



RAG for MS-ADS Website

Team: Nexus Intelligence

Daichi Ishikawa
Liang Gong
Haoran Zheng

Live UI Link: <https://rag-uchicago-website.streamlit.app/>
GitHub Link: <https://github.com/daichi6/rag-uchicago-website>

November 2024

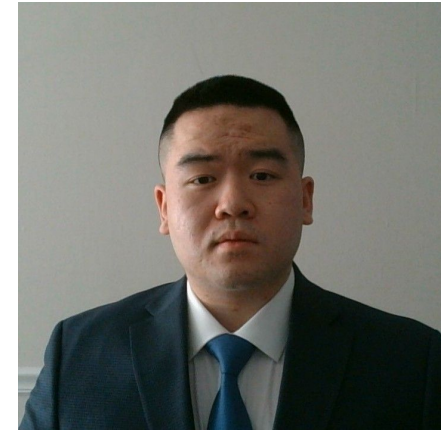
Our Team



Daichi Ishikawa



Liang Gong



Haoran Zheng

Agenda

- Business Problem
- Main Architecture
- Implementation Process
 - 1. Website Review
 - 2. Data Collection and Cleaning
 - 3. Document Creation and Chunking
 - 4. Creating Vector Database
 - 5. Finding Relevant Chunks and Generating Answer
 - 6. Routing for further improvement
 - 7. RAG Fusion for further improvement
- Live Demo
- Evaluation
- Summary
- Future Work
- Appendix

Business Problem



Current Problem

Students interested in the MSADS program often spend a lot of time searching for relevant information on a cluttered website.

No budget seems to be allocated for hiring an external development team to implement this tool.

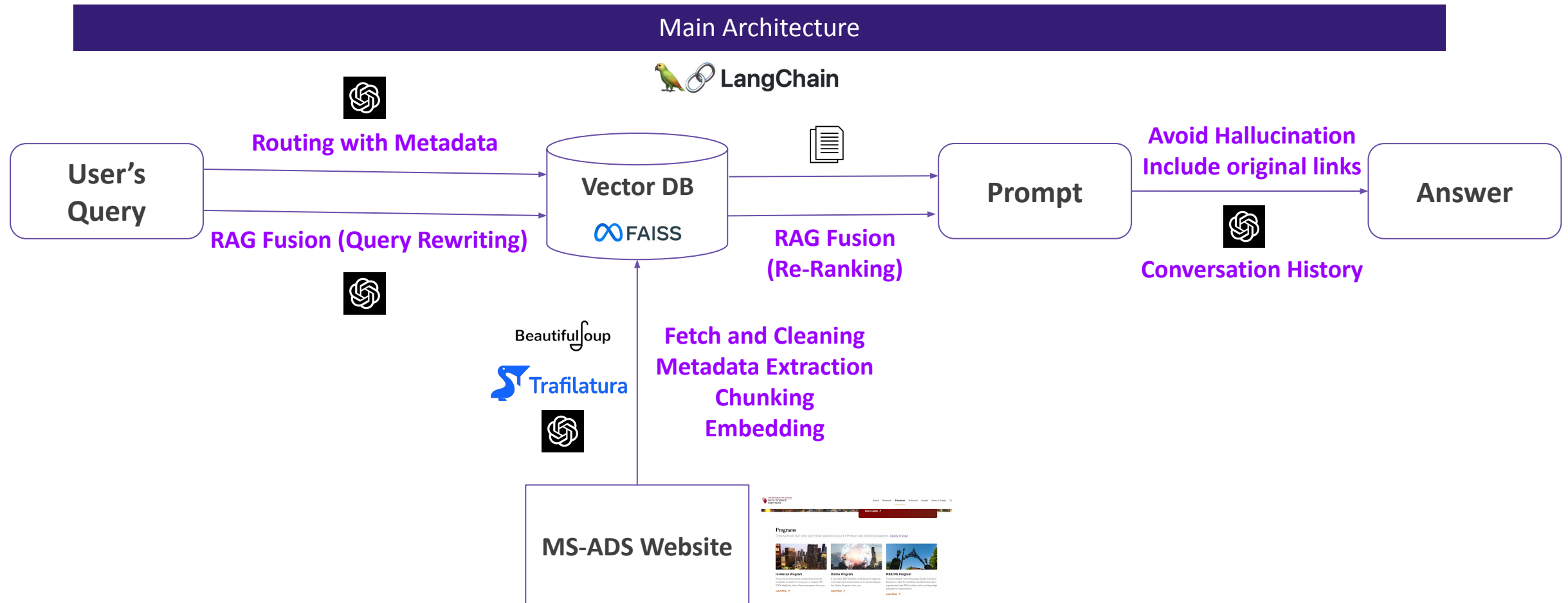
Expected Outcome

Chatbot to find answers accurately and quickly

Midterm project for GenAI students

Main Architecture

- OpenAI are used for LLM and embeddings, and FAISS as the vector database.
- We achieved accuracy improvements through the use of routing and RAG fusion.
- We also crafted prompts to avoid hallucination and ensure that original information sources are included in the response.
- Additionally, our chatbot is designed to make effective use of conversation history within the LLM.



