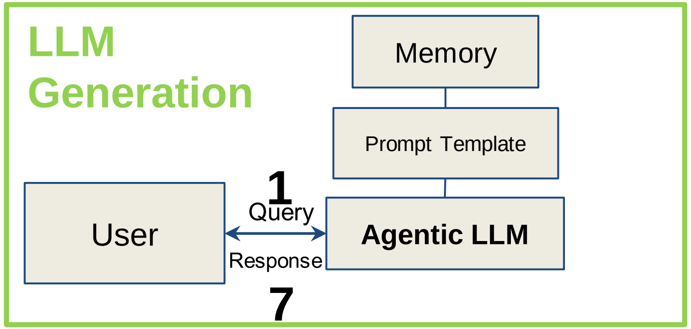
| Model | VoyageAI rerank-2 | Cohere rerank-multilingual-v3.0 |
|---|---|---|
| **Max Context Length** | Up to 16,000 tokens | Up to 4,000 tokens |



# Generation Stage :

## ChatAcadien generation models tested

| Agent LLMs | Context Window | Supports Tool-Calling | Cost | MMLU score |
|---|---|---|---|---|
| Gemini Flash 2.0 | 1M | ✓ | $0.10/M - 0.40/M | 76.4% (MMLU-Pro) |
| GPT 4o | 128K | ✓ | $2.50/M - $10/M | 88.7% |
| Claude 3.5 Sonnet | 200K | ✓ | $3/M - $15/M | 88.3% |



LLM Generation

Memory

Prompt Template

User

1

Query

Agentic LLM

Response

7

# Evaluation Framework

- **Dataset**: 15 curated Q&A pairs from real user interactions with CEAAC experts
  - Original pool: 2,591 user interactions → 367 genealogical questions → 15 final questions
  - Filtered for: answerable from provided documents + challenging entity resolution

- **Framework**: **RAGAS** with Gemini 2.0 Flash "**LLM as a Judge**“, scores averaged across 3 runs.

- **6 Metrics :**

- **Noise Sensitivity** (U, RA, GA, RC) ↓ : Proportion of claims in generated answer that lack supporting evidence in context
- **Faithfulness** (GA, RC) ↑ **:** Proportion of claims in generated answer that can be verified by retrieved context
- **Context Precision** (U, RA, RC) ↑ **:** Evaluates whether relevant chunks rank higher than irrelevant ones in retrieval
- **Context Recall** (RA, RC) ↑ **:** Proportion of reference answer information that appears in retrieved context
- **Context Entity Recall** (RA, RC) ↑ **:** Proportion of entities from reference answer found in retrieved context
- **Response Relevancy** (U, GA) ↑ **:** Measures how directly the generated answer addresses the original query

# Example

**👤 User Query:**

"**If Israël Cormier and Marguerite Robichaud had a daughter named Geneviève born around 1833** who would have **married Octave LeBlanc in 1866**..."

*Confirmation bias*: "If X had Y..." format increases false positive rate

**✗ Baseline System (GPT-4o – Full Lineage Retention ):**

"Israël Cormier and Marguerite Robichaud **indeed had** a daughter named Geneviève Cormier born around 1833. Geneviève married Octave LeBlanc in 1866...“

**✓ ChatAcadien (Gemini 2.0 – Parent-Child Chunking ):**

"**I do not find direct mention** of a Geneviève Cormier born around 1833, daughter of Israël Cormier and Marguerite Robichaud ... However, **Israël CORMIER & Marguerite ROBICHAUD are documented**, and one of their children **Perpétue CORMIER** is explicitly identified ... Their entry shows their **1818 marriage**."

