June 2024

initiative for applied artificial intelligence

Retrieval-augmented Generation Realized:
Strategic & Technical Insights for Industrial Applications

Contents

About This White Paper

Background & Purpose

This white paper is the culmination of a series of studies and appliedAI RAG roundtable discussions conducted with both internal teams and appliedAI industry partners. It delves into the latest developments and challenges surrounding **Retrieval-Augmented Generation (RAG)** in industry, highlighting its emergence as a **pivotal cost-effective technique** in enhancing the **trustworthiness** and **controllability** of Large Language Model (LLM) applications over the past year.

It aims to **analyze industry demands, current methodologies,** and **hurdles** concerning the development and evaluation of RAG, facilitating **strategy development and knowledge exchange** regarding practical use cases across diverse industrial sectors.

Design Principles

Simplify Complexity: We adopt the philosophy of "less is more". Wherever possible, we emphasize conciseness and attempt to minimize lengthy textual explanations in favor of succinct, cheat-sheet style messages that convey essential information.

Foster Intuitiveness: Wherever possible, we employ visual illustrations to elucidate intricate ideas, including the overlapping and distinctive features of RAG frameworks, the RAG industrialization journey, methodological approaches for addressing challenges, and so on.

How to Use This White Paper

Section 1: RAG Industrialization - Landscape & Strategy

Managers, **strategists**, and **technical leaders** may use this section to:

- Quickly grasp fundamental RAG concepts.
- Understand the importance of RAG for the industry and the RAG technical landscape.
- Gain strategic insights on prioritizing RAG enhancement methods.
- Navigate through different stages of RAG industrialization.

Section 2: RAG Recipes for Real-World Challenges

RAG **developers**, **engineers**, and **practitioners** may use this section to:

- Explore enhancement and optimization strategies through five real-world use cases.
- Consider engineering recipes to address challenges such as precise citation based on lessons learned in these cases.

Section 3: A Deep Dive into RAG Evaluation & Metrics

RAG **developers, engineers,** and **practitioners** may use this section to:

- Understand how various RAG evaluation metrics interact with different components of RAG.
- Explore existing frameworks for RAG evaluation and their features.
- Gain an intuitive understanding of how key metrics function.

Key Takeaways

1 *RAG Industrialization - Landscape & Strategy*

For **sustainable** industrial knowledge retrieval and question-answering, **RAG** solutions are essential due to their **trustworthiness, consistency, controllability, cost efficiency**, etc. Exploring **advanced techniques** like HyDE and adaptive retrieval can enhance quality, though **resource constraints** must be considered. Recognizing challenges early in the **RAG industrialization journey** is crucial for effectively **prioritizing development tasks** and **reducing potential risks related to quality, robustness and costs** in productionizing RAG solutions.

2 *RAG Recipes for Real-World Challenges*

We present five recipes addressing challenges such as **limited initial evaluation data** for chunking and embedding method selections, **adapting to complex contexts and domain-specific conventions**, and **enhancing relevance** through metadata, SQL queries, task-specific finetuning, and multimodal RAG-augmented reasoning. Improving **retrieval quality** is essential for creating **reliable and robust RAG solutions.** This begins with **cost-effective strategies** such as metadata filtering and hybrid search, and is followed by **advanced agentic approaches** for further enhancement.

3 *A Deep Dive into RAG Evaluation & Metrics*

Assessing RAG systems is complex due to the need to evaluate the **interplay among questions, contexts, ground truth, and responses** using metrics like context relevance, recall, precision, and answer correctness. Although LLM frameworks are emerging to support RAG evaluation, no single framework covers all aspects comprehensively. The industry seeks a **standardized framework** to ensure **consistent quality, reliability,** and **scalability assessments** throughout RAG development and benchmarking.

Section 1:
RAG Industrialization - Landscape & Strategy

Let's Start with a Crash Course of RAG

Challenges of LLM-Only Approaches

In the era of Generative AI, Large Language Models (LLMs) are transforming information processing and question answering across industries. A typical LLM-only pipeline looks like this:

Three Major Challenges Encountered by LLM-Only Approach

Hallucinations

LLMs can generate inaccurate or nonsensical information.

Knowledge Cut-off

LLMs' knowledge are limited to the information available in their training dataset up to their last update.

No Access to Private Enterprise Data

LLMs cannot access or utilize private, proprietary data.

Info

What LLM Hallucination Looks Like

Ground Truth Data

Igor Fyodorovich Stravinsky (17 June 1882 – 6 April 1971) was a Russian composer and conductor with French and American citizenship.

Béla Viktor János Bartók (25 March 1881 – 26 September 1945) was a Hungarian composer, pianist and ethnomusicologist.

Question Answering with LLM

Q: In which field of music did Igor Fyodorovich Stravinsky and Béla Viktor János Bartók both have expertise?

Wrong Answer: Both Igor Fyodorovich Stravinsky and Béla Viktor János Bartók were famous **violinists.**

Right Answer: Both Igor Fyodorovich Stravinsky and Béla Viktor János Bartók were famous **composers.**

(cf. Ye *et al.* [1])

Why is Retrieval-augmented Generation (RAG) Important for Enterprises?

Trustworthiness & Reliability

RAG responses are backed by verified and up-to-date documents, improving the reliability and trustworthiness of the generated content.

Consistency & Robustness

Ensuring consistent and robust answers by always referencing the same documents, which can be reviewed and updated.

Controllability & Configurability

Allowing enterprises to control and update the knowledge base and pipeline components separately from the model.

Auditability, Explanability, and Transparency

Easier to trace the source of information provided in responses.

Process Optimization

Information retrieval, database design and contents, search modules, and other components can be optimized whenever needed, independently of the model.

IP & Data Secrecy

Fine-grained control over who can update or access the underlying data as well as the range of data that may be retrieved to generate responses.

Cost Efficiency

Minimizing the need for expensive and time-consuming model finetuning while flexibly integrating different LLMs and small language models (SLMs) into various components to optimize cost efficiency.

Scalability

Easier to scale as updates involve modifying the data sources rather than retraining the entire model.

Multi-step Reasoning and Retrieval

RAG enables integration with reasoning capabilities to tackle complex retrieval and planning tasks that demand multi-layered analysis and verification.

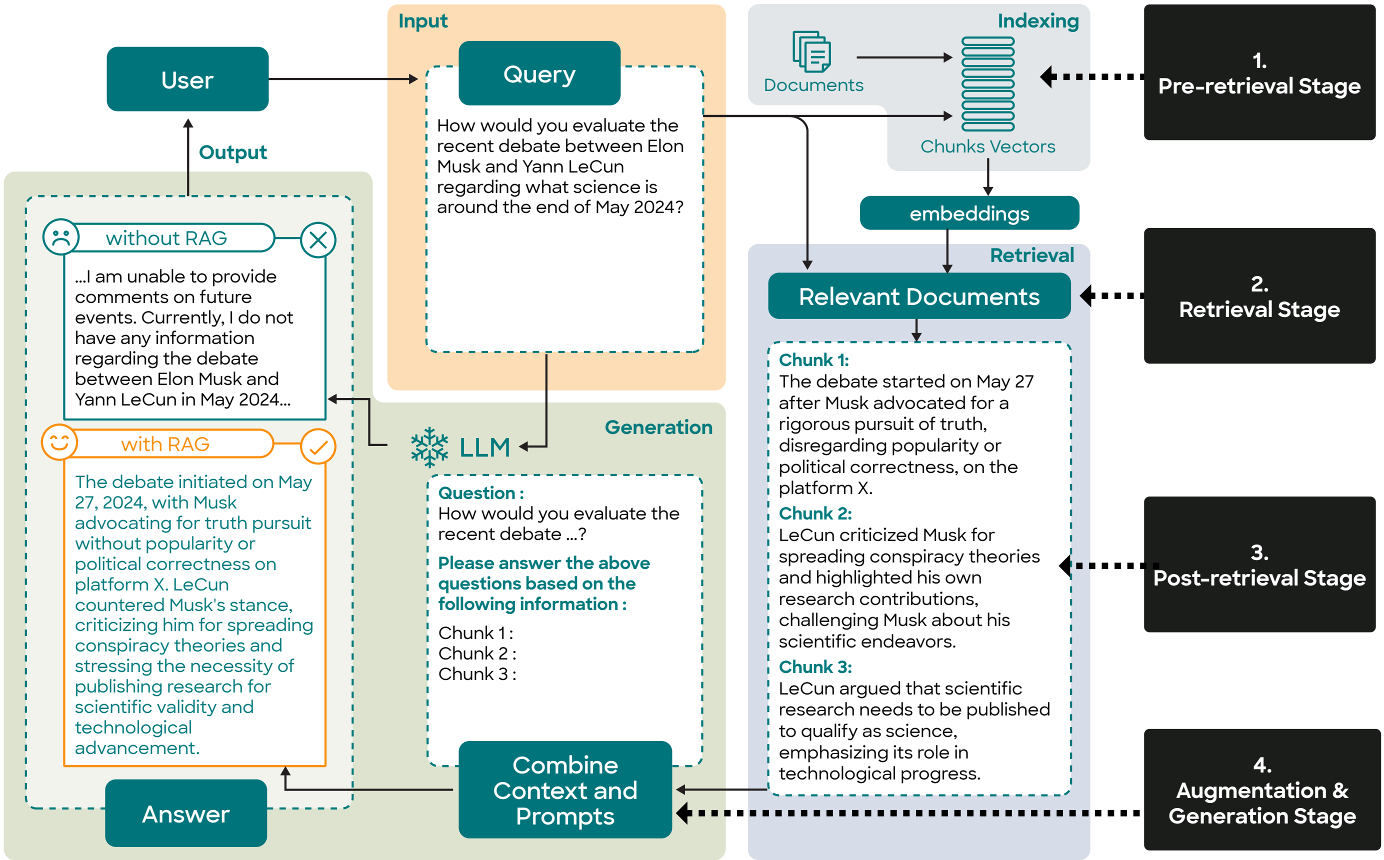
Let's Start with a Crash Course of RAG

A Brief Overview of RAG as a Solution

Retrieval-Augmented Generation (RAG) pairs the capabilities of LLMs with dynamic content from **external databases.**

Specifically, RAG involves four stages of work:

1. **Pre-retrieval Stage:** Preprocess and index the data, and store them in databases. This may also involve chunking the data and obtain their semantic embeddings as well as preprocessing incoming queries on the fly.
2. **Retrieval Stage:** Retrieve relevant documents based on semantic similarities, BM25, or other methods.
3. **Post-retrieval Stage:** Post-process the retrieved contents, such as re-ranking the documents.
4. **Augmentation & Generation Stage:** Combine the post-processed documents with prompts and generate final responses using an LLM.



(cf. Gao *et al.*[2])

Info

Interesting Question:
Does Long Context Window Solve Everything?

Background: With advanced model design and training, LLMs are increasingly efficient and context windows are expanding.

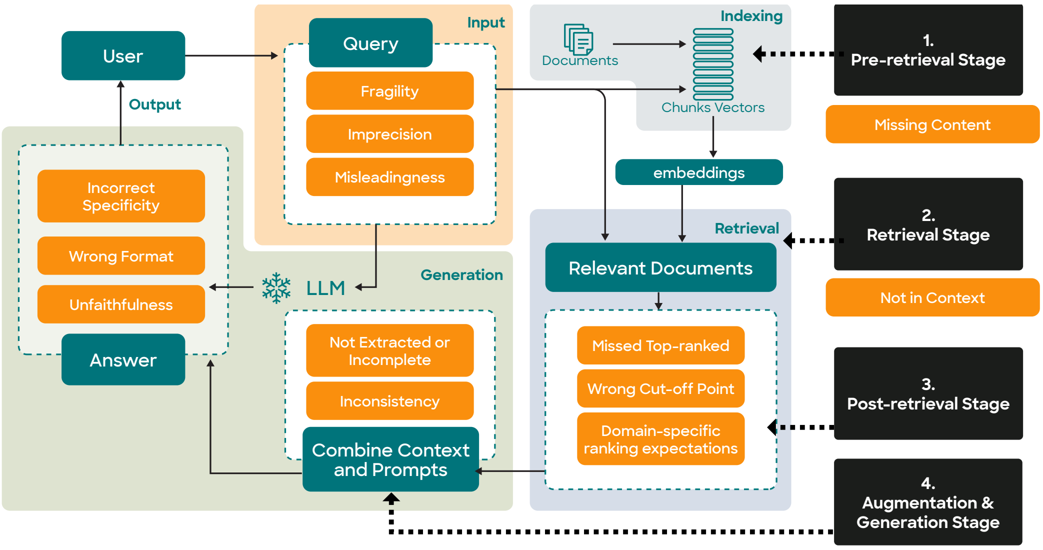
Question: Will we eventually be able to **input entire databases into prompts,** thereby eliminating concerns about factual correctness, etc.?

Consideration: While feasible, it is doubtful that this will be optimal for industry use cases, given concerns about cost efficiency, controllability, troubleshooting speed, model lock-in risks, IP and data secrecy, and other factors discussed in this white paper.

Wait! Naive RAG Doesn't Work Well!