1. Website Review

- The program includes a comprehensive curriculum covering machine learning, data engineering, and data visualization, along with a capstone project for practical experience. The program's expert faculty, strong industry connections, and career support services equip graduates for successful careers across various industries.
- The website provides information on the curriculum, faculty, admissions, career outcomes, and program formats (in-person and online).

<https://datascience.uchicago.edu/education/masters-programs/ms-in-applied-data-science/tuition-fees-aid/>

<https://datascience.uchicago.edu/education/masters-programs/ms-in-applied-data-science/capstone-projects/>

<https://datascience.uchicago.edu/education/masters-programs/ms-in-applied-data-science/online-program/>

<https://datascience.uchicago.edu/education/masters-programs/ms-in-applied-data-science/our-students/>

<https://datascience.uchicago.edu/education/masters-programs/ms-in-applied-data-science/>

<https://datascience.uchicago.edu/education/masters-programs/ms-in-applied-data-science/career-outcomes/>

<https://datascience.uchicago.edu/education/masters-programs/ms-in-applied-data-science/instructors-staff/>

<https://datascience.uchicago.edu/education/masters-programs/ms-in-applied-data-science/in-person-program/>

<https://datascience.uchicago.edu/education/masters-programs/ms-in-applied-data-science/how-to-apply/>

<https://datascience.uchicago.edu/education/masters-programs/ms-in-applied-data-science/faqs/>

<https://datascience.uchicago.edu/education/masters-programs/ms-in-applied-data-science/course-progressions/>

<https://datascience.uchicago.edu/education/masters-programs/ms-in-applied-data-science/events-deadlines/>

2. Data Collection and Cleaning

Fetching Web Content

- Retrieves raw HTML content from provided URLs using requests.
- Handles SSL verification and timeouts for reliability.
- Function: `fetch_web_content`

Metadata Extraction

- Extracts critical **metadata** such as: title, keywords, description, author, and the domain.
- Utilizes **BeautifulSoup** to parse and extract metadata from HTML.
- Includes the page's fetch date to track when the data was collected.
- Function: `extract_metadata`

Content Cleaning

- Cleans and extracts the main content from HTML using **trafilatura**.
- Ensures only the essential, readable content is retained, **removing unnecessary HTML tags**.
- Function: `clean_content`

3. Document Creation and Chunking

Document Creation

- Combines metadata and cleaned content to create a structured Document object.
- Stores URL, domain, and content type for further processing.
- Function: `create_document`

Document Splitting

- **`RecursiveCharacterTextSplitter`** to split large documents into smaller, more manageable chunks (**default 1000 characters, 200-character overlap**).
- Ensures that each chunk can be indexed and queried independently.
- Function: `split_document`

Add Metadata to Chunks

- Adds metadata to each chunk, including:
 - `'page_type'`,
 - `'primary_category'`,
 - `'subcategory'`,
 - `'title'`,
 - `'chunk_index'`,
 - `'total_chunks'`,
 - `'source'`

4. Creating Vector Database

Enhanced metadata: {'source': '<https://datascience.uchicago.edu/education/masters-programs/ms-in-applied-data-science/instructors-staff/>', 'domain': 'datascience.uchicago.edu', 'fetch_date': '2024-11-01T19:31:59.474593', 'content_type': 'web_page', 'title': 'Faculty, Instructors, Staff – DSI', 'chunk_index': 3, 'total_chunks': 62, 'chunk_size': 750, 'primary_category': 'ms-in-applied-data-science', 'subcategory': 'instructors-staff', 'page_type': 'Ms In Applied Data Science - Instructors Staff'}

Enhancing Metadata

- Enhances document **metadata by parsing URL paths (e.g., /how-to-apply/)**.
- Extracts primary category and subcategory from URL to enrich metadata.
- Adds human-readable `page_type` to indicate the document's content type.
- Function: `enhance_metadata`

Embedding

- Initializes **OpenAI's embedding model (text-embedding-3-small by default)** to convert documents into vectors for similarity search.
- Function: `init_embeddings` (model: str = "text-embedding-3-small")

Vector Database

- Creates a **FAISS** vector database from the document embeddings.
- Saves the FAISS vector store locally for efficient retrieval.
- Benefits: Fast approximate nearest neighbor (**ANN**) search, ensuring quick document retrieval based on vector similarity.
- Function: `create_and_save_vectordb`

