Metadata Schema Reconciliation for Multimodal Bioinformatics at AUMC

Submitted on: **08-03-2024**

Han Zhang
han.zhang4@student.uva.nl
University of Amsterdam
Amsterdam, The Netherlands

Rodrigo Roas-Bertolini
r.rosas@amsterdamumc.nl
University of Amsterdam
Amsterdam, The Netherlands

1 METHODOLOGY

1.1 Database Selection

To implement the prototype, the first step involves selecting a database, and the Image Data Resource (IDR) has been chosen. IDR stands out as a public repository for image data that gathers datasets from various published scientific research, enabling the community to submit, search, and access high-quality biological images[1]. Not only IDR hosts high-quality biological images but also integrates them with associated tabular datasets such as Genomics and Proteomics, including links to related scientific literature, making it a rich, interdisciplinary resource.

One of IDR's key strengths is its robust metadata annotation, which is fully searchable and accessible through via the OMERO API, facilitating seamless integration with various analytical tools and platforms. Moreover, IDR is engineered to efficiently manage and process large volumes of data, thereby facilitating rapidly iterate through subsequent steps of the project. The database's support for new submissions means it is continually refreshed with the latest experimental findings, keeping the repository up-to-date.

1.2 Exploratory Data Analysis of IDR Database

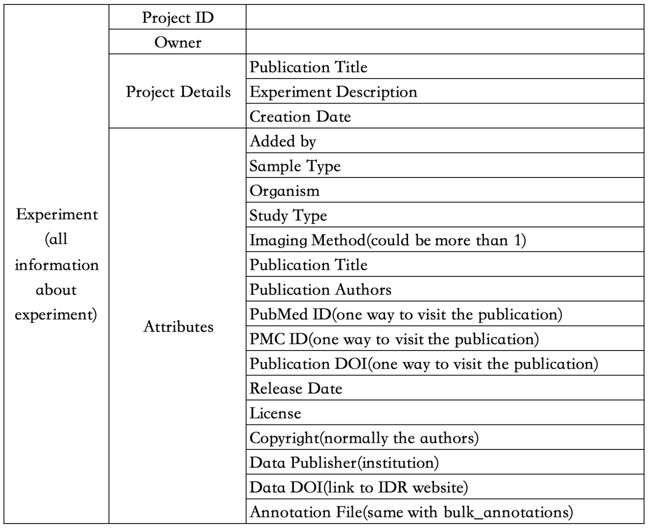
As for now, the IDR database contains 14,017,840 high-quality biomedical images from 127 studies, totaling 385 TB of data. The IDR database primarily includes two major categories of images: cell images and tissue images. Among all the publicly accessible data, there are two main types of datasets. One is the screen dataset, which involves expansive and exploratory large-scale data collection. The other type is the experiment dataset, which is specifically designed for precise control and detailed observation based on specific hypotheses. This paper will focus on the experiment dataset. Because the experiment dataset has fewer images and a simpler, clearer structure, it helps in building models for chatbots.

The experiment database contains a total of 124 datasets, primarily using the OME-TIFF standard as the metadata format. OME-TIFF, which stands for Open Microscopy Environment Tagged Image File Format, is a file format specifically designed for microscopy images. OME-TIFF is commonly used in conjunction with OME-XML [2]. While OME-TIFF provides the actual storage for image data, OME-XML is responsible for describing the detailed metadata of these images. OME-XML (Open Microscopy Environment XML) is an XML-based format that can store comprehensive information from experimental setups to image acquisition details. It also supports the extension of metadata content through the addition of custom tags.

The structure of the experiment database is hierarchical, beginning at the project level. Each project is an independent entity comprised of multiple datasets, and each dataset in turn contains numerous images. The metadata structure follows this hierarchy, where attributes are inherited from each level. For projects, the key attributes include the project ID, a description of the project, and information on related publications. For datasets, the main attributes are the dataset ID and a description of the dataset. The attributes of images can vary depending on the specifics of the project but generally encompass two main categories:

- Technical Details: This includes information such as image dimensions, pixel type, magnification, and other technical specifications.
- Biological and Clinical Relevance: This includes data on biological characteristics, the source of the tissue, and biomedical parameters that are pertinent to the image.

The following example illustrates the metadata associated with a project and its images.