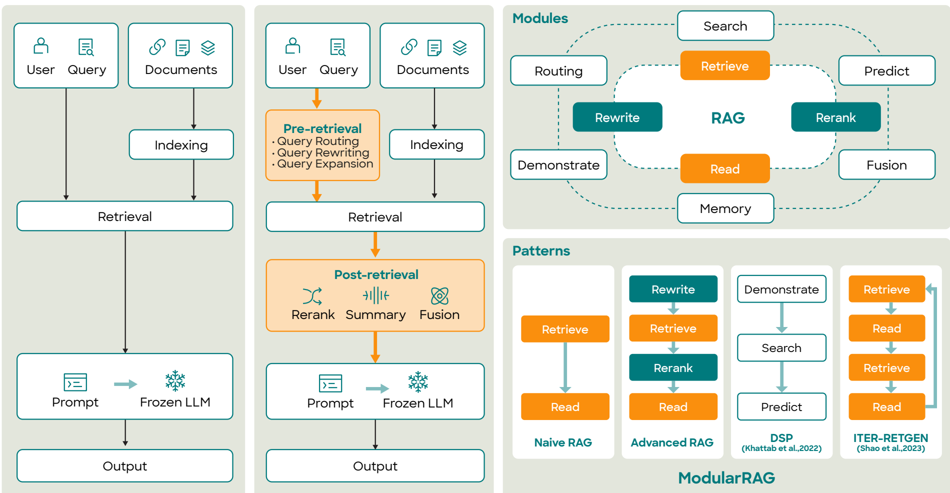
Challenges of a Naive RAG System

Naive RAG systems face inherent limitations of information retrieval and dependence on LLMs, such as those typical failure points reported in Barnett *et al.* [3], and more.



(cf. Gao et al. [2], Barnett et al.[3])

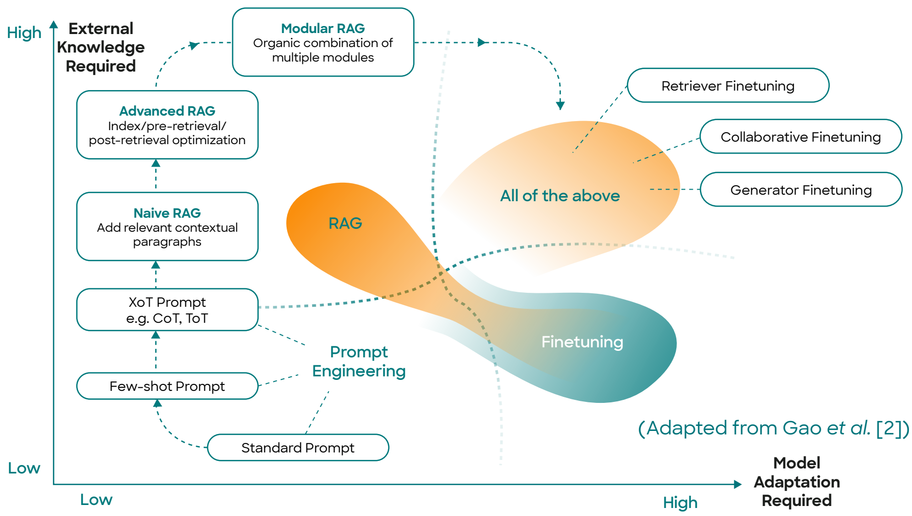
From Naive RAG to Advanced and Modular RAG



(Adapted from Gao *et al.* [2])

- In response to the challenges of naive RAG, Gao *et al.* [2] introduced the concepts of **advanced RAG** and **modular RAG** to describe the **evolution of RAG paradigms** during development.
- **Advanced RAG:** Enhances Naive RAG by improving retrieval quality with **pre- and post-retrieval strategies** and refining **indexing**, e.g., through metadata filtering.
- **Modular RAG:** Introduces **modularity** for greater **flexibility**, featuring enhanced **functional modules** and various **module combination patterns**, such as an additional **search module** for retrieval (see Gao *et al.* [2] for further details).

- **LLM optimization** is a growing area of interest, involving techniques varing in **model adaptation** and **external knowledge** needed. **Prompt engineering** uses the model's existing abilities, while comprehensive optimization often requires both **RAG** and **finetuning**.
- The decision to use RAG or finetuning should align with the specific needs of the context, such as **subject matter knowledge, domain specificity,** and the goal of **improving accuracy.**



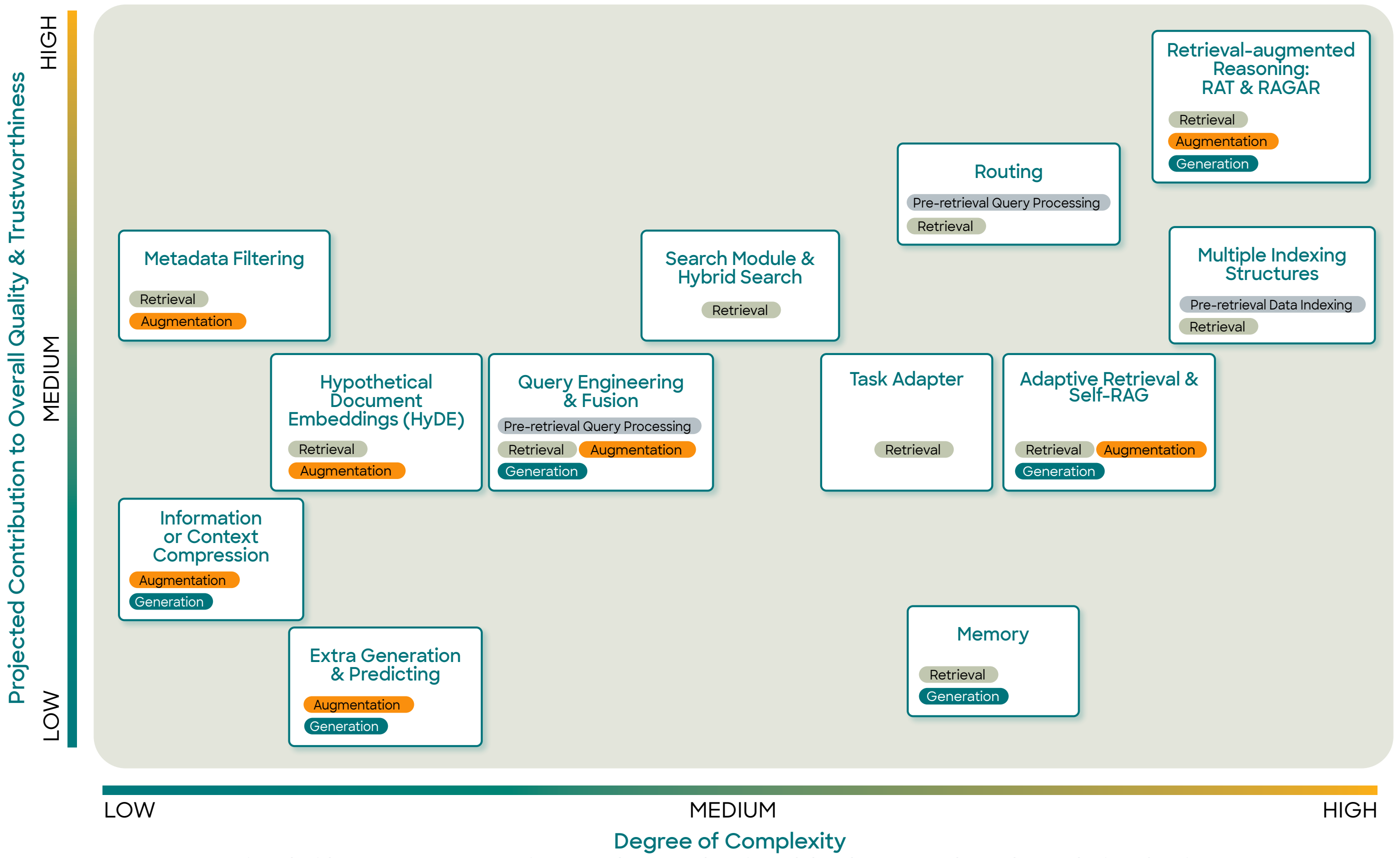
(Adapted from Gao *et al.* [2])

So, What Are Some Techniques to Enhance RAG?

Advanced RAG Techniques and Modules

We identified **12 advanced or modular RAG techniques** that can enhance different RAG components and estimated their **overall complexity** in terms of resource consumption, cost, latency, duration of development cycle, and maintenance, along with their **projected contribution to overall quality and trustworthiness.**

This serves as a **starting point to assess which techniques to prioritize** in a project, considering different levels of resource constraints, especially after identifying weak points in a RAG system.



*Associated with resource consumption, cost, latency, duration of develoment cycle, and complexity of maintenance

So, What Are Some Techniques to Enhance RAG?

Metadata Filtering

Main Improvement Areas:

Retrieval | Augmentation

What & How:

- Automatically recognizing scope of retrieval and filtering the documents using their metadata.
- Attaching metadata like dates, purpose, chapter summaries, etc., to chunks.

Query Engineering & Fusion

Main Improvement Areas:

Pre-retrieval Query Processing | Retrieval | Augmentation | Generation

What & How:

- Editing, rewriting, or spliting user queries to remove errors and biases or to retrieve specific information.
- Expanding user queries into multiple diverse perspectives and running parallel vector searches (multi-query approach).
- Ranking/merging multiple responses and aligning the final response with both explicit information and implicit user intentions.

Task Adapter

Main Improvement Areas:

Retrieval

What & How:

- Automating retrieval of prompts for zero-shot task inputs from a pre-constructed data pool, enhancing universality across tasks and models.
- Utilizing LLM as a few-shot query generator and creates task-specific retrievers based on generated data.
- Leveraging LLM's generalization capability to develop task-specific retrievers with minimal examples.

Memory

Main Improvement Areas:

Retrieval | Generation

What & How:

- Identifying LLM memories most similar to the current input.
- Utilizing a retrieval-enhanced generator for iterative memory creation and self-improvement.
- Aligning the output with the target data distribution during this reasoning process.

Information or Context Compression

Main Improvement Areas:

Augmentation | Generation

What & How:

- Condensing and compressing a vast amount of relevant information extracted from extensive knowledge bases.
- Filtering the contents and keeping only the most relevant points before passing to LLM.

Extra Generation & Predicting

Main Improvement Areas:

Augmentation | Generation

What & How:

- Utilizing LLM to generate necessary context instead of direct retrieval.
- Addressing redundancy and noise in retrieved content.

Routing

Main Improvement Areas:

Pre-retrieval Query Processing | Retrieval

What & How:

- Flexibly alternating between sources diverse in domain, language, and format based on situation.
- Determining subsequent action to user queries, including summarization, specific database searches (vector, graph, or relational databases, or a hierarchy of indices), or merging different paths.

Multiple Indexing Structures

Main Improvement Areas:

Pre-retrieval Data Indexing | Retrieval

What & How:

- Introducing graph structures to enhance retrieval by leveraging nodes and their relationships.
- Creating multi-index paths to increase efficiency.

Hypothetical Document Embeddings (HyDE)

Main Improvement Areas:

Retrieval | Augmentation

What & How:

- Creating a hypothetical document (answer) to a query and retrieve contexts similar to the hypothetical document (answer) based on its embeddings (HyDE).
- Emphasizing the similarities between embeddings of potential documents (answers) and those of real answers.

Search Module & Hybrid Search

Main Improvement Areas:

Retrieval

What & How:

- Implementing direct searches on additional data sources, such as search engines, SQL/No-SQL/graph databases, or user-specified texts or tables.
- Integrating results with those based on semantic search from vector databases.

Adaptive Retrieval & Self-RAG

Main Improvement Areas:

Retrieval | Augmentation | Generation

What & How:

- Enabling LLMs to determine when to search for necessary information, similar to how an agent uses tools.
- Evaluating relevance and level of support of retrieved contexts.
- Critiquing and assessing quality of final output.

Retrieval-augmented Reasoning: RAT & RAGAR

Main Improvement Areas:

Retrieval | Augmentation | Generation

What & How:

- Merging the concepts of RAG with Chain of Thought, enabling the system to logically reason in a certain direction and retrieve relevant contexts (RAT, Retrieval-augmented Thought [4]).
- Incorporating both Chain of RAG and Tree of RAG alongside Chain of Verification steps against the most current external web resources for multimodal fact-checking (RAGAR, RAG-augmented reasoning [5]).

And What Tools Can I Use to Develop a RAG System?

Frequently Used Tools for RAG Solutions: A Quick Glance at Common and Distinct Features