5. Finding Relevant Chunks and Generating Answer

Loading the Vector Database

- Purpose: Loads **FAISS** vector database for efficient access to document embeddings.
- Embedding Initialization: Uses **OpenAI's text-embedding-3-small** for semantic vector generation.
- Error Handling: Catches and logs errors during loading for robust system operation.
- Parameterization: Allows flexible model choice and database path.
- Function: `load_vectordb`

Performing Similarity Search

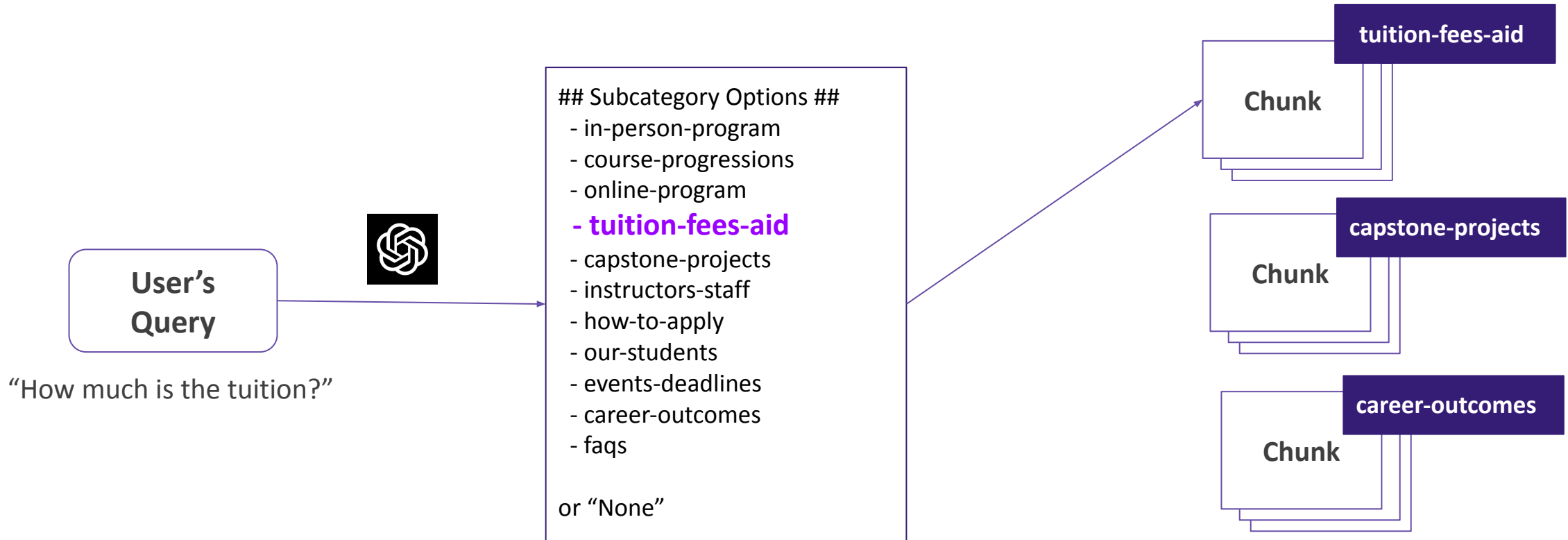
- Purpose: Retrieves top-k similar document chunks based on the user's query.
- Metadata Filtering: Optionally filters results by fields, enhancing relevance.
- Output: Returns a list of Document objects **with content and metadata**.
- Function: `search_similar_chunks(vectorstore, query, k=5, filter_dict)`

Generating answer

- Use GPT-4o
- Following adjustments to the prompt:
- 1) added phrasing to **avoid hallucination**
- 2) included **original links** in the response for **validation and detailed information**.

