Federated Information Retrieval in Cross-Domain Information Systems

Sylvia Melzer[1,2], Hagen Peukert[3], Eliana Dal Sasso[2], Charles Li[2], Thomas Asselborn[1] and Ralf Möller[1]

[1] *University of Lübeck, Institute of Information Systems, Ratzeburger Allee 160, 23562 Lübeck, Germany*
[2] *Universität Hamburg, Centre for the Study of Manuscript Cultures, Warburgstraße 26, 20354 Hamburg, Germany*
[3] *Universität Hamburg, Centre for Sustainable Research Data Management, Monetastraße 4, 20146 Hamburg, Germany*

Abstract

In humanities research projects, scholars examine written artefacts, such as manuscripts, for various purposes based on factors like language, textual content, provenance, codicological aspects, and other characteristics. While humanities scholars can make statements about different aspects of self-contained artefacts based on their expertise, there are instances where the statements are made without numerical verification due to limited research data that are available within a project. If a variable, e.g. the size of a book, is requested, classic search engines provide similar but not precise answers. Our thesis proposes that by combining various cross-domain information sources as a federated database system, these missing variables can be supplemented, thereby validating research questions in the humanities. The article proposes a cross-domain information system that enables efficient federated search for comprehensive research in the humanities. The system combines diverse information sources and provides efficient search capabilities by demonstrating an efficient data matching approach called indexing. This article also presents how users can define their queries in natural language by integrating GPT4all to generate SQL queries from natural language queries. The achieved result is a cross-domain information system that facilitates comprehensive research in the humanities by combining diverse information sources and providing efficient federated information retrieval.

1. Introduction

Depending on the research interest, it may be that different researchers study the same manuscript with a different focus or apply the same research question to written artefacts pertaining to different manuscript traditions. Alongside traditional publications in journals and

Humanities-Centred AI (CHAI), 3rd Workshop at the 46th German Conference on Artificial Intelligence, September 26, 2023, Berlin, Germany

*Sylvia Melzer
sylvia.melzer@uni-hamburg.de (S. Melzer); hagen.peukert@uni-hamburg.de (H. Peukert); eliana.dal.sasso@uni-hamburg.de (E. Dal Sasso); charles.li@uni-hamburg.de (C. Li); asselborn@ifis.uni-luebeck.de (T. Asselborn); moeller@ifis.uni-luebeck.de (R. Möller)
https://www.csmc.uni-hamburg.de/about/people/melzer.html (S. Melzer); https://www.uni-hamburg.de (H. Peukert); https://www.csmc.uni-hamburg.de/about/people/dalsasso.html (E. Dal Sasso); https://www.uni-hamburg.de (C. Li); https://www.ifis.uni-luebeck.de/index.php?id=asselborn (T. Asselborn); https://www.ifis.uni-luebeck.de/~moeller/ (R. Möller)
0000-0002-0144-5429 (S. Melzer); 0000-0002-3228-316X (H. Peukert); 0000-0001-7205-3388 (E. Dal Sasso); 0000-0001-8247-9374 (C. Li); 0009-0005-3011-7626 (T. Asselborn); 0000-0002-1174-3323 (R. Möller)

CEUR Workshop Proceedings (CEUR-WS.org)

CEUR Workshop Proceedings ceur-ws.org ISSN 1613-0073

monographs, research data about written artefacts can be found independently in digital resources produced by research institutions, museums, or libraries. Increasing amounts of sources residing in libraries and archives are digitized and made accessible in an RDR (**R**esearch **D**ata **R**epository) such as Zenodo [1] or adjusted instances of it [2] like at the Universität Hamburg.

In the project *Beta maṣāḥəft* a collection of XML files, based on the TEI (**T**ext **E**ncoding **I**nitiative) Guidelines [3], were created that describe textual and physical features of manuscripts from Ethiopia and Eritrea. These TEI files had been published and are available online[1]. In this machine-readable format users usually do not have an overview of all written artefacts that have the same property.