<table>
<tr><th></th><th></th><th>Langchain (v0.01.13)</th><th>LlamaIndex (v0.10)</th><th>Haystack (v1.25)</th><th>Haystack (v2.0)</th></tr>
<tr><td rowspan="9">Pre-retrieval Stage</td><td rowspan="4">Common Features</td><td colspan="4">Unstructured Data Formats: Plain Text (HTML, MARKDOWN, TXT), Image (JPEG, PNG), Document (CSV, PDF)</td></tr>
<tr><td colspan="4">Chunking Strategies: Fixed-size Chunking</td></tr>
<tr><td colspan="4">Embedding Models: OpenAI, Hugging Face, Cohere, AWS Models</td></tr>
<tr><td colspan="4">Vector Databases: Elasticsearch, FAISS, Milvus, OpenSearch, Pinecone, Qdrant, Weaviate</td></tr>
<tr><td rowspan="5">Distinct Features</td><td>Unstructured Data Formats: JSON</td><td>Unstructured Data Formats: JSON, EPUB, HWP, IPYNB, DOCX, PPT, PPTX</td><td>Unstructured Data Formats: TIFF, BMP, JSON, DOCX</td><td>Unstructured Data Formats: TIFF, BMP, DOCX, PPTX, XLSX</td></tr>
<tr><td colspan="2">Chunking Strategies: Header-based Chunking, Semantic Chunking</td><td></td><td></td></tr>
<tr><td>Chunking Strategies: Recursive Chunking</td><td>Chunking Strategies: Hierachical Chunking</td><td></td><td></td></tr>
<tr><td colspan="2">Embedding Models: AI21, Aleph Alpha, Baidu, Google, Azure, Cohere, Fastembed, Gradient, Jina, Mistral, Voyage...</td><td>Embedding Models: Anthropic, vLLM</td><td>Embedding Models: Azure, Fastembed, Gradient, Jina, Mistral, Ollama...</td></tr>
<tr><td colspan="2">Vector Databases: AstraDB, Chroma, Marqo, Neo4j, Pgvector, Azure, Baudu, Apache Cassandra, MyScale...</td><td></td><td>Vector Databases: AstraDB, Chroma, Marqo, Neo4j, Pgvector</td></tr>
<tr><td rowspan="11">Retrieval Stage</td><td rowspan="4">Common Features</td><td colspan="4">Dense Retrieval: Top-k Listing, Similarity Threshold Listing</td></tr>
<tr><td colspan="4">Sparse Retrieval: Keyword-based Listing</td></tr>
<tr><td colspan="4">Emsemble Retrieval: Hybrid Search, Multi-embedding Search</td></tr>
<tr><td colspan="4">Metadata Filtering</td></tr>
<tr><td rowspan="7">Distinct Features</td><td colspan="3">Time-weighted Search</td><td></td></tr>
<tr><td colspan="2">Self-querying/Auto-retrieval</td><td></td><td></td></tr>
<tr><td colspan="2">Multi-querying</td><td></td><td></td></tr>
<tr><td colspan="2">Contextual Compression</td><td></td><td></td></tr>
<tr><td>Parent Document</td><td></td><td></td><td></td></tr>
<tr><td></td><td>Recursive Retrieval</td><td></td><td></td></tr>
<tr><td></td><td>Auto-merging</td><td></td><td></td></tr>
<tr><td rowspan="9">Post-retrieval Stage</td><td rowspan="3">Common Features</td><td colspan="4">Cross-encoder Reranker: Hugging Face Sentence Transformer</td></tr>
<tr><td colspan="4">Cross-encoder Reranker: Cohere Rerank</td></tr>
<tr><td colspan="4">Long-context Reordering (Reordering most similar documents to context beginning/end to avoid Lost-in-the-Middle issue)</td></tr>
<tr><td rowspan="6">Distinct Features</td><td>Cross-encoder Reranker: Flashrank</td><td></td><td></td><td></td></tr>
<tr><td></td><td>Cross-encoder Reranker: ColBert Rerank</td><td></td><td></td></tr>
<tr><td></td><td>Cross-encoder Reranker: Jina Rerank</td><td></td><td>Cross-encoder Reranker: Jina Ranker</td></tr>
<tr><td></td><td>Cross-encoder Reranker: LLM Rerank</td><td></td><td></td></tr>
<tr><td></td><td></td><td>Diversity Ranker</td><td>MetaFieldRanker</td></tr>
<tr><td></td><td></td><td></td><td></td></tr>
<tr><td rowspan="11">Augmentation & Generation Stage</td><td rowspan="3">Common Features</td><td colspan="4">Generators: OpenAI, Cohere, AWS, Hugging Face Models</td></tr>
<tr><td colspan="4">Prompt Templates (query, retrieved documents, instructions, examples, tone, style, output format)</td></tr>
<tr><td colspan="4">Output Parsing (e.g., structured output, syntactically valid output, semantic validation)</td></tr>
<tr><td rowspan="8">Distinct Features</td><td colspan="3">Generator: Anthropic</td><td></td></tr>
<tr><td colspan="2">Generator: Google</td><td></td><td>Generator: Google</td></tr>
<tr><td colspan="2">Generator: Ollama</td><td></td><td>Generator: Ollama</td></tr>
<tr><td colspan="2">Generator: Azure</td><td></td><td>Generator: Azure</td></tr>
<tr><td></td><td>Generator: Mistral</td><td></td><td>Generator: Mistral</td></tr>
<tr><td>Generator: AlephAlpha</td><td></td><td></td><td></td></tr>
<tr><td colspan="2">Guardrail: Guardrails.ai</td><td></td><td></td></tr>
<tr><td>Guardrail: NVIDIA NeMo</td><td></td><td></td><td></td></tr>
</table>

Alright, How Can I Strategically Prioritize My Development Focus to Industrialize a RAG Project?

The RAG Industrialization Journey

Core Development Focus

Ideation
- Use case identification & pre-evaluation (User requirments, budget, infrastructure, critical level definition, etc.)
- Use case data exploration
- Chunking strategy pre-selection/pre-evaluation
- Embedding model pre-selection/pre-evaluation & decision on finetuning
- Vector DB pre-selection/pre-evaluation
- LLM tech stack pre-selection/pre-evaluation
- LLM pre-selection/pre-evaluation

Prototyping
- Data & metadata preparation
- Data structure design (e.g., additional keyword indices or traditional search modules)
- Vector DB establishment (data preprocessor, dataloader, text embedder, indexer)
- Initial prompt development
- Initial retrieval pipeline development
- Initial augmentation & generation pipeline development
- Initial dialogue flow and orchestration development
- Initial Chat UI
- Initial evaluation mechanism
- Continually collecting expert feedback and guarantee value

Proof of Concept (PoC)
- System architecture development
- Pre-retrieval optimization (Reconsideration of chunking/ embedding strategies, data structure, expansion of data)
- Query optimization (e.g., correction, elaboration, quotation, situational info, etc)
- Retrieval optimization (e.g., different retrieval methods, additional hybrid search or metadata filtering modules)
- Response quality optimization (e.g., iterative response generation)
- Dialogue flow and orchestration optimization
- Anti-attack & ethical/legal compliance mechanism development
- UI development
- Evaluation mechanism and metrics development (e.g., human feedback, automated evaluation etc.) -retrieval/ response/system performance.
- Personal data and log management
- Data updating pipeline development
- DevOps cycle establishment
- Continually collecting expert feedback and guarantee value

Best-known Methods (BKM)
- Long-term infrastructure development (optimizing computing & data storage resources)
- Long-term deployment Automation
- Enhancing scalability & interoperability
- Enhancing system robustness & reliability
- Enhancing data privacy protection
- Optimizing cost & system performance
- Change Management Process establishment & continual benchmarking (e.g., after model/system/data updates)
- Long-term monitoring and alerting mechanism development
- Backup, failover, & recovery mechanism development
- Maintence SOP development & operational Documentation
- Training and knowledge transfer
- Continually collecting expert feedback and guarantee value

Operation
- Deploying model updates and system patches and upgrades
- Continual retrieval/response quality evaluation and monitoring
- System performance, health, and failure monitoring
- Resource & cost monitoring, avoiding over-provisioning.
- Continual quality, performance, & cost optimization
- Conducting regular backups, disaster recovery drills, data consistency checks to ensure business continuity
- Responding to incidents, outages, and emergencies promptly and effectively to minimize impact on users and business operations.
- Continually collecting expert feedback and guarantee value

Key Challenges

Ideation
- Choosing the right technologies according to company requirements and existing infrastructure
- Determining whether finetuning will be needed and securing resources
- Cold-start problem: Little or no evaluation dataset in the beginning
- Risk identification: Discover major risks regarding resources, market demand
- Managing stakeholder expectations

Prototyping
- Sketching the overall scaffold to enhance scalable development
- Maximize long-term re-usability of data/metadata structure and contents
- Managing technical debt to enhance scalability and maintainability in the end.
- Evaluation given a limited amount of data

Proof of Concept (PoC)
- Attaining precise retrieval given a wide spectrum of queries and scenarios
- Prioritizing optimization modules and recipes
- Handling edge cases, unexpected inputs, errors.
- Handling noises and inconsistencies in the data
- Addressing security, privacy, ethical, & legal concerns
- Developing long-term sustainable evaluation metrics and automatic dataset updating mechanisms

Best-known Methods (BKM)
- Ensuring scalability & interoperability across different units/regions during fan-out
- Ensuring system robustness & reliability
- Optimizing resource utilization to maximize efficiency and minimize costs.
- Collaborating with cross-functional teams, including operations, security, compliance etc, to drive improvement

Operation
- Ensuring continuous uptime and availability of the system
- Scaling the system dynamically to handle fluctuating workloads and demand spikes
- Managing updates, patches, upgrades without disrupting operations

Key Capabilities

Ideation
- Ability for fast technique assessment
- Proficiency in data synthesis, augmentation, or bootstrapping techniques

Prototyping
- Proficiency in designing scalable, extensible, and maintainable system architectures
- Ability to identify, prioritize, and mitigate technical debt early in the development process

Proof of Concept (PoC)
- Proficiency in identifying key modules for improvement
- Expertise in compliance and security

Best-known Methods (BKM)
- Ability to envision the long-term goals and requirements of the system
- Proficiency in optimizing resource utilization

Operation
- Expertise in implementing fault-tolerant, scalable architectures
- Proficiency in implementing CI/CD pipelines

Key Metrics

Ideation
- Pairwise statistics

Prototyping
- Human feedback (e.g., binary likes/dislikes or A/B tests)
- Custom LLM-based or RAGAR approaches

Proof of Concept (PoC)
- Regular RAG evaluation metrics (context relevancy, answer relevancy, response completeness, faithfulness, context utilization, factual accuracy, context precision, context recall, answer correctness, answer similarity)
- Custom LLM-based or RAGAR approaches.
- Human feedback (e.g., binary likes/dislikes or A/B tests)

Best-known Methods (BKM)
- Quality metrics (regular RAG metrics, RAGAR etc.)
- Performance metrics (latency, cost, token counts etc)
- Resource utilization metrics (computing, storage)

Operation
- System uptime, health, and failure metrics
- Outage recovery metrics
- Resource utilization metrics (computing, storage)
- Performance metrics (latency, cost, token counts etc)
- Quality metrics (regular RAG metrics, RAGAR etc.)

Section 2:
RAG Recipes for Real-World Challenges

The Cold Start Recipe:

Data-Driven Chunking & Embedding Strategy Without Evaluation Dataset

Background & Goals of this Recipe

- Across diverse domains, a crucial objective in the **initial phase** of RAG development is to establish an **effective strategy for chunking and embedding** the data to enable efficient and relevant retrieval.
- The decisions concerning chunking and embedding at the outset have a significant impact on system design and output quality. These choices not only involve the **costs of computing** resources but are also **difficult to change** once determined.

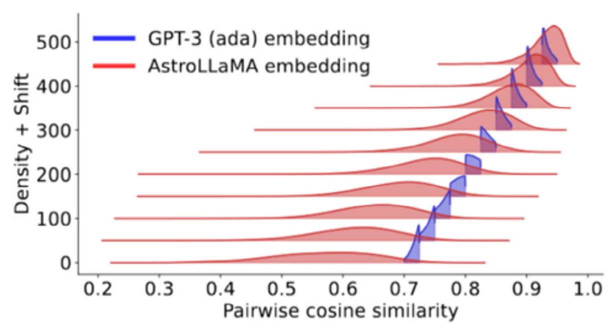
Challenges Addressed

- **Scant Availability of an Evaluation Dataset:** At project outset, a common bottleneck is the absence of a well-constructed **dataset developed by domain experts.**
- **Lack of Time and Resource:** In the early stage, developers often face time and resource constraints that limit extensive validation of their approach. However, they require **a tool to assist in making quick design decisions** due to the urgency of showcasing a working prototype to stakeholders.

RAG Cold Start Analytics: Exploiting the Potential of Pairwise Cosine Similarities

The Underlying Idea

- Nguyen *et al.* [6] assessed the quality of embeddings from a **general** LLM (GPT-3) and a finetuned LLM **specific** to astronomy (AstroLLaMA) by examining the **distribution of pairwise cosine similarity scores** (see right, divided into 10 equal bins based on similarity levels from GPT-3).
- Given this set of domain-specific documents, embeddings by GPT-3 are overly generic with similarities clustering around 0.7-0.9, suggesting **a lack of discriminative power.**
- Embeddings generated by the finetuned model exhibits **a much higher variance** within each decile, pointing to a higher proficiency at capturing the semantic diversity in this domain.
- This hints at the possibility that **quality of embeddings** may be reflected in the **distribution** of pairwise cosine similarity scores.
- On the basis of thorough sanity checks, an even distribution of embeddings may be indicative of a more **granular semantic representation**, contributing to improved document retrieval.

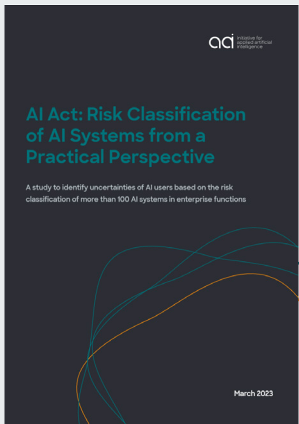


Paper 1: Astrophysical gyrokinetics: kinetic and fluid turbulent cascades in magnetized weakly collisional plasma
Paper 2: Comment on modified Coulomb law in a strongly magnetised vaccum
GPT-3 cosine similarity score: **78.5%**
AstroLLaMa cosine similarity score: 36.3%

Paper 1: A Spitzer census of the IC 348 nebula
Paper 2: Sequential and spontaneous star formation around the mid-infrared halo HII region KR 14
GPT-3 cosine similarity score: 82.4%
AstroLLaMa cosine similarity score: **92.8%**

Applied to the appliedAI AI Act White Paper

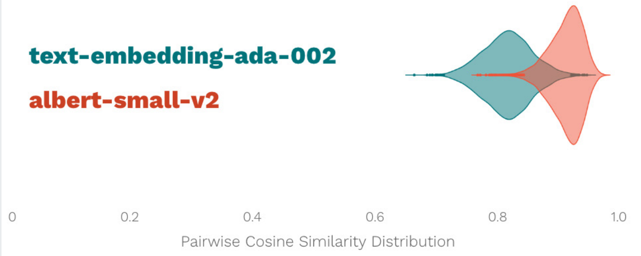
- **Data:** As a prelimiary experiment, we applied this idea of looking into pairwise cosine similarities for initial chunking/embedding assessment to the appliedAI white paper: AI Act: Risk Classification of AI Systems from a Practical Perspective (7511 words).
- **Goal:** Gain a quick understanding of the quality of embeddings from **text-embedding-ada-002** and **albert-small-v2.**
- **Experimental Configurations:** The Langchain CharacterTextSplitter was employed to segment the data, with a chunk size of 1000 and zero overlap between chunks. Additionally, the Langchain QAGenerationChain was utilized to automatically generate question-answer pairs from these chunks for a basic RAGAS evaluation using a naive RAG scenario (Top-k=1, GPT-3.5 Turbo as the response language model).