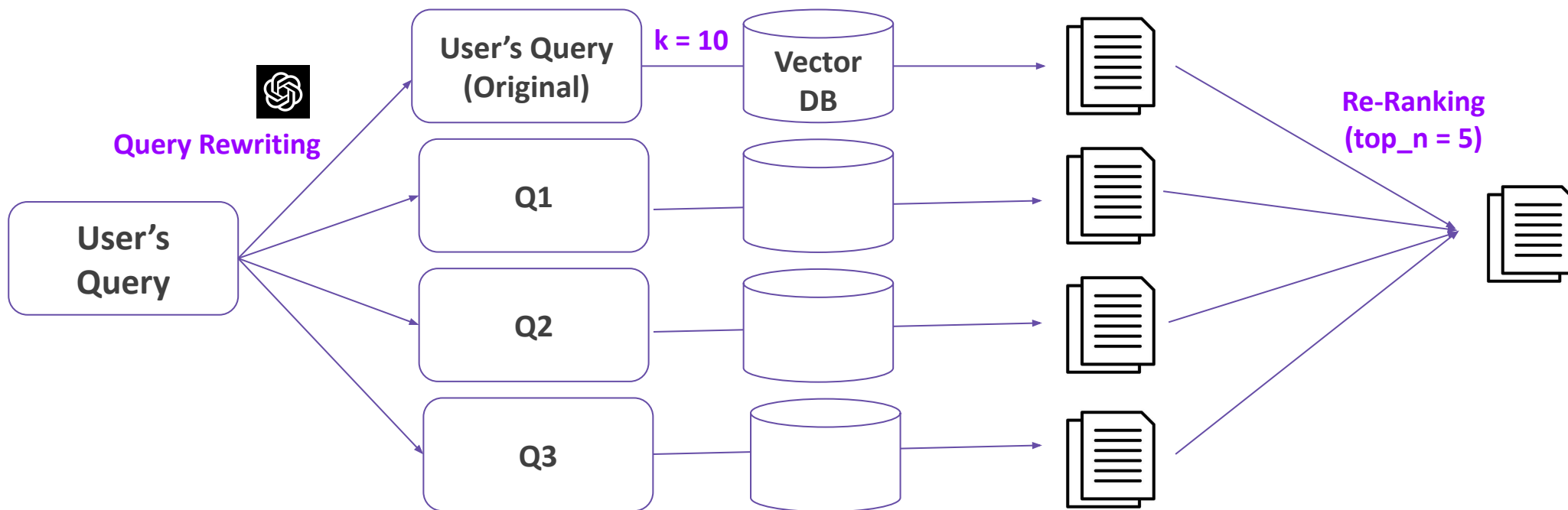
6. Routing for further improvement

- The goal is to link user queries with metadata to search for the appropriate chunks.
- Using an LLM enabled flexible classification of user queries.
- Subcategories were determined using the final part of the URL.



7. RAG Fusion for further improvement

- Through query rewriting, we were able to retrieve a wider variety of chunks, while re-ranking allowed us to successfully obtain chunks with the highest relevance to the query.



LIVE DEMO



<https://rag-uchicago-website.streamlit.app/>

Evaluation



AI Assistant for the UChicago MS-ADS Program

💡 Ask any questions about the MS-ADS Program

👤: What are the deadlines for the in-person program?

🗓️: The deadlines for the in-person program of the MS in Applied Data Science at the University of Chicago are as follows:

- November 7, 2024 – Priority Application Deadline
- December 4, 2024 – Scholarship Priority Deadline
- January 21, 2025 – International Application Deadline (requiring visa sponsorship from UChicago)
- March 4, 2025 – Second Priority Application Deadline
- May 6, 2025 – Third Priority Application Deadline
- June 23, 2025 – Final Application Deadline

In-person application decisions are released approximately 1 to 2 months after each respective deadline.

For more information, you can visit the following URL: [Events & Deadlines – DSI](#)

Enter your question here:

AI Assistant for the UChicago MS-ADS Program

💡 Ask any questions about the MS-ADS Program

👤: Is there an application fee waiver?

🗓️: Yes, there is a mention of an application fee waiver. For questions regarding an application fee waiver, applicants are advised to refer to the Physical Sciences Division fee waiver policy.

For more information, you can visit the following URL: [How to Apply – DSI](#)

👤: Can you answer with a more friendly tone?

🗓️: Absolutely! It looks like you're curious about the application fee waiver for the University of Chicago's MS in Applied Data Science program. Great news—there is indeed an application fee waiver available! To get all the details, you'll want to check out the Physical Sciences Division fee waiver policy. For more information, feel free to visit this link: [How to Apply – DSI](#). If you have any more questions, just let me know!

Enter your question here:

Evaluation - Conversation History

👤: What are the admission requirements for the MS in Applied Data Science program?

🤖: The admission requirements for the MS in Applied Data Science program at the University of Chicago include:

1. **Application Form:** Complete the online application form.
2. **Transcripts:** Submit unofficial transcripts from all previous colleges and universities attended. Official transcripts are required if admitted.
3. **Letters of Recommendation:** Two letters of recommendation are required. At least one should be from a direct manager/supervisor or internship supervisor who can attest to skills demonstrated in a professional setting. Letters from family members, friends, or peers are not accepted.
4. **Candidate Statement:** A candidate statement is required, which should not exceed 1,000 words. It should clearly present your aims and qualifications without restating your resume.
5. **Resume/CV:** A detailed resume including work experience, major accomplishments, awards, educational background, and professional conferences attended.
6. **Programming Supplement:** A PDF demonstrating current programming language skills, preferably in R or Python, is required. The document should be no more than two pages.
7. **Virtual Portfolio:** Short, recorded video responses to given prompts within the application. One video is required, and another is optional, each approximately 1:30 minutes in length.
8. **English Language Proficiency:** For non-native English speakers, proof of English language proficiency is required. Minimum scores are TOEFL 102 or IELTS 7.
9. **Application Fee:** A \$90 non-refundable application fee is required.
10. **Standardized Tests:** GMAT or GRE scores are required for the joint program. International applicants may need to submit TOEFL iBT or IELTS scores.