*Accurately reports no such record exists*

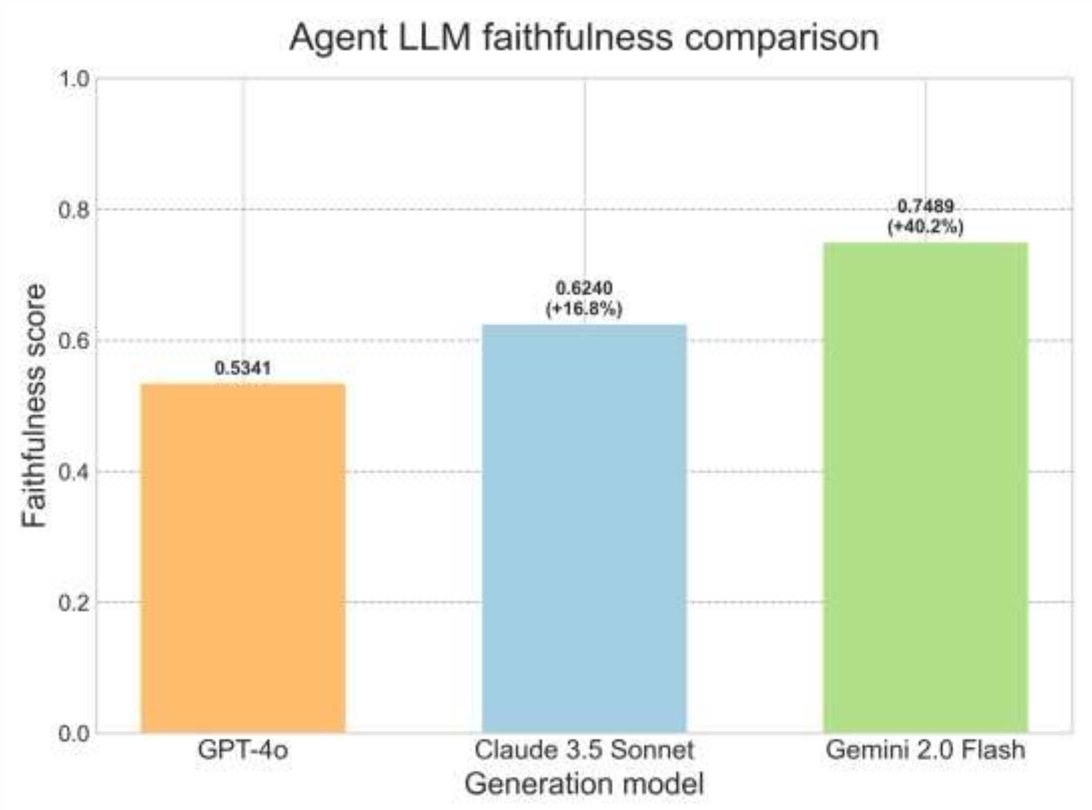
**Retrieved Documents :**

- **Snippet 1** (37fam-cormier.md, line 38): *viii. **Israël CORMIER**, … married … 1818 to **Marguerite ROBICHAUD** (**Marguerite ROBICHAUD** daughter of Dominique ROBICHAUD & **Geneviève**)*
- **Snippet 2** (37fam-leger.md, line 150): *xi. Louis LÉGER, … married … **1864** to **Perpétue CORMIER** (**Perpétue CORMIER** daughter of **Israël CORMIER & Marguerite Robichaud**)*
- **Snippet 3** (37fam-cormier.md, line 69): *vi. Basile CORMIER, … married … le 15 février 1830 à **Marguerite ROBICHAUD** (**Marguerite ROBICHAUD** daughter of Joseph ROBICHAUD & Marguerite Babineau) ; … passed away … le 26 octobre **1866**.*