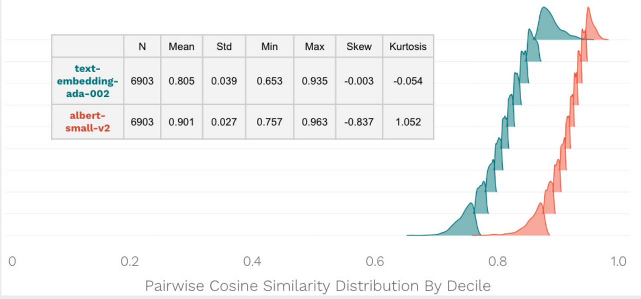


The Cold Start Recipe:

Data-Driven Chunking & Embedding Strategy Without Evaluation Dataset

- **Results:**
  - The range of pairwise cosine similarity scores from either text-embedding-ada-002 or albert-small-v2 clustered above 0.6, which is not ideal for threshold-based retrieval. This suggests that neither model may be optimal for this particular dataset.
  - However, an initial exploratory data analysis revealed that albert-small-v2 yielded a distribution that was slightly more peaked, indicating potentially lower quality compared to text-embedding-ada-002, as also reflected in the RAGAS scores.





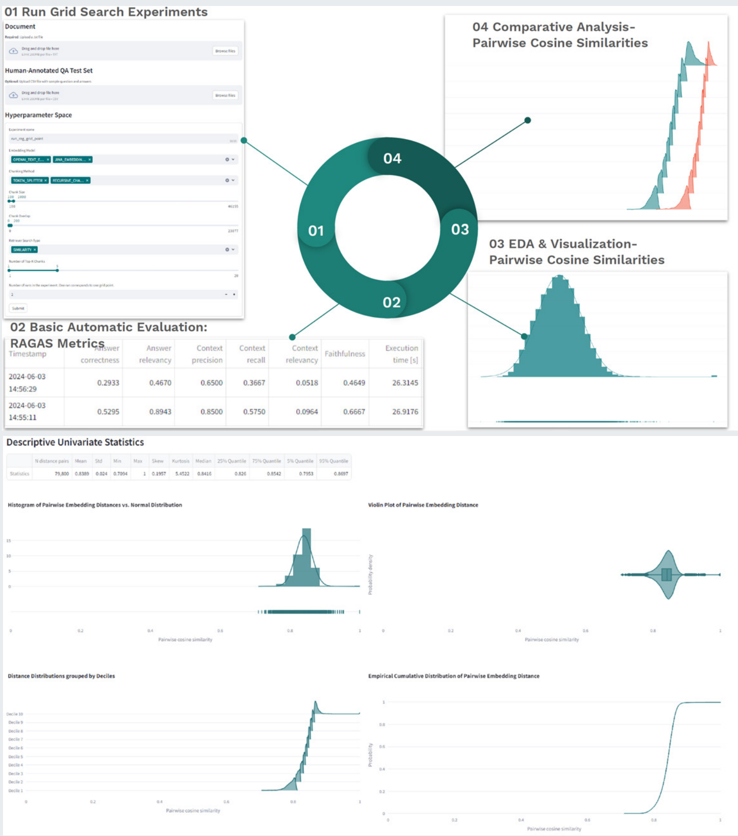
	N	Mean	Std	Min	Max	Skew	Kurtosis
text-embedding-ada-002	6903	0.805	0.039	0.653	0.935	-0.003	-0.054
albert-small-v2	6903	0.901	0.027	0.757	0.963	-0.837	1.052

Questions	Ground Truth	Response text-embedding-ada-002	Response albert-small-v2
According to the study, what percentage of the AI systems examined were in the high-risk class?	18%	According to the study, 18% of the AI systems examined were in the high-risk class.	The study does not provide information on the percentage of AI systems examined that were in the high-risk class.
What are the three risk classes outlined in the AI Regulation, and what are the main requirements for each?	The three risk classes outlined in the AI Regulation are high-risk, low-risk, and unclear. The main requirements for each are not specified in the provided context.	Prohibited AI systems, High-risk AI systems, and Low-risk AI systems.	I'm sorry, but the context provided does not mention the three risk classes outlined in the AI Regulation or their main requirements.

	Context Precision	Context Recall	Context Relevance	Faithfulness	Answer Correctness	Answer Relevance
text-embedding-ada-002	**0.85**	**0.58**	**0.10**	**0.67**	**0.53**	**0.89**
albert-small-v2	0.65	0.37	0.05	0.46	0.29	0.47

Prototyping a RAG Cold Start Analytic App

- Based on this idea, **appliedAI Initiative** developed a prototype RAG cold start analytic app that allows:
  - **Grid search** over different embedding models, chunking methods, chunk sizes, chunk overlaps, retriever search types, and number of top-K chunks.
  - **Automatic QA generation** and evaluation using RAGAS metrics.
  - **Exploratory data analysis (EDA)** on pair-wise cosine similarity score distributions the such as descriptive univariate statistics.
  - **Comparative analysis of pair-wise cosine similarity score distributions** by decile between any two grid points.
- Compared to other approaches that project embeddings into 2D spaces for visual inspection of embedding quality, this approach offers a **concrete, quantitative measure.**
- Basic **sanity checks,** e.g., ensuring that similar documents receive higher pairwise cosine similarity scores, should be conducted to ensure that the models did not yield noises.



The Virtual Havruta Recipe:
Optimizing Queries for Multifaceted Contexts & Precise Citations

Background & Goals of this Recipe

- Virtual Havruta is a **Judaism study companion** that uses **RAG** techniques to generate **research-oriented explanations** based on **reliable scriptural references**.
- The project is an ongoing open-source endeavor, resulting from the collaborative efforts of **TUM Venture Labs, Sefaria,** and the **appliedAI Initiative GmbH.** It leverages the substantial array of resources and digitalization work that Sefaria has contributed to the realm of Jewish scriptures.
- Typically, answering a specific question requires **rich and multifaceted contextual knowledge,** and users expect **highly accurate responses.**

Challenges Addressed

- **Incomplete Queries:** User queries don't always contain critical information for semantic search.
- **Implicit Assumptions:** User queries may be short or fragile but carry rich implicit background knowledge and assumptions.
- **Non-Aligned Reranking:** Similarity scores or reranking may not always align with domain-specific conventions, user habits, concepts, or expectations about the best references, e.g., when primary sources (e.g., Tanakh, Talmud) are preferred over secondary sources (e.g., liturgy, commentary).
- **Precision Demands:** Highly precise citations among similar, neighboring chunks may be required.

"Marcus Jastrow Dictionary"

"Maccabees 1 ?"

"Encyclopedia Judaica"

"Can you explain the concept of דרכי שלום"

References

1. Reference: Rabbeinu Bahya, Shemot 15:25:1. Version Title: Torah Commentary by Rabbi Bachya ben Asher, trans. Eliyahu Munk, 1998. Document Category: Commentary URL: https://www.sefaria.org/Rabbeinu_Bahya,_Shemot.15.25.1.

2. Reference: Sforno on Deuteronomy 4:8:1. Version Title: Eliyahu Munk, HaChut Hameshulash. Document Category: Commentary. URL: https://www.sefaria.org/Sforno_on_Deuteronomy.4.8.1.

3. Reference: Rabbeinu Bahya, Bamidbar 19:2:4. Version Title: Torah Commentary by Rabbi Bachya ben Asher, trans. Eliyahu Munk, 1998.. Document Category: Commentary URL: https://www.sefaria.org/Rabbeinu_Bahya,_Bamidbar.19.2.4.

→ Primary sources (e.g., Tanakh, Talmud), not commentaries, should be the top-ranking sources.

what are the cases in the Talmud pertaining to laws regarding a pwrsons ייחוס, where a testimony of an individual is believed only immediately, but as some very short time elapses, this individual's testimony is not to be believed

References

1. Reference: Sanhedrin 23b:9. Version Title: William Davidson Edition - English. Document Category: Talmud. URL: https://www.sefaria.org/Sanhedrin.23b.9.

2. Reference: Kiddushin 76b:13, Version Title: William Davidson Edition - English. Document Category: Talmud. URL: https://www.sefaria.org/Kiddushin.76b.13.

3. Reference: Mishneh Torah, Forbidden Intercourse 20:4. Version Title: Mishneh Torah, trans. by Eliyahu Touger. Jerusalem, Moznaim Pub. c1986-c2007. Document Category: Halakhah. URL:

https://www.sefaria.org/Mishneh_Torah,_Forbidden_Intercourse.20.4.l

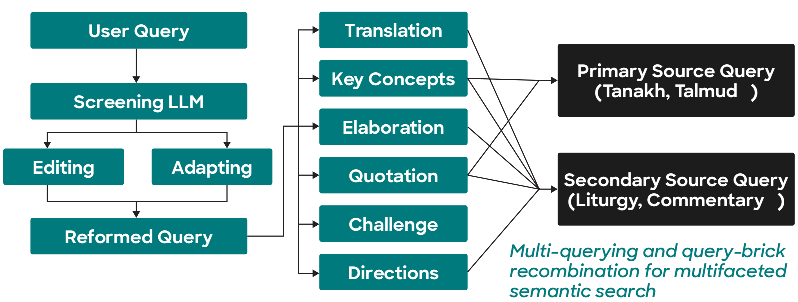
Reference: Mishneh Torah, Forbidden Intercourse 20:4. Version Title: Sefaria Community Translation. Document Category: Halakhah. URL:

https://www.sefaria.org/Mishneh_Torah,_Forbidden_Intercourse.20.4.

→ "Should have retrieved this one [Kiddushin.73b.8]"

Query Engineering, Fusion, & HyDE

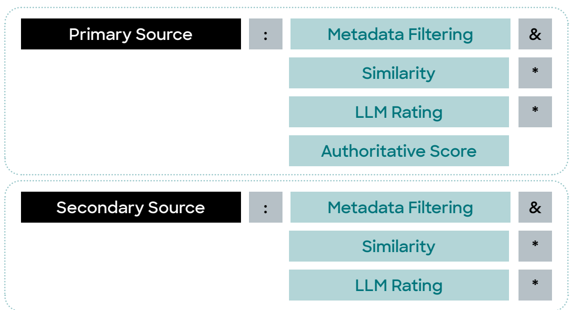
- User queries are firstly **screened, edited,** and **adapted** into **error-free, unbiased, serious,** and **pertinent** research-oriented questions.
- Several further query bricks are then evolved, including the English **translation, key words** and **concepts** associated with the query, an **elaborated version** of the query with more contexts, **(hypothetical) scriptural quotations** related to the query, **critiques** on the query, and **potential directions** to answer the question.
- These query bricks are then recombined into **primary source query** and **secondary source query** for retrieval.



Multi-querying and query-brick recombination for multifaceted semantic search

Metadata Filtering & Customized Reranking

- **Metadata** of retrieved references are used to limit the scope on primary sources (e.g., Tanakh, Talmud) or secondary sources (e.g., liturgy, commentary).
- The results are then ranked based on a weighted combination of **cosine similarity scores**, customized **LLM-based suitability ratings**, and/or **authoritative scores** specific to Sefaria data.
- The authoratative scores indicate the **degree of importance** of certain references in this context and are particular suitable to **highlight primary sources.**



The Deepset Recipe:

Unlocking the Mastery of Metadata for Filtering, Searching, and Reranking

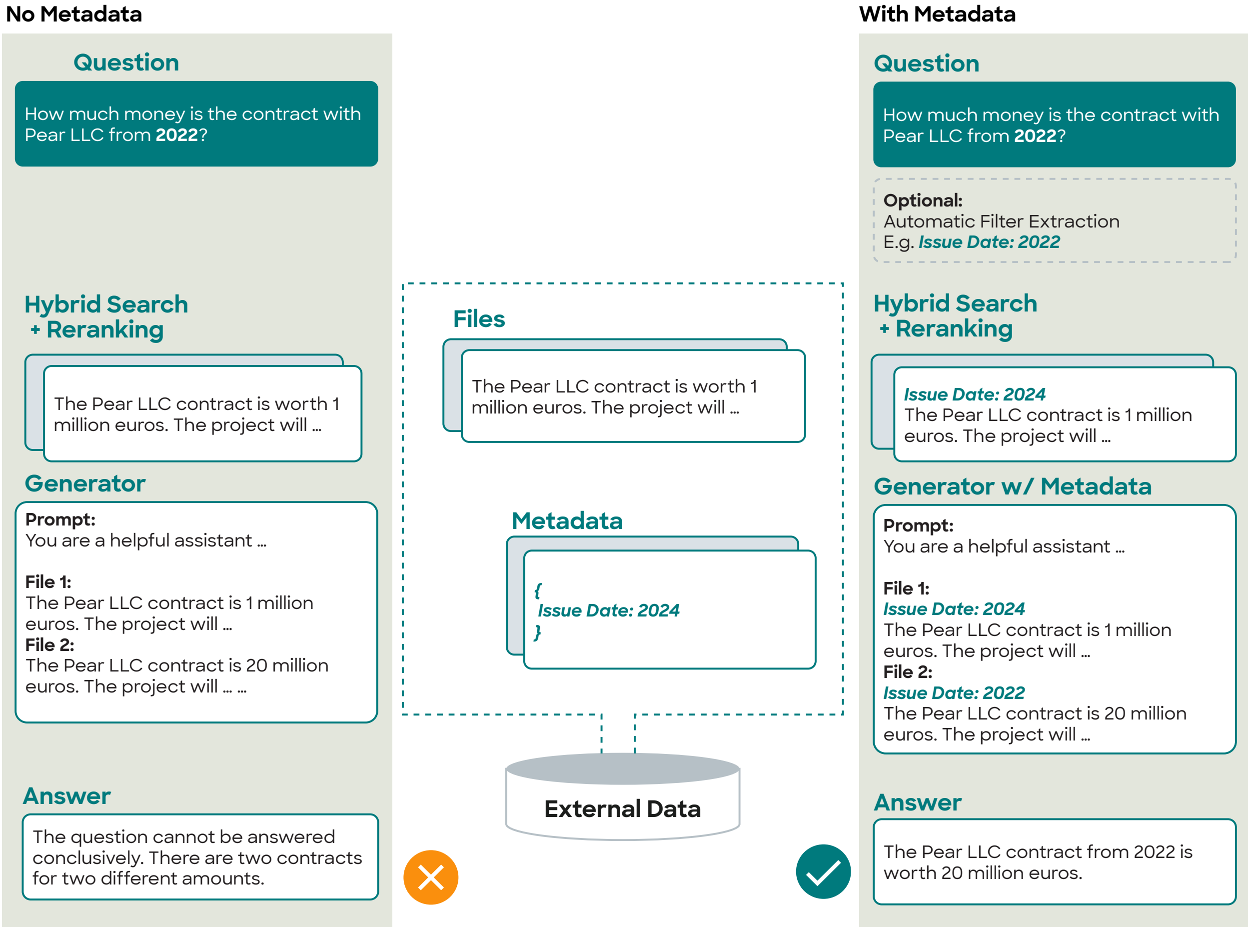
Background & Goals of this Recipe

- This recipe aims at enhancing document retrieval and answer quality by leveraging **metadata** to **preserve contextual understanding** across the pipeline of a RAG-based question answering system.
- This recipe has been applied in several use cases, expanding the functionality of RAG-based systems within **legal, academic,** and **media** domains, particularly in developing RAG-enabled chatbots for efficiently answering queries within large legal and newspaper corpora.

