ULB SPECIALIZED MASTER IN DATA SCIENCE, BIG DATA (MS-BGDA)

Building a RAG Pipeline with MLOps Backend to Support Research in Ecology

Alexandra Sébastien
Arnaud Vanwelsenaere
Boris Zok

Project developed as part of the course:

Current Trends in Artificial Intelligence

INFO-H512

Prof. Hugues BERSINI, Johan LOECKX

Project related to the presentation "LLMs and RAG", by Lluc Bono, May 16th, 2025

June 2025 Academic Year 2024-2025

1. Introduction

Professor Gaël CARO, at the Laboratoire d'Agronomie et Environnement, Université de Lorraine, is currently developing a new synthetic indicator of biodiversity in France. His research group is collecting and scanning relevant naturalistic books to compile data on diverse species (i.e. morphology, range, behavior) into a comprehensive database.

As a first step in this research project, we developed a chatbot based on a Retrieval-Augmented Generation (RAG) pipeline with Machine Learning Operations (MLOps) enabled backend. We worked on extracting information from scanned naturalist old books using Optical Character Recognition (OCR), Large Language Models (LLMs), and RAG on the extracted text enriched with metadata.

2. Materials and methods

2.1. Data and metadata

We received four scanned PDF books published between 1940 and 1990, each ranging from 100 to 450 pages. For this project, we focused on one book in particular: British Spiders, Vol. II by G. H. Locket et al. (1953). Each scanned page corresponds directly to a printed page and contains both textual content and naturalistic illustrations. Although the structure of the book was analyzed, OCR extraction and subsequent LLM processing were hindered by printing artifacts. As a result, the taxonomic index listing families, genera, and species had to be manually reconstructed into an Excel reference file containing all relevant taxonomic information. This file was later used to enrich each OCR- and LLM-processed page with consistent metadata.
It is important to emphasize that, while our detailed analysis focused on this specific book, the pipeline and underlying methods developed are fully generalized to the remaining books, regardless of differences in structure or content.

2.2. Converting the books into a structured database

Our pipeline was built in Python and combined several AI tools to restore, annotate, and organize content from scanned PDF sources.

We started by converting the scanned PDF of our book into images, one per page. Text was extracted using OCR (Tesseract), while a custom image-processing module identified and extracted naturalistic illustrations using adaptive thresholding and contour detection. These images were saved in a structured format for later use. The raw OCR text was then cleaned using a local LLM (via Ollama with Mistral or Deepseek), which improved readability and added explanations in a target language - without changing the original meaning.

The system's flexible setup lets users enable or disable LLM processing and choose how to save results: either in a SQL Server database (including books, pages, images, and cleaned text) or as files on disk. This made it suitable for anything from simple offline use to integration with search tools.

Each page was processed with two LLM prompts: one for basic cleaning (fixing OCR errors while keeping the original structure) and one for adding explanations and annotations. These steps made the old documents easier to read and reuse, and opened possibilities for things like semantic search, indexing, or use in a specialized knowledge base.

2.3. Designing and developing a Windows-based RAG system using a C# application

We developed the application as a Windows Forms (C#) desktop program connected to a SQL Server database. The database schema handled books, pages, OCR output, cleaned text, LLM-generated explanations, and related images. We precomputed text embeddings using nomic-embed-text. Embeddings were stored at the chunk level, along with their L2 norms for fast cosine similarity comparisons. The individual vector values were also saved in a secondary table for more detailed queries. Finally, a stored procedure (SearchSimilarPages) was developed to rank results based on cosine similarity, using a provided embedding passed as a structured parameter (the user's question converted into an embedding).

The user interface supported document import, text format selection, image viewing, and manual browsing. A dedicated "Ask My Data" tab allowed users to ask natural language questions. When a query was submitted, the system computed its embedding, searched for the most relevant text chunks, and passed them as context to the selected LLM. Users could configure several options at runtime: LLM model (e.g., Mistral, Mixtral, Deepseek), number of retrieved chunks (Top N), response language, token limit, and whether to restrict the search to a specific document. Answers were streamed in real time from the LLM, along with any related figures or diagrams from the source pages.

We paid particular attention to handling multi-turn interactions, prompt size limits, and multilingual output. To improve coherence and efficiency, we dynamically built a contextual header using cleaned and summarized text from the selected chunks. All processing - from text extraction to embedding and search - could be done entirely offline, preserving user privacy and supporting local deployment.