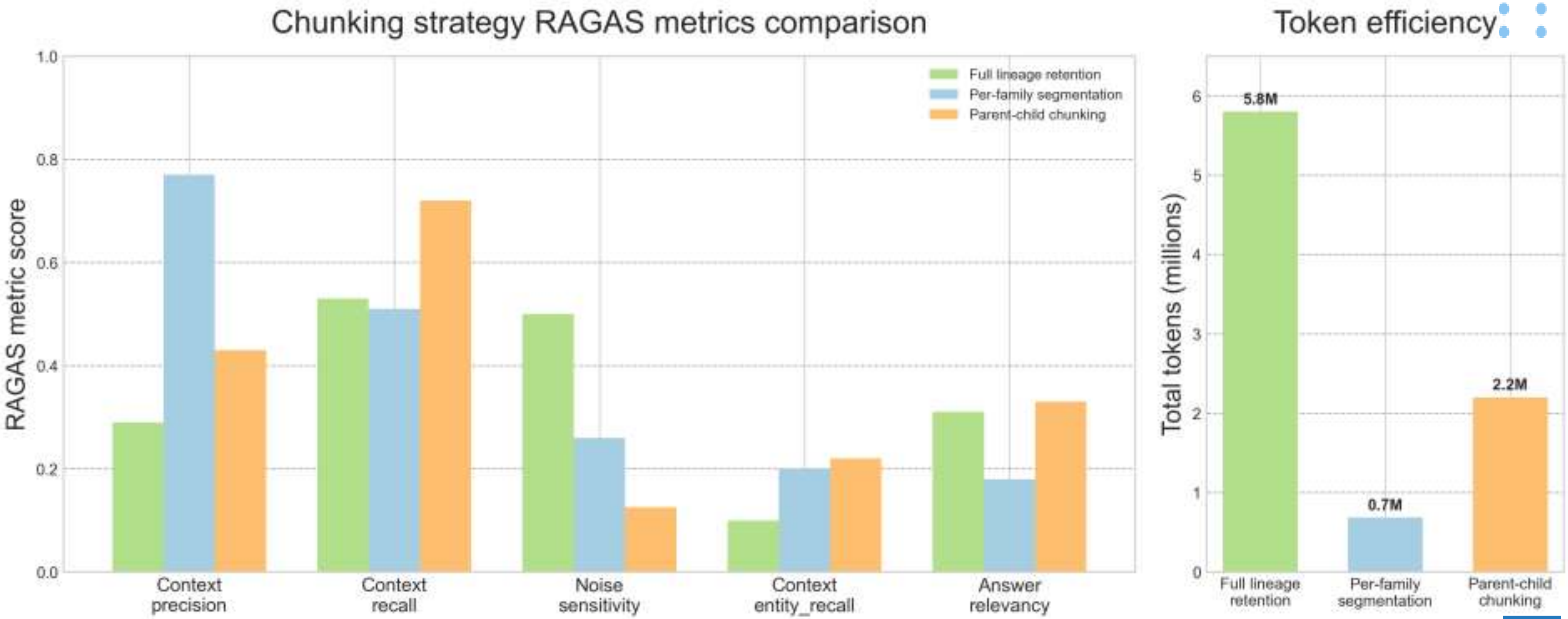
**Confusion causes :**

- **"Geneviève"** appears as Marguerite's *mother's* name, not their daughter
- No mention of years **1833** or **1866** in Israël & Marguerite's record
- **Different Marguerite Robichaud** (married to Basile, not Israël) (31 total mentions and 20 unique individuals named Marguerite Robichaud across the documents)
- Basile Cormier died in **1866** (wrong individual)

# ChatAcadien generation model selection



# ChatAcadien chunking strategy selection



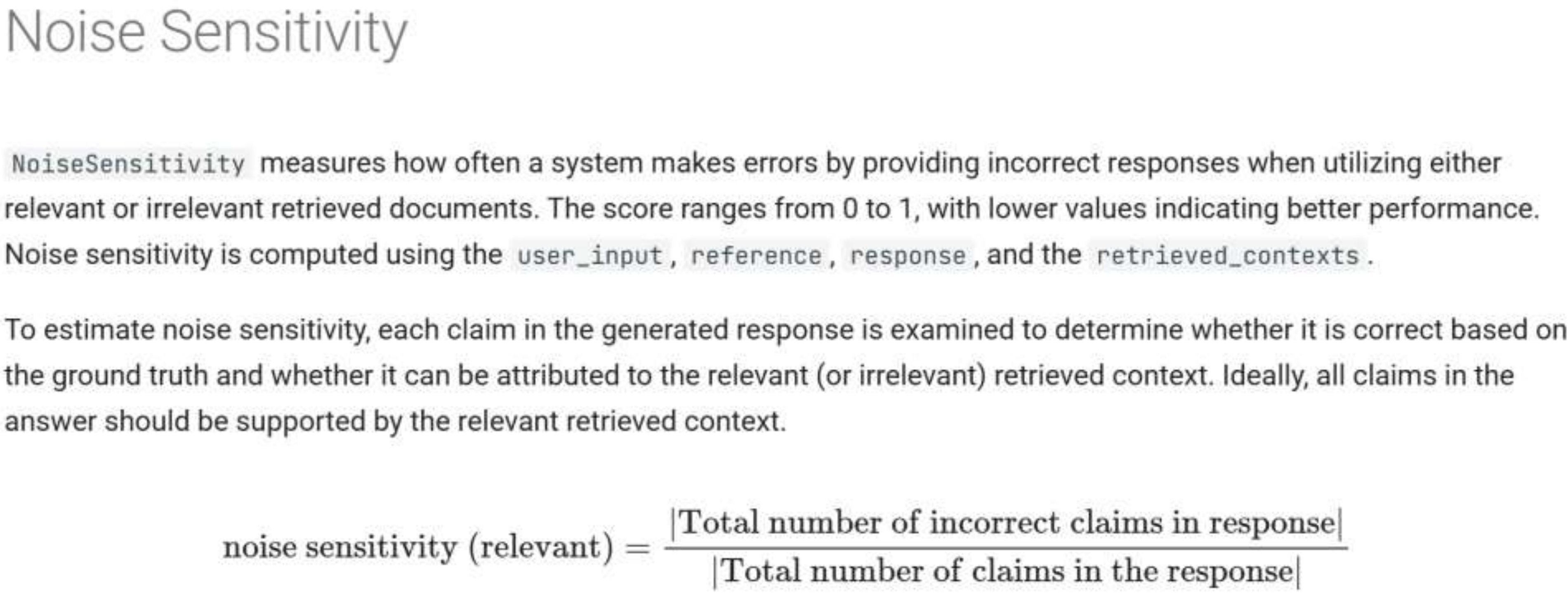
# Conclusion

Tailored segmentation significantly improved search accuracy for genealogy documents helping mitigate hallucinations