Challenges Addressed

- **Data Variability**: Handling **missing** or **inconsistent metadata** across different documents is a significant challenge.
- **Metadata Relevance:** Identifying **which metadata fields are relevant** for the types of questions that end-users will ask can be challenging.

Metadata for Enhancement of Relevance and Consistensy



Metadata for Hybrid Search

- Metadata can be utilized as **strict filters** that refine the scope of search results.
- Metadata can be used to guide **keyword-based search** algorithms (e.g. BM25) by including specific metadata fields, enhancing the relevancy of results.
- Metadata can be used also for **embedding retrieval** (vector search) through **vectorization** of both the document's **textual content** and selected **metadata fields** to improve search precision.

Metadata for Reranking

- **Semantic reranking models,** such as Cross Encoders, can incorporate metadata by **prepending its values to the document text,** ensuring that this additional context is factored into the reranking process.
- **Temporal or recentness reranking models** prioritize documents not just by relevance but also by recency, utilizing **date-based metadata** like the Issue Date to balance the importance of content freshness with topical pertinence.

The Jina AI Recipe:

Agent-Driven SQL Scoping and Task-Specific Finetuning for Patent Search

Background & Goals of this Recipe

- When assessing a new technology for **patenting**, legal experts must take much time to thoroughly comprehend the innovation, identify keywords for patent searches, and determine key elements of the patent application.
- This recipe creates a **co-pilot** to assist patent professionals in **interactively analyzing new patents, mitigating IP risks**, and **identifying innovation scope** through **optimized semantic search** and **patent retrieval** capabilities.

Challenges Addressed

- **Inadequate Precision**: Classic keyword matching, vector search, and reranking fall short for intricate, high-precision semantic matching.
- **Scoping for SQL Search:** Creating dynamic scoping filters and crafting reliable SQL queries for structured database searches presents significant complexities.
- **Domain-Specific Embeddings:** Tailoring an embedding model for a specific domain demands a carefully designed finetuning approach to optimize its discriminative power for domain-specific texts.

Query
Bicycle Equipped with Versatile Dual Chain Drive
...
two cranksets symmetrically mounted
...

Retrieved Contexts
Tandem bicycle
...

Step 1: Establish Baseline

[Key Techniques]
Embeddings + Vector DB

[Observations]
👍 Decent high-level semantic matching and retrieval.
👎 Detailed-level semantic matching and retrieval didn't work.

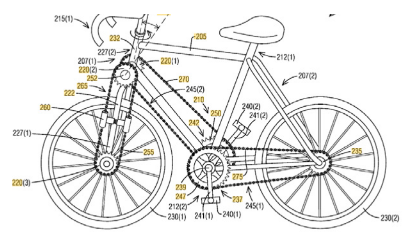
Query
Bicycle Equipped with Versatile Dual Chain Drive
...
two cranksets symmetrically mounted
...

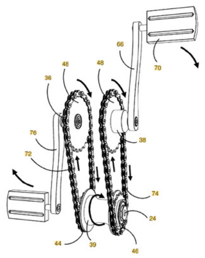
Retrieved Contexts
Dual pedaling mechanism bicycle

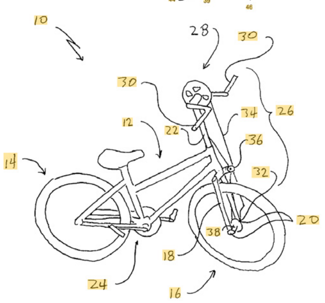
Step 2: Implement Out-of-the-Box Improvements

[Key Techniques]
Embeddings + Vector DB + Reranking (testing different combinations of embeddings and rerankers)

[Observations]
👍 Improved high-level semantic matching and retrieval.
👍 improved detailed-level semantic matching and retrieval.







Query
Bicycle Equipped with Versatile Dual Chain Drive
...
two cranksets symmetrically mounted
...

Retrieved Contexts
Two wheel drive bicycle with a dual chain-and-sprocket drive mechanism driven by a two-sided paddle-sprocket system
...

Step 3a: Apply Additional Optimization Techniques without Finetuning

[Key Techniques]
Embeddings + Vector DB + Reranking + Chunking + Agent-Driven Scoping & SQL Queries

[Observations]
👍 Detailed-level semantic matching and retrieval was significantly improved.

Query
Bicycle Equipped with Versatile Dual Chain Drive
...
two cranksets symmetrically mounted
...

Retrieved Contexts
Two wheel drive bicycle with a dual chain-and-sprocket drive mechanism driven by a two-sided paddle-sprocket system
...

Step 3b: Task-specific Finetuning of Embedding Models

[Key Techniques]
LoRA finetuning

[Observations]
👍 There is a potential for task-specific, "automated" optimization.
👍 Detailed-level semantic matching and retrieval was significantly improved.
👎 Finetuning demands the time-consuming and resource-intensive preparation of training data, which can be a costly endeavor.

Agent-Driven SQL Scoping

- **LLM-agents** are implemented to actively identify appropriate **response scopes** as well as **document filtering criteria.**
- The agents then construct corresponding **SQL queries** from natural language inputs, incorporating the suitable filters.
- The generated SQL queries are **enhanced** using LLM through **multiple iterations,** leading to improved retrieval result quality.

Task-specific Finetuning

- **Task-specific finetuning** focuses on learning private company knowledge, while **domain-specific finetuning** is broader, optimizing the model with public data within a professional field.
- **Triplets** of **[query, true answer, hard negative answer]**, were utilized for finetuning [7].
- Overall performance was dominated by the **quality of the hard negative answers.**
- In this case, **auto-tuning**, i.e., using a LLM to automatically generate high-quality training data, especially hard negative answers, appears a promising approach.
- Task-specific finetuning eventually outperforms domain-specific finetuning on NDCG@10.

The RAGAR Recipe:
Multimodal RAG-augmented Reasoning for Fact-checking

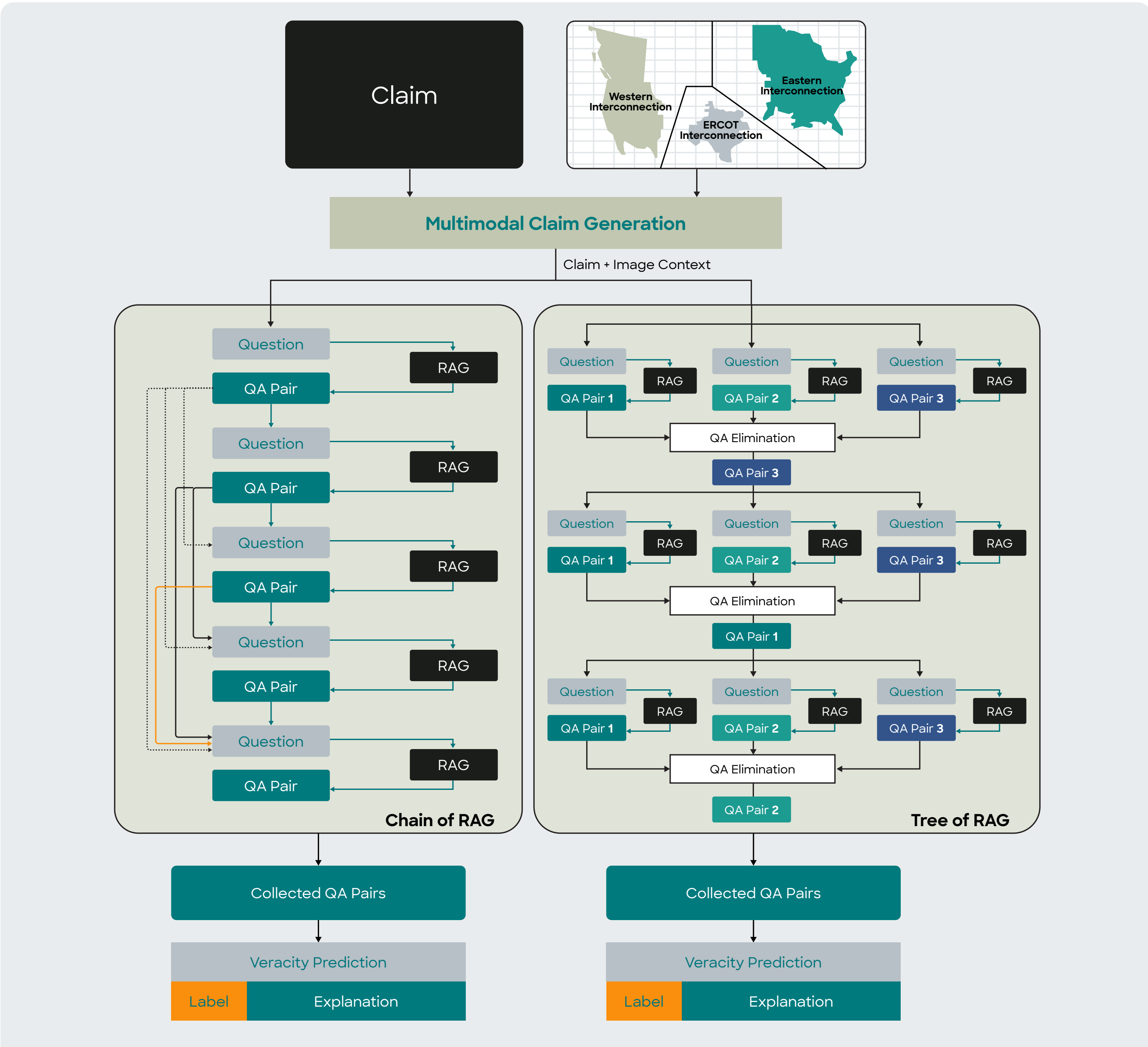
Background & Goals of this Recipe

- The increasing prevalence of misinformation, particularly in the context of political discourse, necessitates advanced solutions for fact-checking.
- This recipe introduces **RAG-augmented reasoning (RAGAR)** approaches, in particular **Chain of RAG (CoRAG)** and **Tree of RAG (ToRAG)** in combination with **Chain of Verification** to achieve evidence-based verification of open-ended multimodal claims [5].
- Originally designed for political fact-checking, RAGAR can be **repurposed** for various industry applications and is ideal for situations requiring **thorough verification**, **complex reasoning**, and **accurate fact-checking.**

Challenges Addressed

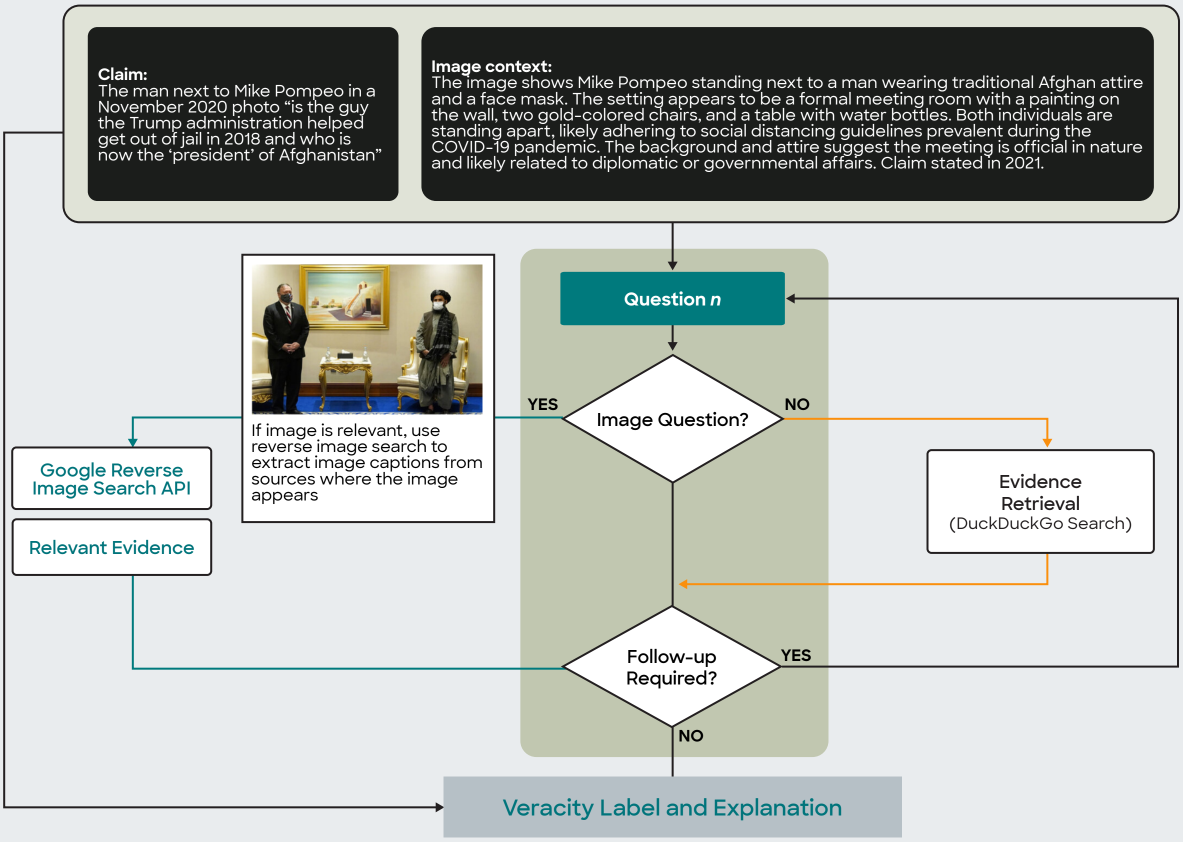
- **Contextualized Multimodal Claim Generation:** Generating claims from textual input, while selectively highlighting relevant contextual information and filtering out irrelevant details from the image, presents its own challenge.
- **Open-ended Multimodal Evidence Retrieval:** Another challenge involves retrieving text and image evidence from the internet while adhering to the correct time frame, filtering out inappropriate sources, and effectively utilizing image metadata.
- **Dynamic, Agentic Reasoning Actions:** Dynamically determining the most logical reasoning path to follow, and deciding when to stop based on sufficient evidence accumulation while applying chain of thought and tree of thought necessitates careful management of prompts and workflow.