Apart from that, project-specific web applications were built. In addition, users cannot perform natural language queries to obtain required data from XML documents. Additional tools are necessary to make XML data searchable. For this reason, we developed and used the generic DBoD (**D**ata**B**asing **o**n **D**emand) process [4] to transform research data from TEI files to a database instance[2], then an information system based on top of the database instance was created. "An information system is an integrated set of components for collecting, storing, and processing data and for providing information, knowledge, and digital products." [5]

In the project *Bookbindings as Instruments of Classification* at the Universität Hamburg a collection of JSON files were created to document the binding technique used in Egypt from the fourth to the twelfth centuries. JSON is also a machine-readable format, so we also created a database instance[3] using the DBoD process using JSON files as input and created an information system based on top of the database instance.

In the project *Text-Surrounding-Text* the research data about binding techniques in South India are stored directly in the National library of France [4].

While digital collections, like the three examples mentioned above, provide valuable data for studying bookbinding techniques, it is important to note that addressing certain research questions often necessitates the use of multiple information sources. For instance:

Are there any similarities between the binding techniques, the object size or written area dimension of manuscripts from Ethiopia, Eritrea, early Egypt, and South India?

Evaluating and retrieving information from diverse sources and domains, FIR (**F**ederated **I**nformation **R**etrieval) in cross-domain information systems is a research area that focuses on advancing the scholars of the humanities, both technically and methodologically, by integrating different sources of data and evaluating them based on various criteria such as accuracy, currency, and relevance. Access to the three different databases is realized through federated search. Federated search is a technique for searching multiple collections simultaneously with a single query. To make the search more efficient, we use the indexing method from Melzer et al. [6]. In the process, EpiDoc (**Epi**graphic **Doc**uments in TEI XML) files [7] (a customized version of TEI) were used as input. In this article, we use TEI and JSON instead of EpiDoc. As a result of the indexing process, we have a similarity score for each of the data sets, whereby only parts of

[1] https://github.com/BetaMasaheft/Manuscripts

[2] https://heurist.fdm.uni-hamburg.de/html/heurist/?db=CSMC_UWA_BETAMASAHEFT

[3] https://heurist.fdm.uni-hamburg.de/html/heurist/?db=CSMC_UWA_RFE09

[4] https://tst-project.github.io/mss/Sanscrit_1129.xml

the data, the so-called index candidates, are used for comparison so that the complexity of the calculation does not increase.

To define queries in natural language, we use the pre-trained transformer model GPT4all which generate SQL (**S**tructured **Q**uery **L**anguage) queries from natural language queries. The results are presented in a single result page. We implemented a *cross-cultural bookbinding information system* as a prototype to demonstrate how to search in multiple databases with one query defined in natural language.

2. Related Work

In the literature, there are several works on the topic of FIR. Federated search can be challenging in terms of retrieving relevant information for the user. We present a few approaches, each describing a different focus.

Shokouhi and Si [8] have provided a foundational definition of federated search and delved into its potential applications and challenges. Their work notably sheds light on the persistent issue of maintaining up-to-date representation sets, proposing innovative methods to address this challenge. In addition to the definition of federated search, their contributions have been pivotal in understanding and mitigating issues related to the timeliness and accuracy of search results in federated systems.

Building on the concept of federated search, Demeester et al. [9] conducted a study focused on the use of snippets, rather than entire webpages, to predict the relevance of a given page. This approach, which examines the content at a more granular level, proves to be particularly valuable in the context of federated search. By offering insights into deeper details of page content, their work contributes to enhance precision and efficiency of federated search algorithms. Efficiency of query processing will also be an issue in our work.

Federated search, while promising, can be inherently challenging when it comes to retrieving pertinent information for users. FedCDR [10] introduces a novel approach known as federated cross-domain recommendation. This innovative method addresses the delicate balance between providing users with tailored recommendations while safeguarding their private data. FedCDR's contributions are instrumental in ensuring that federated search remains user-centric and privacy-conscious. This aspect of safeguarding of private data should definitely be addressed in productive systems.

Furthermore, Melzer et al. [11] present a methodology designed to simulate federated databases, offering a means of conducting feasibility studies before committing to the implementation of real federated databases. This approach enables researchers and organizations to experiment and assess the viability of federated database projects, reducing the risk of investing substantial resources in endeavors that may ultimately prove unfeasible.