For more detailed information, you can visit the following URLs:

- [How to Apply - DSI](#)
- [FAQs - DSI](#)

👤: Can you summarize it?

🤖: The admission requirements for the MS in Applied Data Science program at the University of Chicago include completing an online application form, submitting unofficial transcripts from all previous colleges and universities (official transcripts are required if admitted), and providing two letters of recommendation (at least one from a professional supervisor). Applicants must also submit a candidate statement (up to 1,000 words), a detailed resume/CV, and a programming supplement demonstrating skills in R or Python. Additionally, short recorded video responses are required for a virtual portfolio. Non-native English speakers must provide proof of English proficiency with minimum TOEFL or IELTS scores. There is a non-refundable application fee of \$90, and GMAT or GRE scores are required for the joint MBA/MS program.


For more detailed information, you can visit the following URLs:


- [How to Apply - DSI](#)
- [FAQs - DSI](#)


Evaluation - Hallucination





AI Assistant for the UChicago MS-ADS Program

 Ask any questions about the MS-ADS Program

: What is the tuition cost for Northwestern University?

: I'm sorry, I don't have enough information to answer that question.

: How do I apply to MBA program at Harvard University?

: I'm sorry, I don't have enough information to answer that question.

Enter your question here:

Evaluation - Routing + RAG Fusion



```
# Get AI response with conversation history
response = chatbot(user_input, st.session_state.vectordb, st.session_state.chain, st.session_state.llm_chat_history, routing=False, fusion=True)
```

4 options here
routing=True, fusion=True | routing=True, fusion=False | routing=False, fusion=True | routing=False, fusion=False

Expected Answer (Provided by Uktu)		Baseline	Routing(inc. Indexing)
Can you provide information about the capstone project?		The Capstone Project in the University of Chicago's MS in Applied Data Science program is a culminating experience where students apply their knowledge and skills to real-world business problems. The project spans two quarters and involves research design, implementation, and writing. Full-time students typically start their capstone project in their third quarter, while part-time students begin in their fifth quarter. Students work in teams of four, guided by an instructor and a subject matter expert, to develop data science solutions that address organizational problems, enhance analytics capabilities, and expand talent pools and employment opportunities. The projects are sponsored by partner organizations that provide a detailed problem statement, data access, and regular engagement through meetings. Some examples of past capstone projects include real-time credit card fraud detection, COPD readmission and cost reduction assessment, NFL ticket pricing optimization, yoga pose identification using image recognition, and measuring the speed of a baseball pitch using image recognition algorithms. For more information, you can visit the following URLs: - Course Progressions - Capstone Projects	The Capstone Project in the University of Chicago's MS in Applied Data Science program is a culminating experience where students apply their knowledge and skills to real-world business problems. Students work in teams, guided by an instructor and a subject matter expert, to develop data science solutions that address organizational challenges, enhance analytics capabilities, and expand talent pools and employment opportunities. The project involves collaboration with sponsors who provide a detailed problem statement, data access, and regular engagement through meetings. The Capstone Project is completed over two quarters, covering research design, implementation, and writing. Full-time students typically start their capstone project in their third quarter, while part-time students begin in their fifth quarter. The program accepts projects year-round for placement during the Spring and Autumn quarters, with the Spring quarter being the largest cohort. Selected past projects include real-time credit card fraud detection, COPD readmission and cost reduction assessment, an NFL ticket pricing study, using image recognition to identify yoga poses, and measuring the speed of a baseball pitch using image recognition algorithms. For more information, you can visit the Capstone Projects page: Capstone Projects - DSI.
	The capstone project is a key component of the MS in Applied Data Science program, where students work on real-world problems, applying their learned skills to develop data-driven solutions.		