The RAGAR Approaches: Chain of RAG (CoRAG) & Tree of RAG (ToRAG)

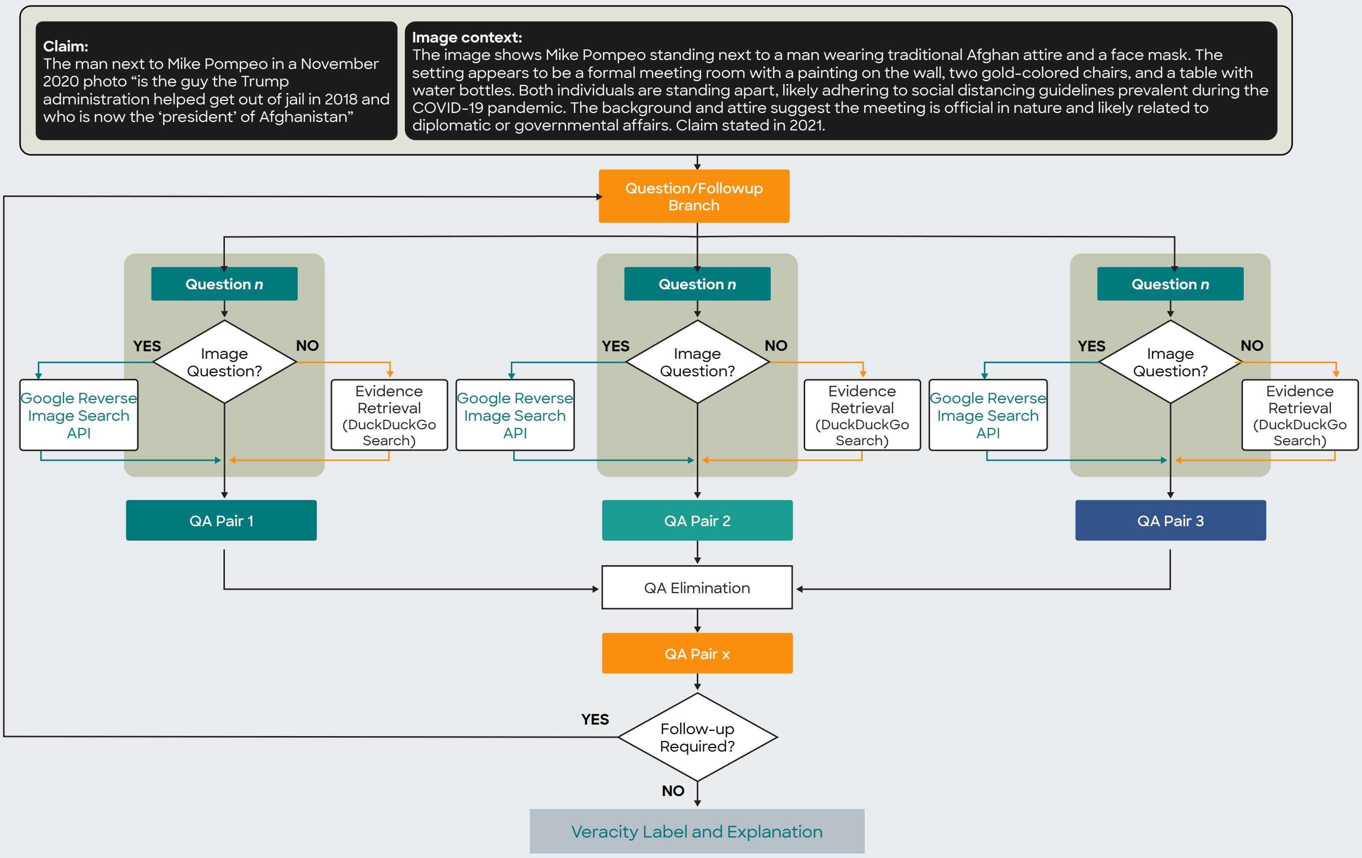


The RAGAR Recipe:
Multimodal RAG-augmented Reasoning for Fact-checking

RAG-Augmented Reasoning Technique (Chain of RAG)



RAG-Augmented Reasoning Technique (Tree of RAG)



The RAGAR Recipe:
Multimodal RAG-augmented Reasoning for Fact-checking

Chain of RAG & Tree of RAG

- **Chain of RAG (CoRAG):**
  - Uses **sequential follow-up questions** augmented from the RAG response to retrieve further evidence.
  - An **early termination check** step takes as input the generated claim and question-answer pair(s) and checks whether enough information to answer the claim has been gathered.
- **Tree of RAG (ToRAG):**
  - Creates **question branches** at each step of the reasoning.
  - In each step, the question-answer pairs are eliminated and only the **best question-answer branch** is chosen as the candidate evidence, based on the criteria of relevance, detail, additional information, and answer confidence.

RAG-enhanced Agents for Enterprises

- Unfolding developments of RAG-enhanced agents signal a **transformative potential in the realm of enterprise solutions** through the harmonization of GenAI and advanced information retrieval techniques and the analytical, reasoning, and agentic prowess of LLMs.
- This confluence empowers the advent of intelligent agents tailored for **complex enterprise applications**—from **decision-making** to **strategic planning**—heralding an era of RAG-enhanced agents equipped to navigate the nuanced demands of strategic enterprise contexts.

“The AI shouldn’t just answer; it should do research first to determine which of the answers are the best.”

Jenson Huang

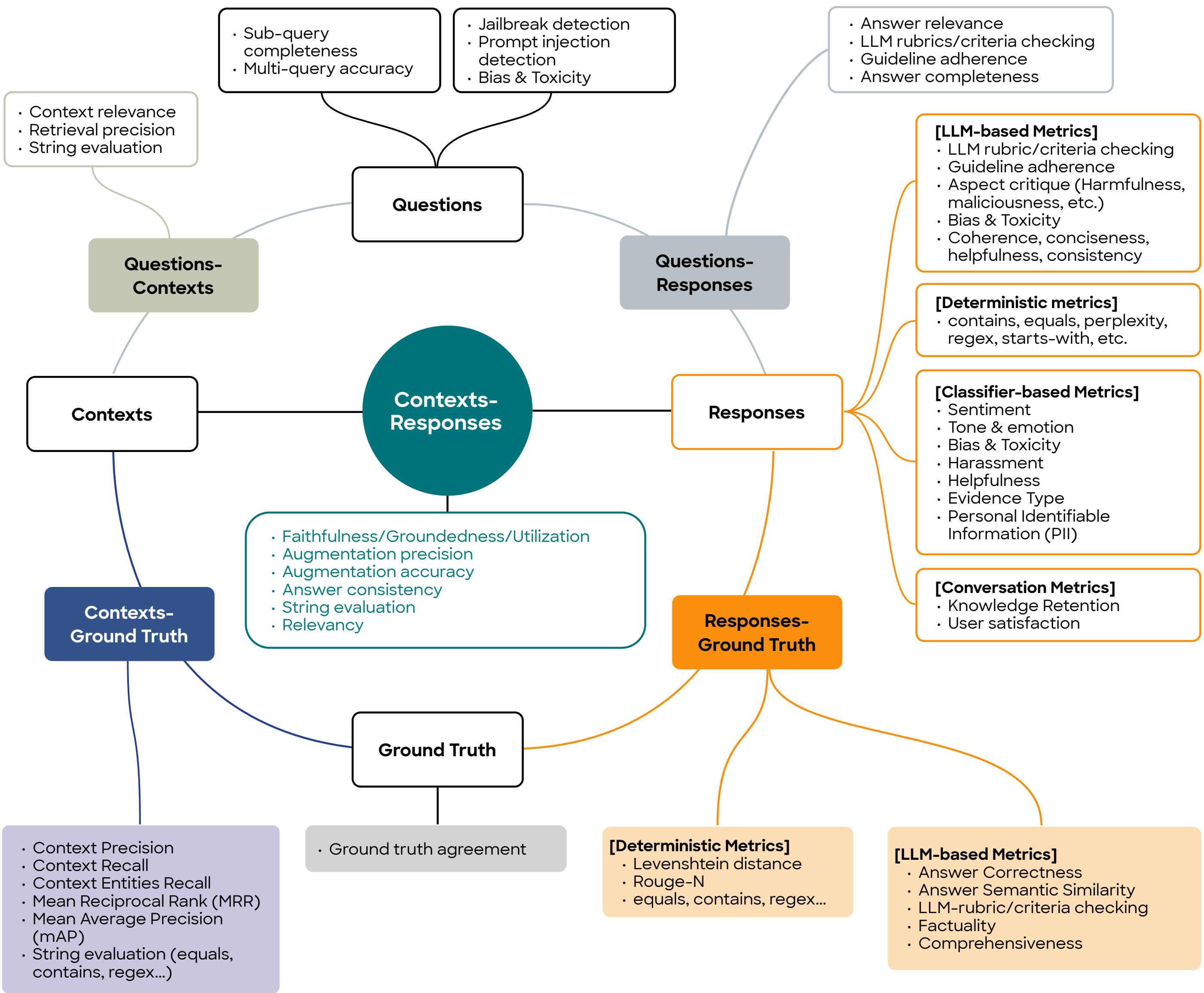
CEO, NVIDIA

Section 3:
A Deep Dive into RAG Evaluation & Metrics

Now, how do we assess RAG systems?

A Mind Map of RAG Metrics: What do they Measure?

- Overall, assessing the quality of a RAG system output requires consideration of 4 types of contents: **questions** (user queries), **contexts** (retrieved relevant references), **ground truth** (golden answers), and **responses** (final system output).
- In the contexts of RAG, besides assessment of a single type of contents, it is particularly important to consider the relationship (relevance, coherence, etc.) between (1) **questions and contexts**; (2) **contexts and ground truth;** (3) **contexts and responses;** (4) **responses and ground truth;** (5) **questions and responses.**
- Below is a **mind map** summarizing different metrics associated with various aspects of RAG contents, showing what these metrics assess.
- This compilation reflects **existing RAG metrics implemented in major RAG frameworks,** as outlined in the detailed table comparing major RAG evaluation frameworks on the next two pages.
- An **intuitive explanation of key RAG metrics**, including their calculation methods, is provided on page 28.