3. Pipeline evaluation

We aimed to develop a pipeline that is both reliable and trustworthy. However, we had to also consider constraints related to computational capacity and financial resources. Therefore, our objective was to strike an optimal balance between quality and cost-effectiveness.
To this end, we conducted two types of evaluations:

- Computational Evaluation: To ensure that the pipeline can operate efficiently on standard computing hardware.
- Qualitative Evaluation: To verify that the pipeline produces high-quality and meaningful results.

3.1. Computational evaluation

We built our pipeline on a computer with the following specifications:

Parameters	Values
RAM	16 GB
CPU	Intel I7-14700F
GPU	NVidia RTX 4060
Minimum disk space	50 GB free (for our complete pipeline)
Approximate price on market	500€ - 1000€

This configuration is common on the market and has approximately the same specifications as the one used by Professor Gaël CARO.

Then, we measure the time taken by each process of our pipeline for a page of data (text and image):

Module	Description	Evaluation
Ingestion	Process a page: - Read characters via OCR. - Extract images from page - Cleaning with LLM	17 seconds
Embedding	For a page: - Creation of 6 x 200-word embedding vector. - Size of a 6 x 200-word embedding vector on the disk	- 2.8 seconds - 48-96 kB
Retrieval	- Retrieve relevant chunks in database of 1 page of information. - The retrieve time is linear so for a 1000 pages size (we will have 4 minutes)	- 270 milliseconds - 4 minutes 30 seconds

3.2. Quality evaluation

To make sure our pipeline is reliable and does not hallucinate, we created a series of 40 questions on the book and performed fact checking and evaluated the quality and conciseness of each response. A subset of these questions is provided in the annexes.

Category	Score		Comments
	Normal	With metadata addition	
Fact-checking	7,9/10	9,25/10	In general, the pipeline provides good results. But the metadata addition helps provide better results.
Quality and conciseness	8,7/10	9,5/10	Despite the computational limitations that necessitate the use of a relatively small language model (Mistral 7B), the quality and the conciseness of our generated response are still very good. Also, when we add the metadata, the general quality increases, which could be explained by better chunks, with less noise.

Discussion

We managed to create an efficient OCR extraction, LLM-based text cleaning, and RAG pipeline for a chatbot with an MLOps-enabled backend. Our pipeline can be easily generalized to other types of documents. When we introduced metadata into our extracted, cleaned text - after OCR and LLM treatment - we observed an improvement in structuring and retrieval.

We believe that for future books, it will be necessary to further develop the taxonomic reference Excel document to ensure correct metadata is applied to each processed page.

Additionally, it is possible to modify the chunking logic. It may be valuable to split the text by species, making chunks more relevant to the queries. However, this could lead to very uneven chunk sizes, as different books treat species with varying levels of detail. Therefore, a balance between semantic chunking and metadata tagging may significantly improve response precision and relevance. This follows the principle that specific problems require specific solutions - in our case, a tailored chatbot pipeline.

That said, the system was designed from the outset to be adaptable and functional beyond scientific documents, including business reports or invoice processing. What makes it truly robust and well-suited to the biodiversity domain, however, is the automated matching between each page’s text and its corresponding taxonomy. This taxonomic context is linked to the page and stored in a dedicated embedding, which enhances retrieval accuracy and domain relevance.

An alternative approach would be to use external APIs. These services allow access to highly optimized OCR engines (e.g., Google Vision), powerful LLMs (e.g., OpenAI GPT-4), and efficient RAG frameworks (e.g., Azure Cognitive Search or LangChain). However, as mentioned in our introduction, these solutions involve costs. Ecology laboratories - and universities more broadly - often operate with limited research funding. These paid solutions, while faster and more robust, may therefore be out of reach. They also raise concerns about data sovereignty and long-term accessibility of research workflows. In this context, we believe our work represents a solid effort to support Professor Gaël CARO in his research.

Conclusion

Our goal was to extract and process information from scanned naturalist books using OCR, LLM-based text cleaning, illustration extraction, and Retrieval-Augmented Generation (RAG) to enable semantic search and reasoning. By building a searchable embedding index and integrating local inference through models such as Mistral and Mixtral (deployed via Ollama), we developed a fully functional platform for the interactive exploration of scientific biodiversity documents — while preserving data sovereignty and fostering independent work. We will now engage further with Professor Gaël Caro and his laboratory to identify the next steps and evolving needs of the project.