ChatAcadien successfully facilitates access to Acadian genealogical heritage and reduces archivist's workload

# Thanks for your attention

ChatAcadien

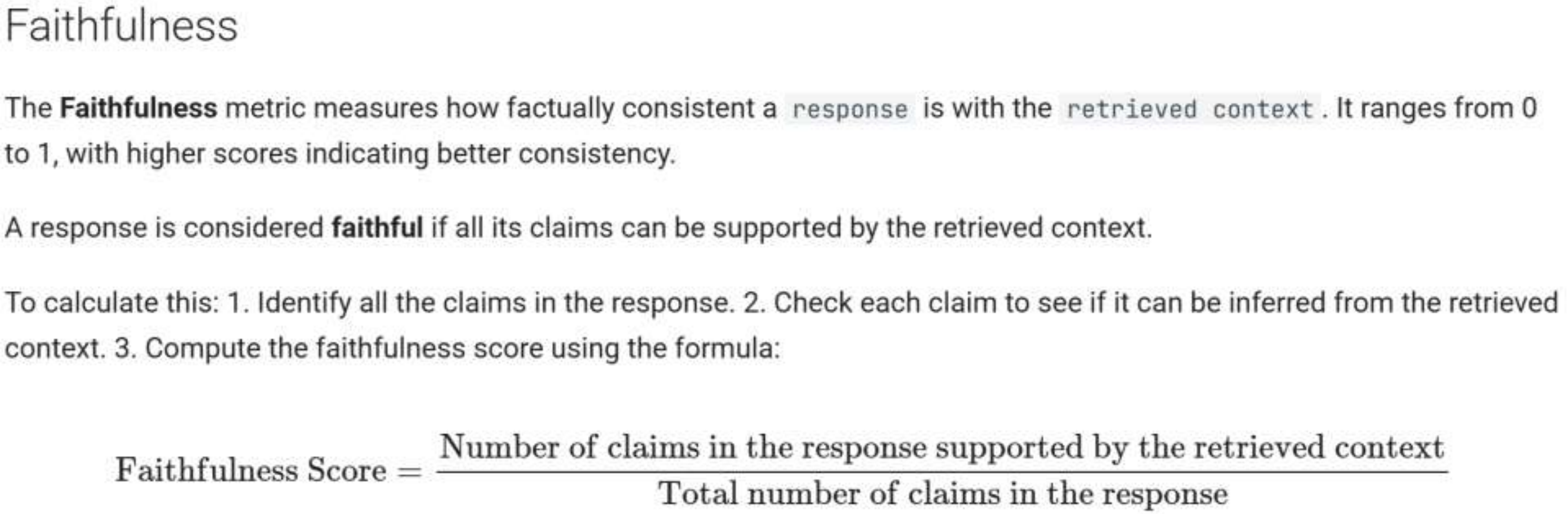


# Noise Sensitivity

`NoiseSensitivity` measures how often a system makes errors by providing incorrect responses when utilizing either relevant or irrelevant retrieved documents. The score ranges from 0 to 1, with lower values indicating better performance. Noise sensitivity is computed using the `user_input`, `reference`, `response`, and the `retrieved_contexts`.

To estimate noise sensitivity, each claim in the generated response is examined to determine whether it is correct based on the ground truth and whether it can be attributed to the relevant (or irrelevant) retrieved context. Ideally, all claims in the answer should be supported by the relevant retrieved context.

$$\text{noise sensitivity (relevant)} = \frac{|\text{Total number of incorrect claims in response}|}{|\text{Total number of claims in the response}|}$$



# Faithfulness

The **Faithfulness** metric measures how factually consistent a `response` is with the `retrieved context`. It ranges from 0 to 1, with higher scores indicating better consistency.

A response is considered **faithful** if all its claims can be supported by the retrieved context.

To calculate this: 1. Identify all the claims in the response. 2. Check each claim to see if it can be inferred from the retrieved context. 3. Compute the faithfulness score using the formula:

$$\text{Faithfulness Score} = \frac{\text{Number of claims in the response supported by the retrieved context}}{\text{Total number of claims in the response}}$$

**Example**

**Question**: Where and when was Einstein born?

**Context**: Albert Einstein (born 14 March 1879) was a German-born theoretical physicist, widely held to be one of the greatest and most influential scientists of all time