Fusion	Routing + Fusion
The Capstone Project in the University of Chicago's MS in Applied Data Science program is a culminating experience where students apply their knowledge and skills to real-world business problems. The project is completed over two quarters and involves research design, implementation, and writing. Full-time students typically start their capstone project in their third quarter, while part-time students begin in their fifth quarter. Students work in teams, guided by an instructor and a subject matter expert, to develop data science solutions that address organizational problems, enhance analytics capabilities, and expand talent pools and employment opportunities. The projects are sponsored by partner organizations that provide real-world problems, data, and expected results. These sponsors benefit from the insights generated by the students and often use the project as a recruitment opportunity. Examples of past capstone projects include real-time credit card fraud detection, COPD readmission and cost reduction assessment, an NFL ticket pricing study, and using image recognition to identify yoga poses or measure the speed of a baseball pitch. For more information, you can visit the Capstone Projects page: Capstone Projects - DSI.	The Capstone Project is the culminating experience in the University of Chicago's Master's in Applied Data Science program. It allows students to apply their knowledge and skills to real-world business problems. Students work with project sponsors to develop data science solutions that address organizational challenges, enhance analytics capabilities, and expand talent pools and employment opportunities. The project involves research design, implementation, and writing, and is completed over two quarters. Full-time students typically start their capstone project in their third quarter, while part-time students begin in their fifth quarter. Capstone projects can be research-focused, leveraging the university's rich research portfolio. Some examples of past capstone projects include: - Real-Time Credit Card Fraud Detection: Students created synthetic data representing a large population of credit card users and built a model to catch fraud in real time. - COPD Readmission and Cost Reduction Assessment: Students built data models to rank-order readmission risk, allowing for interventions to prevent readmission. - An NFL Ticket Pricing Study: Students developed strategies for an NFL team to implement variable and dynamic pricing methods to optimize revenue. - Using Image Recognition to Identify Yoga Poses: Students built an app using a neural network to recognize yoga poses and provide feedback. - Using Image Recognition to Measure the Speed of a Pitch: Students developed an app to measure the speed of a pitched baseball using image recognition algorithms. Capstone sponsors are expected to provide a detailed problem statement, access to relevant data, and regular engagement through meetings. The program accepts projects year-round for placement during the Spring and Autumn quarters. For more information, you can visit the Capstone Projects page: Capstone Projects - DSI.

Summary of Key Differences:

- **Baseline** focuses on general project details and student benefits.
- **Routing** emphasizes program logistics (timing of project acceptance and larger cohorts).
- **Fusion** highlights sponsor benefits and the potential for recruitment.
- **Routing + Fusion** combines all aspects, adding a research-focused angle and a broader appeal to both students and sponsors.

Summary

- This RAG model architecture answers user queries about the MS-ADS website by combining several tools and techniques:
 - 1. User Query: Routed with metadata and enhanced with query rewriting via OpenAI models.
 - 2. Vector Database (FAISS): Central storage for embedded website content, processed through **fetching, cleaning, indexing, chunking, and embedding with tools like BeautifulSoup and Trafilatura.
 - 3. Prompt & Answer Generation: Retrieved results are re-ranked (RAG Fusion) for relevance and compiled into responses that include original links to avoid hallucinations, with conversation history maintained for context.
 - 4. Key Tools: LangChain manages operations, FAISS handles vector storage, and OpenAI models support embedding, routing, rewriting, and answer generation.
- This setup ensures accurate, context-aware responses, grounded in the MS-ADS website data.

Future Work




- **Include both hyperlink text and URL in retrieve context:**
 - Ideal result: "For questions regarding an application fee waiver, you can refer to the Physical Sciences Division fee waiver policy <https://physicalsciences.uchicago.edu/academics/admissions/application-requirements/#FeeWaiver>. (Source: <https://datascience.uchicago.edu/education/masters-programs/ms-in-applied-data-science/how-to-apply/>)"
- **Expanded Metadata and Contextual Cues:** Include additional metadata fields (e.g., content freshness, user intent) to enhance filtering accuracy.
- **Automated Evaluation Metrics:** Develop automated relevance scoring metrics to evaluate and monitor response quality over time.
- **User Personalization:** Tailor responses based on user history or preferences to enhance engagement and relevance.
- **Upgrade or Fine-tune to More Advanced Embedding Models:** Experiment with newer or more specialized embedding models as they become available to improve retrieval precision.


Appendix


Appendix



AI Assistant for the UChicago MS-ADS Program

 Ask any questions about the MS-ADS Program


 : How many courses must you complete to earn UChicago's Master's in Applied Data Science?


 : To earn UChicago's Master's in Applied Data Science, you must successfully complete 12 courses, which include 6 core courses, 4 elective courses, and 2 Capstone courses, along with a tailored Career Seminar.

Relevant URLs:

- [In-Person Program](#)
- [Online Program](#)

Enter your question here:


 : Where can I mail my official transcripts?