Experiment (all information about experiment)	Project ID	
	Owner	
	Project Details	Publication Title
		Experiment Description
		Creation Date
	Attributes	Added by
		Sample Type
		Organism
		Study Type
		Imaging Method(could be more than 1)
		Publication Title
		Publication Authors
		PubMed ID(one way to visit the publication)
		PMC ID(one way to visit the publication)
		Publication DOI(one way to visit the publication)
		Release Date
		License
		Copyright(normally the authors)
		Data Publisher(institution)
		Data DOI(link to IDR website)
		Annotation File(same with bulk_annotations)

Figure 1: Project Metadata

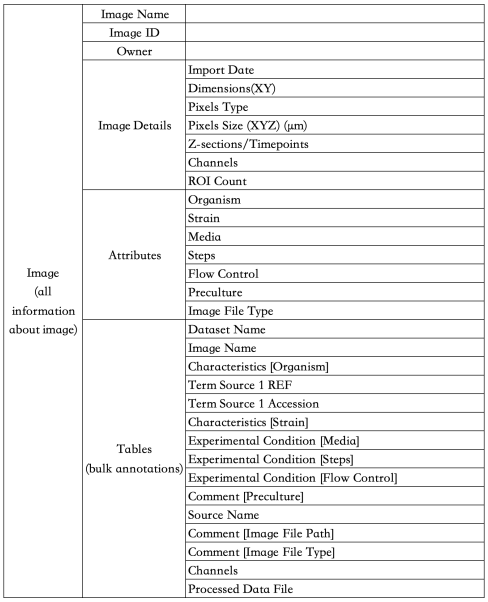


Image (all information about image)	Image Name	
	Image ID	
	Owner	
	Image Details	Import Date
		Dimensions(XY)
		Pixels Type
		Pixels Size (XYZ) (µm)
		Z-sections/Timepoints
		Channels
		ROI Count
	Attributes	Organism
		Strain
		Media
		Steps
		Flow Control
		Preculture
		Image File Type
	Tables (bulk annotations)	Dataset Name
		Image Name
		Characteristics [Organism]
		Term Source 1 REF
		Term Source 1 Accession
		Characteristics [Strain]
		Experimental Condition [Media]
		Experimental Condition [Steps]
		Experimental Condition [Flow Control]
		Comment [Preculture]
		Source Name
		Comment [Image File Path]
		Comment [Image File Type]
		Channels
		Processed Data File

Figure 2: Image Metadata

1.3 System Overview and Implementation Steps

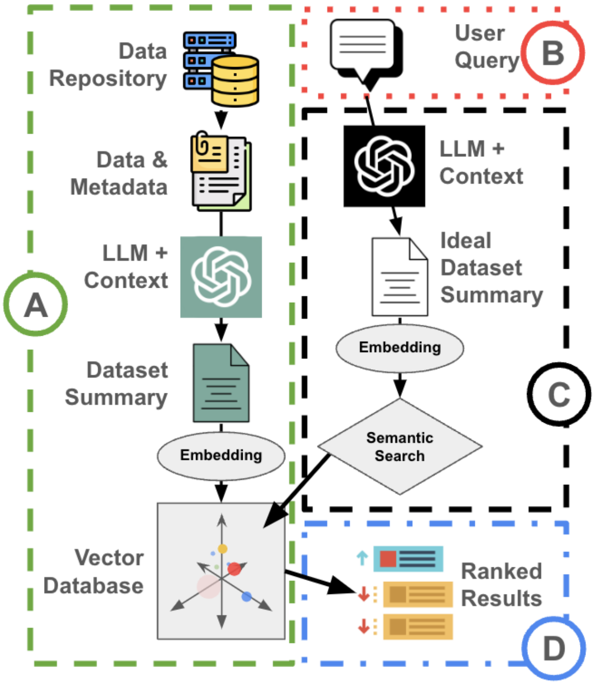


Figure 3: MetaSummarizer

1.3.1 System Rationale. The above figure3 shows the rationale of this information retrieval system. This component focuses on extracting metadata from IDR database, summarizing this information using a LLM(e.g.ChatGPT), and then vectorizing these summaries for efficient storage and retrieval. This process can be broke down into 4 steps:A, B, C and D

- A) Metadata is extracted from dataset of the chosen repository. This metadata of the dataset, along with the publications of the whole project, is processed using an LLM to generate concise summaries. These summaries are then embedded into vector representations for upcoming retrieval.
- B) Intake of user queries.
- C) User queries undergo a similar process, where they are contextualized using LLM, summarized into an ideal dataset summary. Then, the ideal summary is also converted through embedding and in this form it is compared to all embedded summaries created in part A.
- D) The search results are then ranked, and presented as results to the user query.

1.3.2 Methodology Merits. Compared to traditional information retrieval systems, such as the native search functionality of the IDR, the chatbot system incorporates RAG and LLM technologies, showcasing superior performance. This is evidenced in several ways: the chatbot utilizes abstracts and methodologies from project-related publications as external sources of knowledge to supplement the LLM's internal representation of information. Therefore, users can perform searches using natural language, rather than specific dataset attributes as provided by the IDR system. Moreover, the chatbot also employs a language model to understand the deep intent and context of queries, thus enhancing the quality and relevance of the retrieval. Traditional search systems do not offer an interactive user experience, such as adjusting responses based on user feedback during a conversation. The chatbot model uses LangChain to equip the model with conversational memory, offering search results through human-computer interaction that better meet user needs.