In the area of federated search, the diversity of data sources often poses a significant challenge due to the heterogeneity of data formats and structures. Addressing this concern, Melzer et al. [6] introduce a novel indexing process tailored to matching data from XML files and the relational representation of research data so that efficient searches across heterogeneous data sets are given.

The process of building information systems on demand is described in [4]. This innovative approach empowers humanities scholars by allowing them to construct information systems without the arduous task of manually transferring data.

Finally, recent advances in NLP (**N**atural **L**anguage **P**rocessing), particularly the use of models such as GPT (**G**enerative **P**re-trained **T**ransformer), have shown promise in simplifying the creation of SQL queries.[5] Using GPT-based NLP techniques, users can formulate queries in natural language, which are then automatically translated into SQL queries that retrieve relevant information from various federated databases. This innovative approach not only streamlines the query process, but also enables a wider range of users, including those who do not have extensive SQL knowledge, to effectively use the full potential of federated information systems. Therefore, we will integrate this functionality into a cross-domain information system.

3. Bookbinding

Binding is the process by which stacked sheets or quires are secured along one edge with needle and thread or other materials such as loose-leaf rings, binding posts, twin-loop spine coils, plastic spiral coils, and plastic spine combs. The bound stack of leaves can then be enclosed in a cover. Bookbinding is a skilled craft that requires measuring, cutting, and gluing, and combines skills from the trades of paper making, textile and leather-working crafts, model making, and graphic design. There are various types of bookbinding techniques, they are imparted by tradition, evolve across time taught from one generation to the next, and assume distinctive traits according to the area to which they belong. The presence of recurring patterns in the structures allows to group the bindings accordingly, thus identifying macro-areas corresponding to different binding traditions (Coptic, Ethiopian, Islamic, Byzantine, etc.). Modern binding methods are numerous, such as perfect binding, case binding, saddle stitch binding, PUR binding, singer sewn binding, section sewn binding, Coptic stitch binding, wiro and comb binding. [12, 13] Three different bookbinding techniques are described in the following.

3.1. Ethiopian Bookbinding

When writing was adopted by the Semites who settled in the area between the northern highlands of the Horn of Africa and the Red Sea. The existence of an extensive Christian literature going back to the fourth century CE implies the use of manuscripts. The Ethiopian language and script used for centuries as the literary language of the Christian kingdom of Ethiopia are very similar to those used in the fourth century. Ethiopian bookbinding is one of the material expressions of the ancient manuscript culture of Ethiopia and Eritrea, which is the research field of the *Beta maṣāḥəft* project. The expression ‘Ethiopian bookbinding’ identifies a set of structural features shared by the bindings of Christian manuscripts produced in Ethiopia and Eritrea. These include chainstitch sewing (mostly) on paired sewing stations, slit-braid endbands, and wooden boards, which may be covered with leather and lined with colourful textiles. In Ethiopic manuscripts, the writing support is usually parchment, produced without making use of lime baths. [14]

[5] https://github.com/soumyansh/NLP-To-SQL

3.2. Coptic Bookbinding

The expression *Coptic bookbinding* is commonly used to refer to the binding techniques prevalent in Egypt in the Late Antique and Early Medieval eras. *Coptic bookbinding* is a historical expression, deeply rooted in the literature, which refers to the binding tradition prevalent in Egypt during the Late Antique and Early Medieval periods. Coptic book structures vary, and include single quires attached directly to the leather cover using tackets; multi-quire codices sewn with chainstitch and furnished with wooden boards, or laminated papyrus boards with leather covers.

3.3. Pothi and Codex Binding in South India

In South India, a traditional manuscript — or *pothi* — consists of a stack of palm leaves, in landscape format, inscribed with a stylus, and bound together with a string thread through holes in the folios. These folios were often protected with wooden board covers. But with the arrival of Portuguese traders and missionaries in the 16th century, a new manuscript format became increasingly common: the codex. The early codices from South India and the way in which Western bookbinding techniques were learnt and applied by local craftsmen have hardly been researched so far. By the 19th century, new hybrid formats had begun to emerge across India: Sanscrit 1232, preserved at the National Library of France, is a fascinating codex-pothi hybrid, a lithograph printed in horizontal pothi format but collated in sections of two bifold each. This data, on early modern South Indian bookbinding, has been collected by the Texts Surrounding Texts project, a catalogue of Indian manuscripts from the National Library of France and the Staats- und Universitätsbibliothek Hamburg.