A Further Look into Existing RAG Evaluation Frameworks

Frameworks	Version	License	Integration: Compatibile Frameworks	Integration: Models	Metrics: Questions-Contexts	Metrics: Questions-Responses (given contexts)	Metrics: Contexts-Responses (given questions)	Metrics: Contexts-Ground Truth (given questions and/or responses)	Metrics: Responses-Ground Truth (given questions)	Metrics: Responses (given questions and/or contexts)	Metrics: GroundTruth	Metrics: Questions	Metrics: Conversation	Customized Metrics	Performance Metrics	Human Feedback	A/B Tests	LLM Vulnerability Scan	ML Vulnerability Scan	Model/Pipeline Comparison	Prompt Playground UI	Manual Test Set Creation UI	Automatic Test Set Creation	Evaluation UI	LLM Monitoring Dashboard	Easy Deployment	Team Management/ Collaboration
Giskard	2.10.0	Apache-2.0	Github, MLflow, Weights & Biases, DagsHub, HuggingFace, AVID, Pytest	Catboost, Hugging Face models, Langchain models, Pytorch models, Sklearn models, Tensorflow models	✗	· ragas_answer_relevancy	· ragas_faithfulness	· ragas_context_precision · ragas_context_recall	· Correctness (CoT) · LLM-as-a-judge · Ground Truth Similarity	✗	✗	✗	✗	✓	✗	✗	✗	· Hallucination and Misinformation (Plausibility & coherency) · Harmful Content Generation · Prompt Injection (character level, jailbreaking) · Robustness (LLM output coherency) · Output Formatting · Information Disclosure · Stereotypes and Discrimination	· Performance Bias · Unrobustness · Overconfidence · Underconfidence · Unethical behaviour · Data Leakage · Stochasticity · Spurious correlation	✗	✗	✗	✓	✓	✓	✓	✓
promptfoo	0.50.1	MIT license	GitHub Actions, GitLab CI, Jenkins, Jest, Mocha/Chai	OpenAI, Anthropic, Azure OpenAI models, Llama.cpp, Ollama, Google Vertex models, Google AI Studio, Generic webhook, Custom API Provider, Custom scripts, HuggingFace models, LocalAI models, Replicate models, Amazon Bedrock models, Cohere, Groq, Mistral AI models, OpenLLM, OpenRouter, Perplexity, text-generation-webui, Together AI, vllm	[Model-graded metrics] · context-relevance	[Model-graded metrics] · llm-rubric, answer · relevance- Similarity	[Model-graded metrics] · context-faithfulness	[Model-graded metrics] · context-recall	[Deterministic metrics] · contains, equals, Levenshtein distance, perplexity, regex, rouge-n, starts-with [Model-graded metrics] · llm-rubric, model-graded-closedqa, factuality- Similarity	· Model-graded metrics: llm-rubric, model-graded-closedqa, classifier grading · Sentiment, Tone and emotion, Helpfulness, Grounding, factuality, and evidence-type	✗	✗	✗	✗	· Latency · Cost	✗	✗	· Model-graded metrics: classifier grading - Toxicity	✗	✓	✓	✗	✓	✓	✓	✓	✗
Ragas	0.1.6	Apache-2.0	LlamaIndex, Langchain, Langsmith, Arize-Phoenix, Langfuse, Athina, Zeno, TonicValidate, Haystack	OpenAI models, Azure OpenAI models, Amazon Bedrock models, Google Vertex AI models	· Context Relevancy	· Answer Relevance	· Faithfulness	· Context Precision · Context Recall · Context entities recall	· Answer semantic similarity · Answer correctness	· Aspect Critique (harmfulness, maliciousness)	✗	✗	✗	✓	✗	✗	✗	· Aspect Critique	✗	✗	✗	✗	✓	✓	✓	✓	✓
DeepEval	0.21.15	Apache-2.0	LlamaIndex, Hugging Face	LlamaIndex models, Hugging Face models	ContextualRelevancyMetric	· SummarizationMetric · AnswerRelevancyMetric (RAGAS)	Faithfulness (RAGAS)	· ContextualRecallMetric (RAGAS) · ContextualPrecisionMetric (RAGAS)	· GEval	· HallucinationMetric · BiasMetric · ToxicityMetric · KnowledgeRetentionMetric	✗	✗	✗	✓	· Latency · Cost	✗	✗	· HallucinationMetric · BiasMetric · ToxicityMetric · KnowledgeRetentionMetric	✗	✓	✗	✗	✓	✓	✓	✓	✗
agenta	0.12.6	MIT license	Langchain, LlamaIndex, and "any others"	OpenAI models, Cohere, or local models, and "any others"	✗	✗	✗	✗	· Exact match · Regex match · Webhook evaluator (Correctness) · Similarity match (Jaccard) · AI Critic (LLM-based)	✗	✗	✗	✗	✓	✗	✓	✓	✗	✗	✗	✓	✓	✗	✓	✓	✓	✓
Trulens	0.27.0	MIT license	Langchain, LlamaIndex, NeMo, Guardrails	OpenAI models, AzureOpenAI models, Amazon Bedrock models, LiteLLM models, Langchain models	[Generation-based Stock Feedback Functions] context_relevance, qs_relevance	[Generation-based Stock Feedback Functions] relevance	[Generation-based Stock Feedback Functions] Combinators: Groundedness	✗	[Stock Feedback Functions] HF language_match [Generation-based Stock Feedback Functions] comprehensiveness	[Stock Feedback Functions] HF PII detection, HF positive sentiment, HF toxic, OpenAI moderation_harassment [Generation-based Stock Feedback Functions] coherence, conciseness, correctness, helpfulness	Combinators: Ground Truth Agreement (agreement_ measure, bert_score, bleu, mae, rouge)	✗	✗	✓	· Latency · Cost · Token Counts	✗	✗	[Classification-based Stock Feedback Functions] HF toxic, OpenAI moderation_harassment, OpenAI moderation_harassment_threatening, OpenAI moderation_hate, OpenAI moderation_ hatethreatening, OpenAI moderation_selfharm, OpenAI moderation_sexual, OpenAI moderation_ sexualminors, OpenAI moderation_violence, OpenAI moderation_violencegraphic [Generation-based Stock Feedback Functions] controversiality, criminality, harmfulness, insensitivity, maliciousness, misogyny, sentiment, stereotypes	✗	✗	✗	✗	✗	✓	✗	✗	✗
Tonic Validate	4.0.4	MIT license	LlamaIndex	OpenAI models, AzureOpenAI models	Retrieval precision	✗	· Augmentation precision · Augmentation accuracy · Answer consistency (binary) · Retrieval k-recall	✗	Answer Similarity	Contains Text	✗	✗	✗	✓	· Latency	✗	✗	✗	✗	✗	✗	✓	✗	✓	✓	✗	✓
LangChain	0.1.13	MIT license	Various	Various	· String Evaluator · Scoring Evaluator	· String Evaluator · Scoring Evaluator	· String Evaluator · Scoring Evaluator	· String Evaluator · Scoring Evaluator	· String Evaluator · Criteria Evaluation (correctness) · String Evaluator · Embedding Distance · String Evaluator · Exact Match, Regex Match · String Evaluator · Scoring Evaluator · Comparison Evaluators · Pairwise embedding distance	· String Evaluator · Criteria Evaluation (conciseness or custom) · String Evaluator · Custom String Evaluator · HF evaluate library (perplexity etc.) · String Evaluator · Scoring EvaluatorW · Comparison Evaluators · Pairwise string comparison · Comparison Evaluators · Pairwise embedding distance	· String Evaluator · Scoring Evaluator	✗	✗	✓	· Callback · Token counts	✓	✓	· String Evaluator · Criteria Evaluation (constitutional principles)	·	✓	✓	✗	✓	✗	✗	✗	✗
LangSmith	0.1.40	Closed beta	Various	Various	· LangChain evaluators	· LangChain evaluators	· LangChain evaluators	· LangChain evaluators	· LangChain evaluators	· LangChain evaluators	· LangChain evaluators	✗	✗	✓	· Latency · Cost · Token Counts	✓	✓	· Chat Bot Ben · chmarking using Simulation	✗	✓	✓	✓	✓	✓	✓	✓	✓
LlamaIndex	0.10	MIT license	UpTrain, DeepEval, Ragas, Tonic validate, and various others	Various	· RelevancyEvaluator · RetrieverEvaluator · ContextRelevancyEvaluator	· RetrieverEvaluator · ContextRelevancyEvaluator · AnswerRelevancyEvaluator · GuidelineEvaluator	· FaithfulnessEvaluator · RelevancyEvaluator	✗	· CorrectnessEvaluator · SemanticSimilarityEvaluator	✗	✗	✗	✗	✓	Cost	✓	PairwiseEvaluator	· GuidelineEvaluator	✗	✗	✗	✗	✓	✗	✗	✗	✗
Haystack	1.25	Apache-2.0	Beir, Basic Agent Memory Tool, Chainlit, Traceloop, and others	OpenAI models, Anthropic models, Cohere models, Hugging Face models, Amazon Bedrock models	✗	✗	✗	· Recall · Mean Reciprocal Rank (MRR) · Mean Average Precision (mAP)	· Exact Match (EM) · F1 · Semantic Answer Similarity (SAS)	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗	✗	✗	✓	✓	✓	✓	✓
Haystack	2.0	Apache-2.0	DeepEval, Context AI, Ragas, UpTrain, Gradient, fastRAG, Titan Takeoff Inference Server, and others	OpenAI models, Azure, OpenAI models, Google AI modes, Google Vertex AI models, Anthropic models, Cohere models, Hugging Face models, Amazon Bedrock/Sagemaker models, vLLM Invocation Layer, FastEmbed, Jina AI embedding models, INSTRUCTOR embedding models, Voyage AI embedding models, Llama.cpp models, Mistral models, mixedbread models, Ollama models	[RagasEvaluator] · CONTEXT_ RELEVANCY[DeepEvalEvaluator] · CONTEXTUAL_ RELEVANCE[UpTrainEvaluator] · CONTEXT_RELEVANCE	[RagasEvaluator] · ANSWER_ RELEVANCY[DeepEvalEvaluator] · ANSWER_ RELEVANCY[UpTrainEvaluator] · RESPONSE_RELEVANCE · RESPONSE_COMPLETENESS · RESPONSE_COMPLETENESS_ WRT_CONTEXT	[RagasEvaluator] · FAITHFULNESS · CONTEXT_ UTILIZATION[DeepEvalEvaluator] · FAITHFULNESS[UpTrainEvaluator] · RESPONSE_CONSISTENCY · FACTUAL_ACCURACY	[RagasEvaluator] · CONTEXT_PRECISION · CONTEXT_ RECALL[DeepEvalEvaluator] · CONTEXTUAL_PRECISION · CONTEXT_RECALL	[RagasEvaluator] · ANSWER_CORRECTNESS · ANSWER_ SIMILARITY[UpTrainEvaluator] · RESPONSE_MATCHING	· [RagasEvaluator] · ASPECT_ CRITIQUE[UpTrainEvaluator] · RESPONSE_CONCISENESS · CRITIQUE_LANGUAGE · CRITIQUE_TONE- GUIDELINE_ ADHERENCE	✗	✗	✗	✓	✗	✗	✗	✗	✗	✓	✗	✗	✗	✓	✓	✗	✗
Deepset Cloud	0.041	Closed source	Haystack integrated frameworks	Haystack models	Reference analysis (top-k optimization)	✗	Groundedness	✗	✗	✗	✗	✗	✗	✗	Latency	✓	✗	✗	✗	✓	✓	✓	✗	✓	✓	✓	✓
UpTrain	0.6.12	Apache-2.0	OpenAI Evals, LlamaIndex, Replicate, Hugging Face, Langfuse, Helicone, Zeno, and others	OpenAI models, Azure OpenAI models, Claude models, Mistral models, Ollama models, Together AI models, Anyscale models	· Context Relevance	· Response Relevance	· Context Utilization · Factual Accuracy · Context Conciseness · Context Reranking	✗	· Response Matching	· Response Completeness · Response Conciseness · Response Validity · Response Consistency · Language Features · Tonality · Code Hallucination	✗	· Sub-Query Completeness · Multi-Query Accuracy · Prompt Injection · Jailbreak Detection	· User Satisfaction	✓	✗	✗	✗	· Prompt Injection · Jailbreak Detection	✗	✓	✗	✗	✗	✓	✗	✗	✗

Unveiling the Mechanisms Behind These Metrics

An Intuitive View of Key RAG Metrics in Common Practice

Metric	What Does It Measure?	Formula in an Intuitive Form	Intuitive Explanation of Formula
Context Relevance	Relevance of retrieved contexts given the questions	# Relevant Sentences in Contexts / # Sentences in Contexts	The proportion of sentences in the retrieved contexts that are relevant to the questions
Context Recall	How much ground truth appear in retrieved contexts	# Ground Truth Sentences Attributable to Contexts / # Sentences in Ground Truth	The proportion of sentences in the ground truth that are attributable to the contexts
Precision@k	How many top-ranked k contexts are relevant to the ground truth	True Positives@k / True Positives@k + False Positives@k	The proportion of top-ranked k contexts that are relevant to the ground truth
Context Precision@k	Effectiveness of ranking ground-truth relevant contexts (at rank k)	Sum(Precision@Each Relevant Items in Top-k) / # Relevant Items in Top-k	Get a cumulative view of precision at each relevant item till rank k, and then average over the number of relevant items in top-k
Context Entities Recall	Alignment between retrieved contexts and ground truth in terms entities	# Entities in Both Contexts & Ground Truth / # Entities in Ground Truth	The proportion of entities in the ground truth that appear also in the contexts
Mean Reciprocal Rank (MMR)	An average of effectiveness of ranking the first ground-truth relevant context on top across multipe queries	Average (Reciprocal Rank of the First Relevant Context from Multiple Queries)	Averaging the reciprocal rank of the first relevant context over multiple queries. The earlier the first relevant context appears, the better.
Mean Average Precision (mAP)	An average of effectiveness of ranking ground-truth relevant contexts (at rank k) across multipe queries	Average (Context Precision@k from Multiple Queries)	Averaging Context Precision@k over multiple queries
Answer Semantic Similarity	Similarity between response and ground truth	Similarity(Response, Ground Truth)	Cosine similarity between the response and the ground truth in the embedding space
Answer Correctness	Accuracy of answers when compared to the ground truth	Weighted Average(Answer Semantic Similarity, Factual Correctness F1 (Response, Ground Truth))	Weighted average between answer semantic similarity and factual correctness (F1 based on factual overlap)
Answer Relevance	Relevance of answers given the questions	Average(Similarity(Original Question, Artificial Reverse-Engineered Question [i]))	Averaging cosine similarity of the original question to artificial questions generated based on the answer
Answer Completeness	Completeness of responses given the questions	# Aspects Asked in Question and Answered in Response / # Aspects Asked in Question	The proportion of aspects asked in the question that are answered in the response
Faithfulness Groundedness	Factual consistency between the answers and the given contexts	# Generated Claims Attributable to Contexts / # Generated Claims in Response	The proportion of generated claims in the response that are attributable to the contexts

Reflections and Future Opportunities in RAG Technology

Reflections

Section 1: RAG Industrialization - Landscape & Strategy

- From a sustainable perspective, RAG solutions are crucial for industrial knowledge retrieval and question-answering systems, considering factors such as **trustworthiness, consistency, controllability, auditability, explanability, transparency, process optimization, IP & data secrety, cost efficiency,** and **scalability.**
- To harness RAG solutions, it is crucial to explore **advanced, modular techniques** for optimizing retrieval, augmentation, and generation, such as HyDE or RAGAR. Finetuning models might be necessary as a last resort, and should be evaluated early on. The complexity and potential benefits must be balanced against **resource constraints** and **development challenges.**
- Recognizing **realistic challenges** early in the **RAG industrialization journey** is essential for prioritizing development tasks and mitigating risks. For example, limited resources at the start may hamper creating a thorough evaluation dataset based on expert feedback.