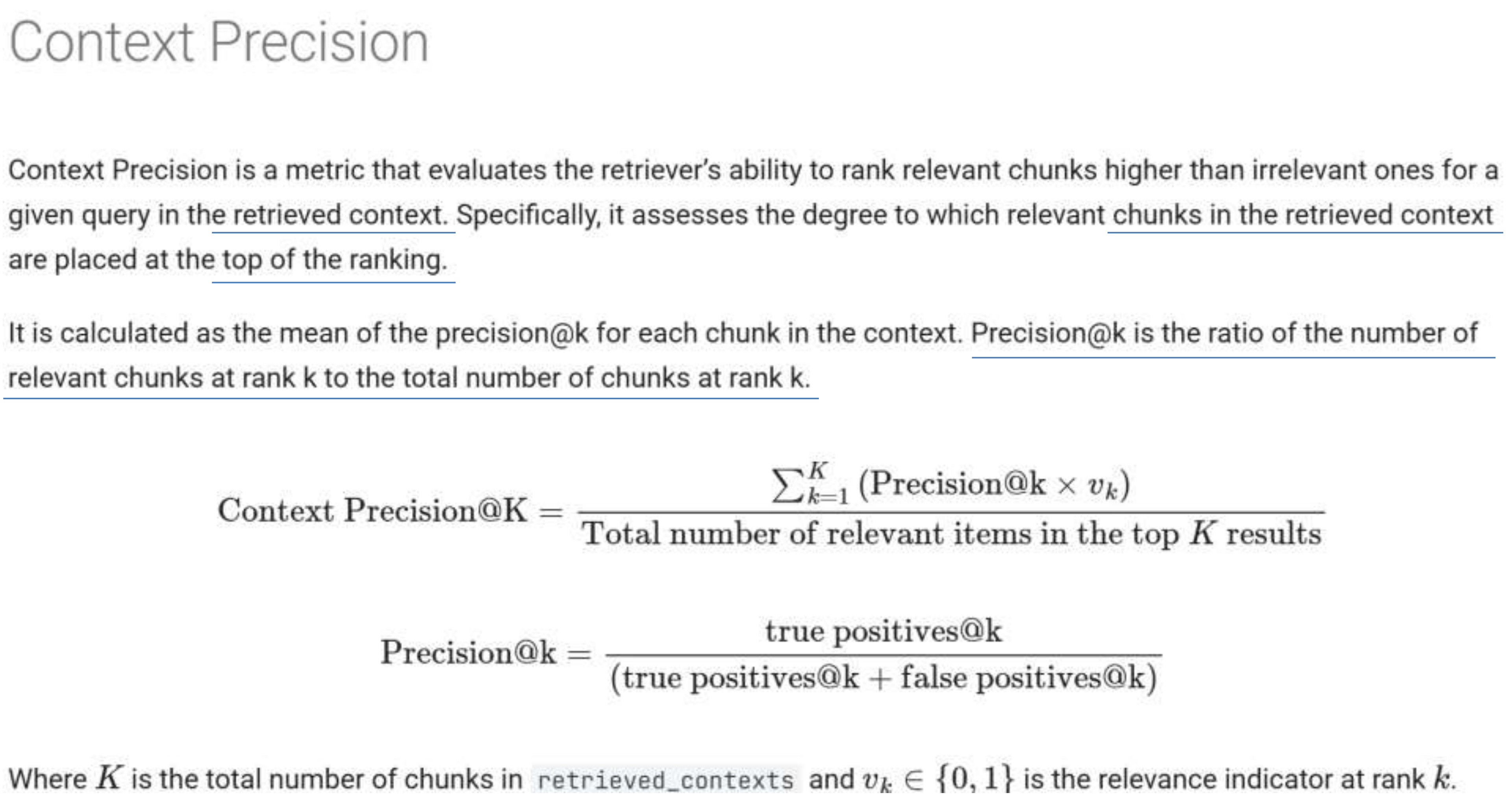
**High faithfulness answer**: Einstein was born in Germany on 14th March 1879.

**Low faithfulness answer**: Einstein was born in Germany on 20th March 1879.

Let's examine how faithfulness was calculated using the low faithfulness answer:

- **Step 1:** Break the generated answer into individual statements.
  - Statements:
    - Statement 1: "Einstein was born in Germany."
    - Statement 2: "Einstein was born on 20th March 1879."
- **Step 2:** For each of the generated statements, verify if it can be inferred from the given context.
  - Statement 1: Yes
  - Statement 2: No
- **Step 3:** Use the formula depicted above to calculate faithfulness.