3.4. The Need for a Cross-Cultural Bookbinding Information System

To date, there has been little work on comparing bookbinding practices across cultures. Codicological expertise does not necessarily translate from one field to another; an expert in Coptic bookbinding would not know how to approach an Indian manuscript, or vice versa. As a result, research projects usually focus on a specific culture and a specific time period. To compare practices across cultures, we would need to, firstly, understand which data can be compared, and secondly, to collate that data by extracting it from different, heterogeneous databases. In the three aforementioned databases that will be used as the foundation for the Cross-Cultural Bookbinding Information System, we have initially selected three features to be compared: the number of sewing stations, leaf width, and leaf height. For the first time, we will be able to compare bookbinding techniques as they spread across space and time, and as they crossed boundaries of language, religion, and material tradition.

4. Matching Bookbinding Data

In general, matching data sets involves comparing two or more data sets to identify similar elements. The process of matching data involves several steps (see Figure 1). According to [15], the first step is data pre-processing, which involves preparing the data sets for matching.

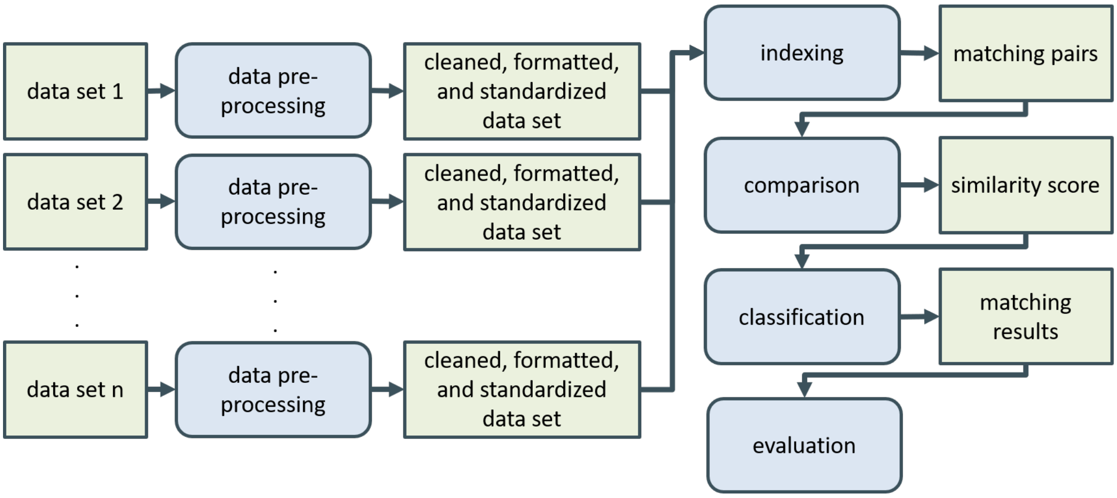


Figure 1: The general process of matching n data sets. Based on [15] (extended). Image source: [6].

This includes cleaning, formatting, and standardizing the data sets to ensure compatibility and effective comparison. The second step is indexing, which is a strategy to pre-select potential matches and leads to a reduction in the number of matches. Indexing usually involves identifying the key variables that will be used to match the data sets. The third step is comparison, where the data sets are compared to identify matches. The fourth step is classification, where the matching records are classified as match or non-match. The final step is evaluation, which involves validating the matched data sets and reviewing the results for accuracy and completeness. This may involve checking for errors, inconsistencies, or missing data and making any necessary adjustments.

In [6] an improvement of the indexing procedure using XML (EpiDoc) and relational representations of research data as input, is presented.

Pre-Processing The following projects have different relational representations to describe manuscripts and bookbinding techniques.