 : If your institution cannot send your documents electronically, you should mail your official transcripts to the following address:

The University of Chicago
Attention: MS in Applied Data Science Admissions
455 N Cityfront Plaza Dr., Suite 950
Chicago, Illinois 60611

For more information, you can visit the following URLs:

- [FAQs – DSI](#)
- [How to Apply – DSI](#)

 : What are the admission requirements for the MS in Applied Data Science program?

 : The admission requirements for the MS in Applied Data Science program at the University of Chicago include:

1. **Application Form:** Complete the online application form.
2. **Transcripts:** Submit unofficial transcripts from all previous colleges and universities attended. Official transcripts are required if admitted.
3. **Letters of Recommendation:** Two letters of recommendation are required. At least one should be from a direct manager/supervisor or internship supervisor who can attest to skills demonstrated in a professional setting. Letters from family members, friends, or peers are not accepted.
4. **Candidate Statement:** A candidate statement is required, which should not exceed 1,000 words. It should clearly present your aims and qualifications without restating your resume.
5. **Resume/CV:** A detailed resume including work experience, major accomplishments, awards, educational background, and professional conferences attended.
6. **Programming Supplement:** A PDF demonstrating current programming language skills, preferably in R or Python, is required. The document should be no more than two pages.
7. **Virtual Portfolio:** Short, recorded video responses to given prompts within the application. One video is required, and another is optional, each approximately 1:30 minutes in length.
8. **English Language Proficiency:** For non-native English speakers, proof of English language proficiency is required. Minimum scores are TOEFL 102 or IELTS 7.
9. **Application Fee:** A \$90 non-refundable application fee is required.
10. **Standardized Tests:** GMAT or GRE scores are required for the joint program. International applicants may need to submit TOEFL iBT or IELTS scores.

For more detailed information, you can visit the following URLs:

- [How to Apply – DSI](#)
- [FAQs – DSI](#)

Appendix



AI Assistant for the UChicago MS-ADS Program

💡 Ask any questions about the MS-ADS Program

👤: What are the deadlines for the in-person program?

🗓️: The deadlines for the in-person program of the MS in Applied Data Science at the University of Chicago are as follows:

- November 7, 2024 – Priority Application Deadline
- December 4, 2024 – Scholarship Priority Deadline
- January 21, 2025 – International Application Deadline (requiring visa sponsorship from UChicago)
- March 4, 2025 – Second Priority Application Deadline
- May 6, 2025 – Third Priority Application Deadline
- June 23, 2025 – Final Application Deadline

In-person application decisions are released approximately 1 to 2 months after each respective deadline.

For more information, you can visit the following URL: [Events & Deadlines – DSI](#)

Enter your question here:

AI Assistant for the UChicago MS-ADS Program

💡 Ask any questions about the MS-ADS Program

👤: What are the minimum scores for the TOEFL and IELTS English Language Requirement?

🗓️: The minimum required score for the TOEFL is 102 overall, and for the IELTS, it is an overall score of 7. Note that students are required to take the Academic Reading/Writing test within IELTS, not the General Training Reading/Writing test. TOEFL or IELTS score reports are valid for two years and scores will be considered expired if the test was taken more than two years prior to submission of the application.

For more information, you can visit the following URLs:

- [FAQs – DSI](#)
- [How to Apply – DSI](#)

Enter your question here:

AI Assistant for the UChicago MS-ADS Program

💡 Ask any questions about the MS-ADS Program

👤: Can you provide information about the capstone project?

🎓: The Capstone Project is a culminating experience in the University of Chicago's MS in Applied Data Science program. It allows students to apply their knowledge and skills to real business problems, gaining valuable, data-driven insights using authentic data. Students work with project sponsors to develop data science solutions that address organizational problems, enhance analytics capabilities, and expand talent pools and employment opportunities. The project is completed over two quarters and covers research design, implementation, and writing. Full-time students typically start their capstone project in their third quarter, while part-time students generally begin in their fifth quarter.

Capstone projects can include a variety of topics, such as:

- Real-Time Credit Card Fraud Detection: Students created synthetic data representing a large population of credit card users and built a model to catch credit card fraud in real time.
- COPD Readmission and Cost Reduction Assessment: Students built data models to rank-order readmission risk, allowing for interventions to prevent readmission.
- An NFL Ticket Pricing Study: Students strategized ticket pricing for an NFL team using variable and dynamic pricing methods.
- Using Image Recognition to Identify Yoga Poses: Students developed an app using a neural network to recognize yoga poses and provide feedback.
- Using Image Recognition to Measure the Speed of a Pitch: Students created an app to measure the speed of a pitched baseball using image recognition algorithms.

Capstone sponsors are expected to provide a detailed problem statement, access to relevant data, and regular engagement through meetings. The program accepts projects year-round for placement during the Spring and Autumn quarters.