$$\text{Faithfulness} = \frac{1}{2} = 0.5$$



# Context Precision

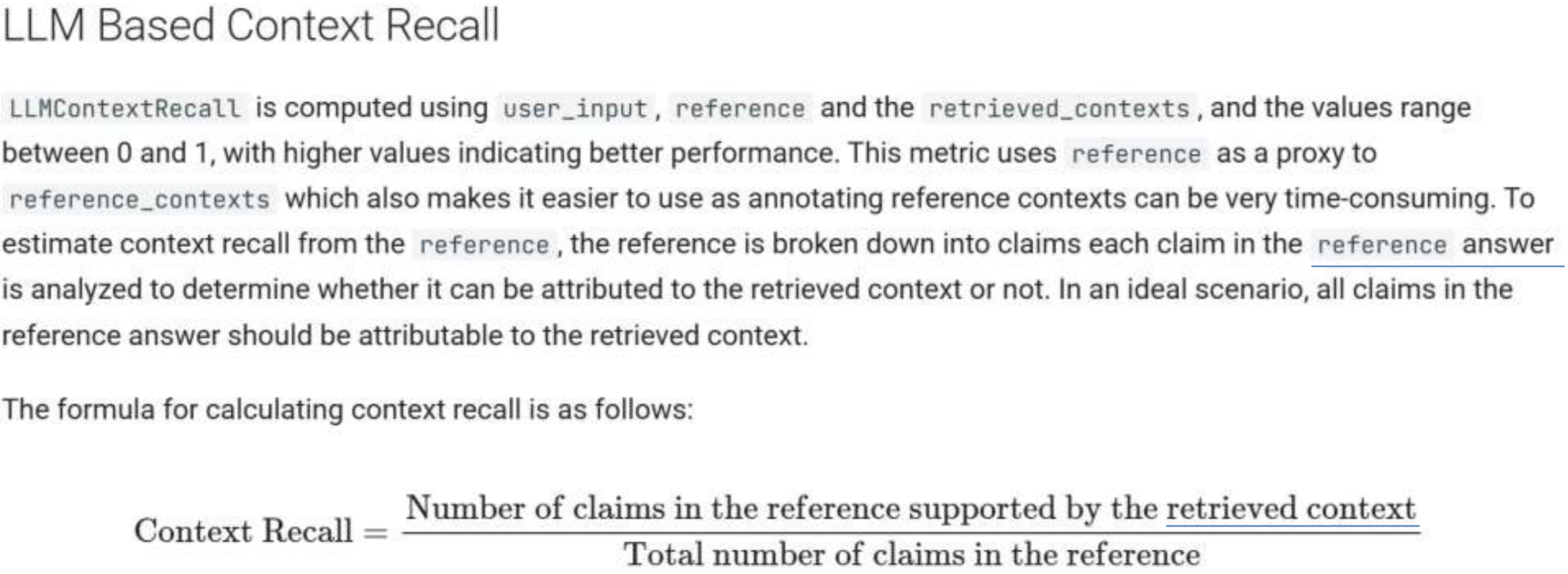
Context Precision is a metric that evaluates the retriever's ability to rank relevant chunks higher than irrelevant ones for a given query in the retrieved context. Specifically, it assesses the degree to which relevant chunks in the retrieved context are placed at the top of the ranking.

It is calculated as the mean of the precision@k for each chunk in the context. Precision@k is the ratio of the number of relevant chunks at rank k to the total number of chunks at rank k.

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

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

Where $K$ is the total number of chunks in `retrieved_contexts` and $v_k \in \{0, 1\}$ is the relevance indicator at rank $k$.



## LLM Based Context Recall

`LLMContextRecall` is computed using `user_input`, `reference` and the `retrieved_contexts`, and the values range between 0 and 1, with higher values indicating better performance. This metric uses `reference` as a proxy to `reference_contexts` which also makes it easier to use as annotating reference contexts can be very time-consuming. To estimate context recall from the `reference`, the reference is broken down into claims each claim in the `reference` answer is analyzed to determine whether it can be attributed to the retrieved context or not. In an ideal scenario, all claims in the reference answer should be attributable to the retrieved context.

The formula for calculating context recall is as follows:

$$\text{Context Recall} = \frac{\text{Number of claims in the reference supported by the retrieved context}}{\text{Total number of claims in the reference}}$$