- The *Beta maṣāḥǝft* project has the following relation representation. The column names are “Title”, “Editor(s)”, “PubPlace”,“Manuscript Item(s)”, “idno”, “Material”, “Deco Note(s)”, “Hand Description”, “Binding”, ”Orig. Date”, ...
- The *Coptic Bookbinding* project has the following relation representation. The column names are “CLM”, “TM”, “Shelfmarks”, “Leaf width”, “Leaf height”, “Board height”, “Board width”, ”Spine width”, “Type of sewing”, “No. of sewing stations”, “Fold pattern”, ...
- The *Texts Surrounding Texts* project has the following relation representation. The column names are “Title”, “Shelfmark”, “Format”, “Technology”, “Material”, ”Leaf width”, ”Leaf height”, “Leaf depth”, “Binding”, ...

It can be seen that not all column names have the same name. The bindings are described under “Deco Note(s)” in *Beta maṣāḥǝft*, this data can be found under “Binding” in the other both

Table 1
Snippet of matching candidates

Project	Column name	XML tag / JSON node	matching candidate C
Coptic Bookbinding	CLM	clmid	no
Coptic Bookbinding	TM	tm	no
Coptic Bookbinding	leaf width	width	yes
Coptic Bookbinding	leaf height	height	yes
Coptic Bookbinding	No. of sewing stations	sewingstationsno	no
Beta maṣāḥəft	Title	title	no
Beta maṣāḥəft	leaf width	width	yes
Beta maṣāḥəft	leaf height	height	yes
Beta maṣāḥəft	Deco Note(s)	decoNote	no

projects. However, while the "number of sewing stations" is described under "'Deco Note(s)" in *Beta maṣāḥəft*, this data is found under "No. of sewing stations" in *Coptic Bookbinding*. At this point, a mapping function must therefore be defined (by the humanities scholars) so that one knows which column names are mapped to one another. If the mapping rules are not known, one can also use large language models (LLMs), as also shown in [6], to obtain them. However, it should be noted that the schemes should be given as input so that the results of the LLMs can be used.

In this article, we explain the indexing process using the two projects: *Beta maṣāḥəft* and *Coptic Bookbinding*.

Indexing Indexing includes identifying the key variables for an efficient data matching process. For existing relational databases, it can be assumed that the column names belong to the key variables and are used for their project-specific analysis. Therefore, the column names are regarded as key variables.

To identify the matching candidates, we use the XML and the JSON scheme used in the projects (where the raw data is stored).

Formally: If a set A of XML tags and B a set of JSON nodes, where the sets A and B are from different schemes, are mapped to the same element, then that element is a matching candidate to be added to the matching candidate set C.

Let $A = \{a_1, \dots a_i\}$ and $B = \{b_1, \dots b_j\}$ be sets of XML tags or JSON nodes, and let f be a function which represents a mapping from A to B: $f : A \to B$, then the matching candidates C are given by:

$$C = \{b \in B : \exists a \in A \text{ with } f(a) = b\} \tag{1}$$

In our example, the "JSON" schema belongs to set A and the TEI schema *Beta maṣāḥəft* to set B. Table 1 displays the column names used in the respective projects and the corresponding XML tags and JSON nodes. The matching candidates are $C = \{\text{leaf width}, \text{leaf height}\}$.

In our project, however, we also need the "number of sewings" that are not considered in the matching candidates.

I. e. the comparison of the "number of sewings" in this example is done via height and width. In order for "number of sewings" to be a matching candidate, it should be noted here that a

Table 2
Matching data of *Coptic Bookbinding* and *Beta maṣāḥǝft*

Project	ID	CLM or ID	width (mm)	height (mm)
Coptic Bookbinding	a_1	193	235	290
Coptic Bookbinding	a_2	36	140	145
Coptic Bookbinding	a_3	179	130	295
Coptic Bookbinding	a_4	3011	130	165
Beta maṣāḥǝft	b_1	DSEthiop13	90	115
Beta maṣāḥǝft	b_2	SinaiNewEt001	130	165
Beta maṣāḥǝft	b_3	C4IV123	286	326
Beta maṣāḥǝft	b_4	ESumo58	230	306

standard should be applied semantically correctly in the various projects or an adjustment could be made in the pre-processing.

Matching In Table 2 each matching data (width and height) of both projects (*Coptic Bookbinding*, *Beta maṣāḥǝft*) were assigned an id. The table also present some more data (CLM/ID) to have a better overview of the data.