For more information, you can visit the Capstone Projects page: [Capstone Projects – DSI](#).

👤: Can you summarize your answer?

🎓: The Capstone Project in the University of Chicago's MS in Applied Data Science program is a two-quarter culminating experience where students apply their skills to real business problems. They work with sponsors to develop data science solutions, covering research design, implementation, and writing. Full-time students typically start in their third quarter, while part-time students begin in their fifth. Projects can range from credit card fraud detection to image recognition applications. Sponsors provide problem statements, data access, and regular engagement. Projects are accepted year-round for Spring and Autumn placements. For more details, visit the Capstone Projects page: [Capstone Projects – DSI](#).

Appendix



System Design - System Overview and Data Flow

- **System Overview:**

- Integrates several key components: data preprocessing, embedding and vector storage, query handling, response generation, and user interface.
- Components interact seamlessly to create an AI-powered information retrieval chatbot focused on UChicago's MS in Applied Data Science program.

- **Data Flow:**

1. User Query Input:
 - User submits a question through the Streamlit interface (main_streamlit.py).
2. Query Preprocessing:
 - The query is preprocessed, rewritten, and expanded using functions like `rewrite_queries` in `chat_utils.py` to improve retrieval.
3. Document Retrieval:
 - FAISS vector database (`vector_search.py`) performs similarity-based retrieval to find relevant document chunks.
4. Response Generation:
 - Retrieved chunks and user query are passed to the GPT-4 model (`create_chat_chain` in `chat_utils.py`) to generate a final response.
5. Response Display:
 - The response is displayed in the Streamlit interface, with conversational history maintained for continuity.

Appendix



System Design - Core Components and Integration

• 1. Data Preprocessing and Vectorization:

- Script: `vectordb_creation.py`
- Key Functions:
 - `fetch_web_content` and `extract_metadata` for retrieving and processing data from UChicago's website.
 - `create_document` and `split_document` to structure and chunk documents.
 - `create_and_save_vectordb` to generate and save the FAISS vector database with embeddings.
- Purpose: Prepares data for efficient retrieval by structuring documents, enhancing metadata, and creating embeddings stored in FAISS.

• 2. Vector Database for Document Storage and Retrieval:

- Library: FAISS (Facebook AI Similarity Search)
- Script: `vector_search.py`
- Functions:
 - `load_vectordb` loads the vector database with precomputed embeddings.
 - `search_similar_chunks` retrieves document chunks similar to the user query based on vector similarity.
- Integration:
 - The vector database is structured to support fast similarity searches and metadata filtering, enabling targeted and efficient retrieval.

• 3. Language Model Chain for Response Generation:

- Model: GPT-4, via OpenAI's API
- Script: `chat_utils.py`
- Functions:
 - `create_chat_chain` initializes the model chain, with a structured prompt to integrate retrieved document chunks and query.
 - `rewrite_queries` and `subcategory_finder` improve the relevance of retrieved information.
- Integration:
 - The language model chain takes the retrieved chunks and user question, using the context to generate a coherent and contextually accurate response.

Appendix



System Design - User Interaction and Additional Functionalities

• 1. User Interface (Streamlit):

- Script: `main_streamlit.py`
- Purpose: Provides a simple and interactive UI for users to ask questions and view responses.
- Key Elements:
 - Session State Management: `st.session_state` is used to track chat history (`chat_history`) and maintain the conversation flow (`llm_chat_history`).
 - Dynamic Input and Response Display:
 - User inputs are managed with `text_input`, and responses are displayed with a typing effect for a natural experience.
 - Toggle Mechanism for Input Reset:
 - Clears the input box after each interaction using `st.session_state.clear_input`, making it easier for users to enter new queries.

• 2. Conversational Memory Management:

- Class: `ChatHistory` in `chat_utils.py`
- Functionality:
 - Maintains the conversation history, enabling context to persist across multiple turns.
 - `add_interaction` and `get_history` methods allow tracking of user queries and responses, which are integrated into subsequent interactions.
- Purpose: Ensures a seamless user experience by preserving context across multiple exchanges.

• 3. Result Ranking and Filtering:

- Reciprocal Rank Fusion:
 - Function: `reciprocal_rank_fusion` in `chat_utils.py`
 - Merges results from multiple rephrased queries, ranking documents based on their cumulative relevance.
- Metadata-Based Filtering:
 - Function: `search_similar_chunks` in `vector_search.py` with `filter_dict` support.
 - Filters results based on metadata fields (e.g., `primary_category`, `subcategory`) to improve the precision of retrievals.
- Purpose: Enhances response quality by prioritizing the most relevant content through ranking and filtering mechanisms.