1.4 Implementation Steps

1.4.1 Dataset Partitioning. The dataset is split approximately into 70% for training, 15% for validation, and 15% for testing. Given that there are 124 projects in total in the experiment, 90 projects are selected for the training set, 17 projects for the validation set, and another 17 for the test set. The method of sampling is random.

Although there is a significant variation in the memory size of each project, the model generates summaries based on the publications related to the project and the metadata contained within each project according to a set of agreed rules. Since we cannot predict the frequency with which users will search different datasets, the variation in project memory sizes will not impact the performance of the final retrieval system.

1.4.2 Create the natural language summaries. ChatGPT API key will be the used for generating summaries of datasets. This approach involves leveraging the GPT-3.5 Turbo models which utilize deep learning to understand and generate natural language text . Unlike Rule-Based or Abstraction-Based methods which rely on specified algorithms or extracting and rephrasing key sentences from the text, GPT-based summarization does not rely on predefined rules or simply rephrasing existing sentences. Instead, it uses the context and semantic understanding capabilities of GPT models to produce coherent and relevant summaries directly from the metadata, including data types, research methods, and sample sizes[**?**]. The next phase involves evaluating the quality of the GPT-generated summaries in terms of accuracy and completeness. Following the selection of the GPT model for summary generation, the focus shifts to the vectorization of these summaries. This process starts with text cleaning and preprocessing, which involves eliminating unnecessary symbols and stop words, as well as applying stemming techniques to refine the text data. The cleaned summaries are then transformed into vector representations using the TF-IDF method, facilitating further analysis and application.

1.4.3 User Query Analysis. The next step involves the collection of potential queries, a task that presents notable challenges. One possible solution is extracting research questions from academic papers associated with the datasets. These questions, derived from publications linked to the datasets, will be utilized as representative queries. Then an evaluation will be conducted to test if the research questions can be used to recover the corresponding dataset. Once the queries are preparedthey will be subjected to a process similar to the embedding of dataset metadata. The objective is to extracted accurate entity information from queries, analyze the intent behind queries and understand the semantics, enabling the generation or retrieval of appropriate responses. One of the challenges in this process is maintaining dialogue coherence, which involves managing and remembering the context of the conversation, including information mentioned by the user and responses from the information retrieval system, to ensure that subsequent interactions can build on the content of previous conversations, thus maintaining a logical and coherent dialogue flow.

1.4.4 Develop information retrieval system. This step is a crucial integration of the previous stages, where the information retrieval system emerges as the primary interface through which users interact with the system. It functions as the visible layer where users can input their queries, the information retrieval system processes these queries and displays these ranked results in a clear and organized manner, with the most relevant datasets appearing at the top of the list, allowing users to quickly review and select the dataset that best meets their research needs or interests. It's important to balance the accuracy of results with response time in this process. Additionally, providing users with an intuitive and user-friendly interface is also important.

1.5 Validation of the System

This component allows users to query the system, generates an ideal summary based on the query (a synthetic summary), embeds this summary, and then performs vector comparison to find the best matches in the database created by Component.

A) This stage involves deriving research questions from literature related to specific data repositories. These questions are meant to act as substitutes for user queries that the system will later process. B) Here, the system receives the curated research questions as its input, simulating real user interactions. C) Upon receiving the questions, the system executes a targeted search operation, looking for the most relevant information in response to the queries. D) The validation step is crucial; it assesses whether the system's search algorithm is able to consistently and accurately recommend the original data repository among its top results. Success in this step would affirm the system's reliability and precision in retrieving information, thus validating its ability to adequately respond to actual user queries.

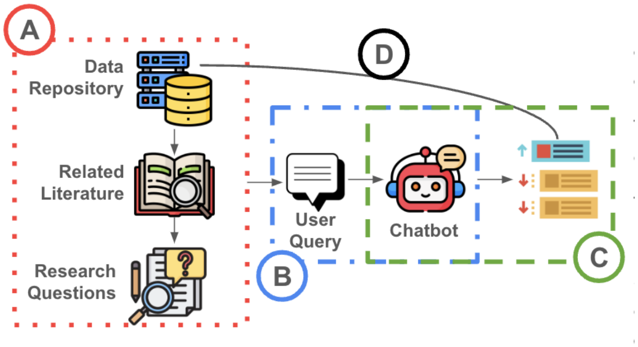


Figure 4: QueryVectorMatcher

REFERENCES

[1] IDR. [n. d.]. IDR. https://idr.openmicroscopy.org/. Accessed: 23-02-2024.
[2] OME-TIFF. [n. d.]. OME-TIFF. https://docs.openmicroscopy.org/ome-model/5.6.3/ome-tiff/. Accessed: 19-04-2024.