# Context Entities Recall

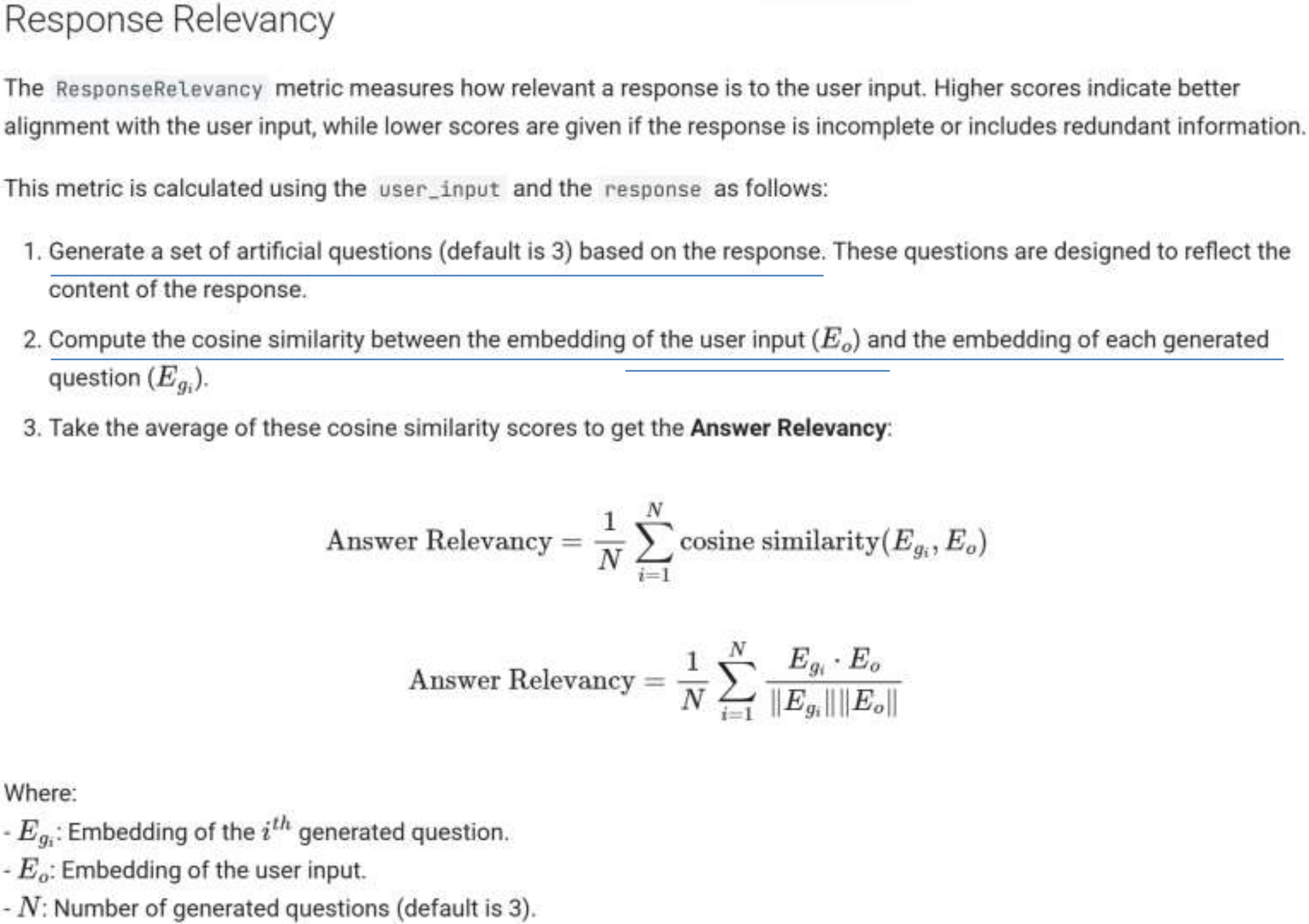
`ContextEntityRecall` metric gives the measure of recall of the retrieved context, based on the number of entities present in both `reference` and `retrieved_contexts` relative to the number of entities present in the `reference` alone. Simply put, it is a measure of what fraction of entities is recalled from `reference`. This metric is useful in fact-based use cases like tourism help desk, historical QA, etc. This metric can help evaluate the retrieval mechanism for entities, based on comparison with entities present in `reference`, because in cases where entities matter, we need the `retrieved_contexts` which cover them.

To compute this metric, we use two sets:

- $RE$: The set of entities in the reference.
- $RCE$: The set of entities in the retrieved contexts.

We calculate the number of entities common to both sets ($RCE \cap RE$) and divide it by the total number of entities in the reference ($RE$). The formula is:

$$\text{Context Entity Recall} = \frac{\text{Number of common entities between } RCE \text{ and } RE}{\text{Total number of entities in } RE}$$



## Response Relevancy

The `ResponseRelevancy` metric measures how relevant a response is to the user input. Higher scores indicate better alignment with the user input, while lower scores are given if the response is incomplete or includes redundant information.

This metric is calculated using the `user_input` and the `response` as follows:

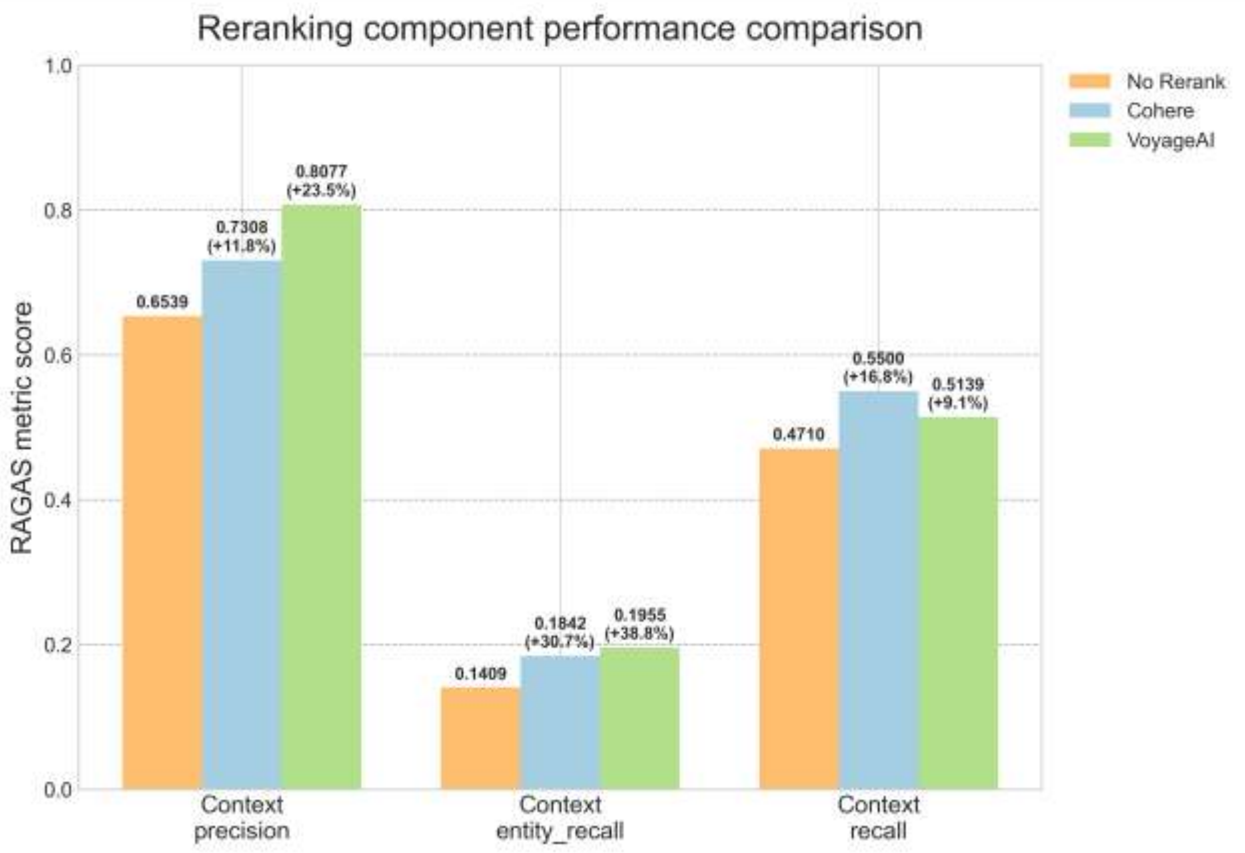
1. Generate a set of artificial questions (default is 3) based on the response. These questions are designed to reflect the content of the response.
2. Compute the cosine similarity between the embedding of the user input ($E_o$) and the embedding of each generated question ($E_{g_i}$).
3. Take the average of these cosine similarity scores to get the **Answer Relevancy**:

$$\text{Answer Relevancy} = \frac{1}{N} \sum_{i=1}^{N} \text{cosine similarity}(E_{g_i}, E_o)$$

$$\text{Answer Relevancy} = \frac{1}{N} \sum_{i=1}^{N} \frac{E_{g_i} \cdot E_o}{\|E_{g_i}\| \|E_o\|}$$

Where:
- $E_{g_i}$: Embedding of the $i^{th}$ generated question.
- $E_o$: Embedding of the user input.
- $N$: Number of generated questions (default is 3).

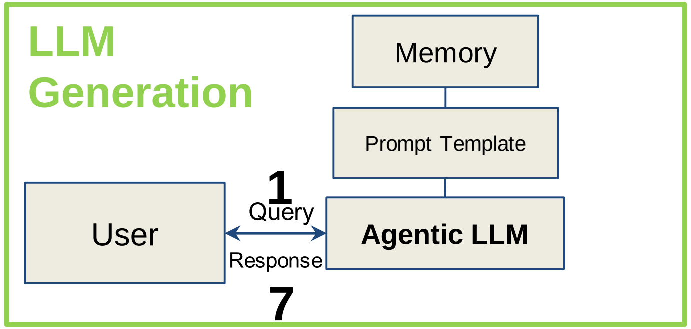
# ChatAcadien reranking component selection



# Generation Stage :

## System & Tool Prompt Design

- **Requires name/date verification** before answering genealogical queries to prevent identity confusion
- **Grounds all responses in retrieved documents** and includes source citations for verification
- **Redirects unanswerable queries** to appropriate CEAAC staff email contacts


LLM Generation
Memory
Prompt Template
1
Query
User
Agentic LLM
Response
7