Section 2: RAG Recipes for Real-World Challenges

- We tackled realistic challenges such as **initial scarcity of evaluation data** for determining chunking and embedding methods, adapting to **multifaceted contextual knowledge** and **domain-specific conventions,** enhancing **document relevance** and **consistency** through **metadata**, constructing **SQL queries** for targeted search, **task-specific finetuning,** and employing **multimodal RAG-augmented reasoning (RAGAR)** for input verification.
- Overall, **retrieval quality** emerges as the key area requiring improvement in RAG development. **Cost-effective strategies** like **metadata filtering, query engineering, fusion, hybrid search,** and **HyDE** are initial considerations. Subsequently, more sophisticated **agentic approaches** could further enhance quality, aligning with observations in the complexity/contribution map in Section 1.

Section 3: A Deep Dive into RAG Evaluation & Metrics

- Assessing RAG systems is complex due to the **interplay among questions, contexts, ground truth, and responses,** along with the need to evaluate these components **individually**. Key metrics encompass context relevance, recall, precision, answer semantic similarity, correctness, relevance, faithfulness, etc.
- We're observing a trend of **growing LLM frameworks tailored for various aspects of RAG evaluation.** However, while the metrics in these frameworks align with the mind map presented at the outset of Section 3, there isn't a single framework that fully encompasses every aspect of RAG evaluation to date.
- While these metrics capture specific facets of RAG, it's important to acknowledge their **occasional insufficiency.** For instance, evaluating the quality of open-ended questions can pose challenges as there's no fixed set of golden references for such queries.

Future Opportunities in RAG Technology

Seamless Integration

The future of RAG holds the potential for **seamless integration into a wide array of knowledge retrieval and question answering applications**, such as search engines, customer service platforms, in-car assistants, social media, and knowledge management systems, enhancing user experiences with context-aware responses.

Reasoning & Agents

Integrating reasoning and agent capabilities into future RAG solutions enhances **precision** and **factuality** by actively assessing content l**ogic, quality,** and **consistency** while autonomously **adapting to user needs** and **taking suitable actions.**

Evaluation Framework

The industry anticipates a **standardized** and **generalized** framework to comprehensively evaluate various aspects of RAG systems across different development stages. Such a framework would ensure **consistent assessment** of quality, reliability, and scalability, guiding improvements throughout the **RAG industrialization journey.**

Coordinated Modules

Currently, LLM modules for retrieval may not interpret user intent in the same way as those for generation, leading to inconsistencies. Hence, there is potential for these modules to achieve **better communication and coordination** in the future, ensuring a more unified understanding and response.

Cross-Modal Capabilities

The future of RAG envisions the integration of **multimodal data sources,** including **text, images, video, audio,** and other types of **(un)structured data**, to provide richer and more comprehensive responses, leveraging diverse data types to enhance information retrieval and generation.

Sustainability & Long Context Window

In the near future, RAG solutions may remain a top option for industry due to factors like controllability (see section 1). Ensuring **long-term viability** is crucial, especially as LLMs evolve with **longer context windows**, likely **complementing rather than replacing RAG.** RAG systems should factor in this evolution for future extensibility in their design.

References

[1] H. Ye, T. Liu, A. Zhang, W. Hua, and W. Jia, “Cognitive Mirage: A Review of Hallucinations in Large Language Models.” arXiv, Sep. 13, 2023. Accessed: Sep. 19, 2023. [Online]. Available: http://arxiv.org/abs/2309.06794.

[2] Y. Gao et al., “Retrieval-Augmented Generation for Large Language Models: A Survey.” arXiv, Jan. 03, 2024. Accessed: Jan. 06, 2024. [Online]. Available: http://arxiv.org/abs/2312.10997.

[3] S. Barnett, S. Kurniawan, S. Thudumu, Z. Brannelly, and M. Abdelrazek, “Seven Failure Points When Engineering a Retrieval Augmented Generation System.” arXiv, Jan. 11, 2024. Accessed: Feb. 21, 2024. [Online]. Available: http://arxiv.org/abs/2401.05856.

[4] Z. Wang, A. Liu, H. Lin, J. Li, X. Ma, and Y. Liang, “RAT: Retrieval Augmented Thoughts Elicit Context-Aware Reasoning in Long-Horizon Generation.” arXiv, Mar. 08, 2024. Accessed: Mar. 14, 2024. [Online]. Available: http://arxiv.org/abs/2403.05313

[5] M. A. Khaliq, P. Chang, M. Ma, B. Pflugfelder, and F. Miletić, “RAGAR, Your Falsehood RADAR: RAG-Augmented Reasoning for Political Fact-Checking using Multimodal Large Language Models.” arXiv, Apr. 18, 2024. Accessed: Apr. 19, 2024. [Online]. Available: http://arxiv.org/abs/2404.12065

[6] T. D. Nguyen et al., “AstroLLaMA: Towards Specialized Foundation Models in Astronomy.” arXiv, Sep. 12, 2023. Accessed: Sep. 13, 2023. [Online]. Available: http://arxiv.org/abs/2309.06126.

[7] M. Günther et al., “Jina Embeddings 2: 8192-Token General-Purpose Text Embeddings for Long Documents.” arXiv, Feb. 04, 2024. doi: 10.48550/arXiv.2310.19923.

"Generative AI's impressive natural-language processing, combined with RAG's capabilities, revolutionizes knowledge management and decision-making by allowing employees to retrieve stored internal knowledge and manage information about products or processes swiftly and effectively, just as they would when asking a human."

Bernhard Pflugfelder
Head of Innovation Lab (GenAI), applied AI initiative

Authors

Dr. Paul Yu-Chun Chang

Senior AI Expert: Foundation Models - LLMs,
appliedAI Initiative GmbH
p.chang@appliedai.de

Paul Yu-Chun Chang works as an Senior AI Expert specializing in Large Language Models at appliedAI Initiative GmbH. He has 10 years of interdisciplinary research experience in computational linguistics, cognitive neuroscience, and AI, and 6 years of industrial experience in developing AI algorithms in language modeling and image analytics. Paul holds a PhD in Linguistics from LMU Munich, where he integrated NLP and machine learning methods to study brain language cognition.

Bernhard Pflugfelder

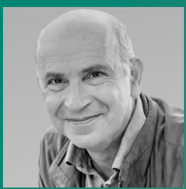
Head of Innovation Lab (GenAI),
appliedAI Initiative GmbH
b.pflugfelder@appliedai.de

Bernhard Pflugfelder works as Head of Innovation Lab (GenAI) at the appliedAI Initiative GmbH. Bernhard has 15 years of experience in the fields of Data Science, Natural Language Processing (NLP), as well as data and AI across different companies such as BMW Group or Volkswagen Group. He is renowned for his expertise especially in the field of AI in general, NLP and Generative AI in particular.

Contributors


Johannes Birk
Generative AI Engineer,
appliedAI Initiative GmbH
j.birk@appliedai.de


Emre Demirci
Jun. AI Engineering LLM,
appliedAI Initiative GmbH
e.demirci@appliedai.de


Antoine Leboyer
Managing Director,
TUM Venture Labs
antoine.leboyer@
unternehmertum.de


Lev Eliezer Israel
Chief Product Officer,
Sefaria
lev@sefaria.org


Noah Santacruz
Senior Research Engineer,
Sefaria
noah@sefaria.org


Damian Depaoli
AI Engineer,
appliedAI Initiative GmbH
d.depaoli@appliedai.de


Hadara Steinberg
Associate Director, Engineering Team,
Sefaria
hadara@sefaria.org


Dr. Sebastian Husch Lee
Solution Engineering Tech Lead,
deepset GmbH
sebastian.huschlee@deepset.ai


Dr. Saahil Ognawala
Head of Product,
Jina AI GmbH
saahil.ognawala@jina.ai


Maximilian Werk
Head of Engineering,
Jina AI GmbH
maximilian.werk@jina.ai


Mohammed Abdul Khaliq
Institut für Maschinelle
Sprachverarbeitung,
University of Stuttgart
mohammed.abdul-khaliq@ims.uni-
stuttgart.de


Mingyang Ma
Principal AI Strategist & Product
Manager,
appliedAI Initiative GmbH
m.ma@appliedai.de

Contributing Companies

About appliedAI Initiative GmbH

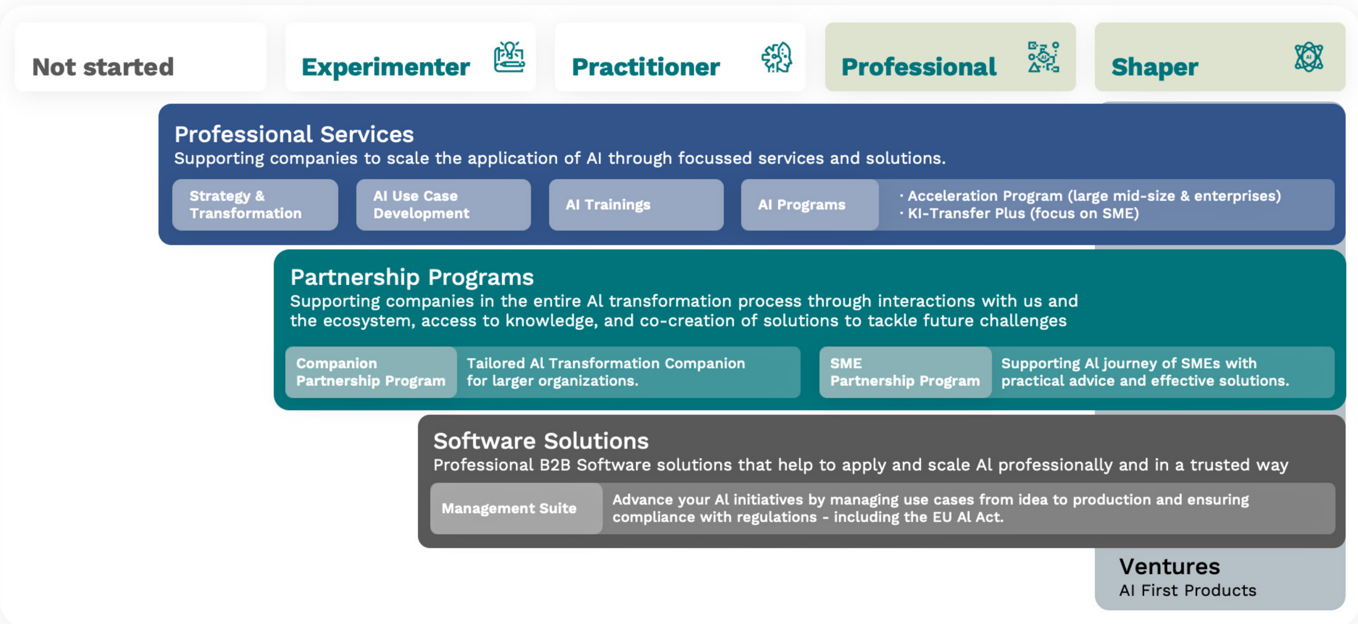
appliedAI is Europe's largest initiative for the application of trusted AI technology. It aims to advance Europe's industry to stay competitive in the Age of AI. The initiative was established in 2017 by Dr. Andreas Liebl as a division of UnternehmerTUM in Munich and transferred to a joint venture with Innovation Park Artificial Intelligence (IPAI) in Heilbronn in 2022.

At appliedAI, more than 160 employees work together with >20 companies in our **Partnership** to create best practices on how to apply AI, in **Professional Services and Accelerator Programs** to engineer AI powered solutions, develop **AI strategies and operating models** as well as **upskill** thousands of employees and managers. Moreover, appliedAI offers an **software for managing the AI application portfolio** to enhance AI Act compliance. appliedAI holistically supports international corporations, like BMW, Porsche, or Siemens, as well as medium-sized companies in their AI transformation.

For more information, please visit

https://www.appliedai.de/en/

We offer a unique set of offerings to help companies on their way to becoming AI shapers



Acknowledgement

We express our sincere appreciation for the remarkable work carried out by the authors, reviewers, and designer of this white paper. Their great expertise, motivation and dedication to every detail made this white paper an exceptional contribution to the AI Community.

Furthermore, we extend our gratitude to the contributors from Deepset, Jina.ai, Sefaria, and TUM Venture Labs for their valuable contributions. Their profound expertise and commitment have played a pivotal role in shaping the ideas and knowledge presented in this paper.

We would also like to thank all the appliedAI partner companies that actively participated in the appliedAI RAG Roundtables and bilateral interactions. Your involvement has been indispensable, providing fundamental references and serving as a constant source of inspiration.

The content presented in this white paper has been influenced and inspired by discussions and exchanges within the appliedAI Roundtables on Retrieval Augmented Generation in October 2023 and April 2024, including the appliedAI industry partners Atruvia AG, BMW Group, Giesecke+Devrient GmbH, EnBW Energie Baden-Württemberg AG, Infineon Technologies AG, Linde Group, Miele & Cie. KG, Munich Re, Group, MTU Aero Engines AG, NVIDIA, Roche Diagnostics GmbH, Rohde & Schwarz GmbH & Co. KG, Siemens AG, Telekom Deutschland GmbH and Wacker Chemie AG.

The collective expertise, exchange and dedication to advancing the knowledge in Generative AI and RAG was a great inspiration throughout the process of creating this white paper.

Retrieval-augmented Generation Realized: Strategic & Technical Insights for Industrial Applications

appliedAI Initiative GmbH

August-Everding-Straße 25
81671 München
Germany
www.appliedai.de