Annexes

1. Qualitative evaluation form

Questions	**Expected answer**	**LLM answer “Mistral-7B” (With metadata)**	**Fact checking**	**Quality and conciseness**
Who are the authors of British Spiders Book?	G. H. LOCKET, M.A. (OXON.), M.Sc. (LOND.), F.Z.S. & A. F. MILLIDGE, Ph.D., B.Sc. (LOND.)	The authors of the book "British Spiders" are G. H. Lockett and A. F. Millidge.	True = 1	Good = 1
Which species families are presented in the volume of British spiders book?	AGELENIDAE, MIMETIDAE, THERIDIIDAE, NESTICIDAE, TETRAGNATHIDAE, ARGIOPIDAE and LINYPHIIDAE	The families presented in this volume of British spiders are AGELENIDAE, MIMETIDAE, THERIDIIDAE, NESTICIDAE, TETRAGNATHIDAE, ARGIOPIDAE and LINYPHIIDAE.	True = 1	Good = 1
What is the shape of the carapace of the AGELENIDAE family?	Oval, attenuated in front with a longitudinal fovea	The carapace (exoskeleton) of the AGELENIDAE family is oval and attenuated (tapering) in front, except for the species Ama1trobiu8 and Oicurea, where it is not mentioned if they have this same shape. This shape also features a longitudinal fovea or groove, which houses the genalia larva.	True = 1	Good = 1
Do the eye of AGELENIDAE family vary in size?	Not very different in size (except in Cryphoeca, Tetrilus, Tuberta and Hahnia)	Yes, the eyes of the Agelenidae (Funnel-web spiders) family can vary in size among different species within this family. This is a common characteristic across many families of spiders. However, it's important to note that specific information about the size and variation of their eyes for each species would require consulting scientific literature or resources dedicated to the study of these particular species.	Somewhat True = 0.5 => didn’t reply and invite the user to an external source.	Fairly Good = 0,5
What is the shape of the carapace of the Genus ARGYRONETA?	Cephalic area long and some- what attenuated. Fovea not distinct, represented by a longitudinal narrow dark streak. Faint radiating strire visible.	The carapace of the Genus ARGYRONETA is long and somewhat attenuated.	True = 1	Good = 1

Describe the eyes of the Genus ARGYRONETA?	Anterior row slightly recurved; medians the smallest, O· 5 diameter apart and 1·5 diameters from laterals. Posterior row slightly recurved, eyes almost equidistant.	The Genus Argyroneta has eyes that are characterized by a slightly recurved anterior row, with medians being the smallest and located0.5 diameter apart from each other and1.5 diameters away from the laterals. The posterior row is slightly recurved as well, and the eyes are almost equidistant. The cephalic area of the carapace is long and somewhat attenuated, with a faint dark streak representing a non-distinct fovea and faint radiating stripes visible.	True = 1	Good = 1
Describe the abdomen of the Genus ARGYRONETA?	Thickly clothed with rather short hairs. Tracheal spiracles situated far from spinners (indicated by a fold just behind the epigastric fold	The abdomen of the genus Argyroneta is thickly clothed with relatively short hairs. It does not have a distinct fovea, instead it has a longitudinal narrow dark streak to represent it. Faint radiating stripes are visible on the carapace. The stridulating organ is usually not well developed and is indistinct. The coloration varies but can be yellowish-brown with no very definite markings except for the faint radiating stripe. In some cases, the abdomen may appear purplish violet to almost black with light spots that also vary and can be nearly absent.	True = 1	Good = 1
Describe the spinners of the Genus ARGYRONETA?	Anteriors conical, placed close together Posteriors about the same length, but rather slenderer	The spinners of the Genus ARGYRONETA are described as conical in shape, placed close together. They are located on the abdomen of these spiders. In addition, the anterior spinners are slightly recurved while the posterior ones are about the same length but a bit slenderer.	True = 1	Good = 1
How big is the Argyroneta aquatica?	female: about 8-15 mm. male: about 9-12 mm. (Greater variations are likely to occur, specimens from different localities show considerable differences in size.)	The Argyroneta aquatica typically measures about 8-15 mm in length, with the abdomen measuring about 9-12 mm. However, variations are likely to occur, and specimens from different localities may show considerable differences in size.	Half true and half false = 0,25 (female = 8-15mm / male = 9-12mm)	Good = 1
Describe the carapace of the Argyroneta aquatica?	Light yellow brown with faint darker stripe; with only a few hairs, mostly, along stripe in cephalic area.	The carapace of the Argyroneta aquatica is light yellow brown with faint darker stripes, and it has only a few hairs, mostly along the stripe in the cephalic area. It is long and somewhat attenuated, with a fovea not distinct, represented by a longitudinal narrow dark streak. Faint radiating stripes are visible on the carapace.	True = 1	Good = 1