The content of the matching candidates are compared are compared for equality (:=1) or inequality (:=0). We use this simple comparison because only values need to be compared. For words, texts or dates, other comparison approaches such as the Soundex algorithm [16], Levenshtein distance [17] or suitable artificial intelligence (AI) algorithm can be used instead. If we consider all matching candidates (separate comparison of width and height data), the identified record pairs are: (a_3, b_2), (a_4, b_2).

The fact that only one record pair was identified is due to the simple number matching. With the leaf width and height, one could also allow smaller deviations if it fits the content. For a simple illustration of the indexing process, we will first continue with the one identified matching candidate.

Comparison The comparison process in schema matching indicates the degree of similarity between two record pairs to determine whether they are a match or not. In general, for the comparison process all fields are considered. In Table 3 the column names are: width (mm), height (mm), and No.sewing. Consider that "decoNote" and "sewingstationsno" represent both "No.sewing."

The comparison function $c(a_i, b_j)$ maps the content of each column value of a_i and b_j in the range $[0, 1]$, where 0 indicates no similarity and 1 indicates a perfect match. The comparison function can be defined using different similarity metrics depending on the characteristics of the schema elements and the matching criteria.

The following comparison function can be used to rank the candidate matches based on their similarity scores:

$$\text{sim}_{\text{all}}(a_i, b_j) = \sum_{n=0}^{\text{number of attributes - 1}} c(a_i(n), b_j(n)), \tag{2}$$

Table 3
Comparison

ID	width (mm)	height (mm)	No.sewing	sim$_{\text{all}}$
a_3	130	295	4	
b_2	130	165	4	
	1	0	1	2.0
a_4	130	165	4	
b_2	130	165	4	
	1	1	1	3.0

where an attribute is a column name and n is the position of the column.

The classification of each compared record pair can be based on either the full comparison vectors or on the summed similarities. Based on the summed similarity score, a match is defined as:

$$\text{match} = \begin{cases} 1 & \text{sim} \geq \theta \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

In the context of the project, a good value for θ is between the “number of attributes” divided by 2 and the total “number of attributes” to achieve matching results between approximately 50% and below 100%. Formally:

$$\frac{\text{number of attributes}}{2} \leq \theta < \text{number of attributes}. \tag{4}$$

If $\theta =$ “number of attributes” (100% similarity), then it could indicate a duplicate.

This matching algorithm can compare the data in an offline process. This algorithm can then be implemented in FIR in such a way that the category, such as “No. of sewings” is created and the most similar data sets are displayed to the user.

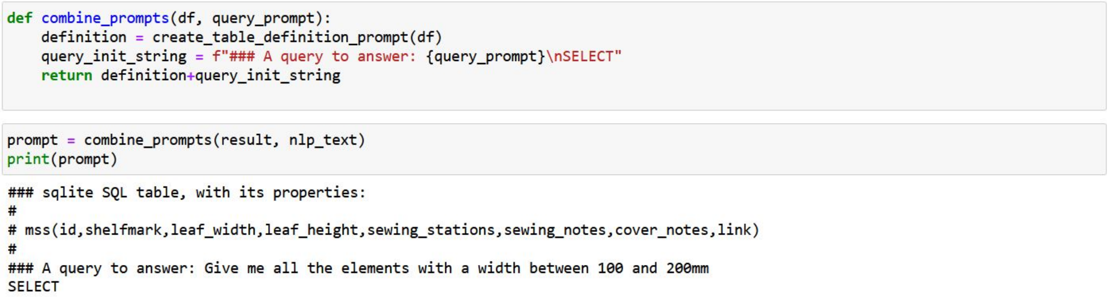
5. Federated Information Retrieval

FIR, also known as distributed information retrieval or federated search, is a technique used to search multiple data sources simultaneously. It allows users to retrieve information from various content locations with just one query and one search interface. Federated search has revolutionized how user search and retrieve information online, making it easier for researchers to manage data and search for data. Implementing a federated search engine can be challenging, especially when integrating the system with heterogeneous databases. Federated search is an efficient option for mid-to-low funnel users who know exactly what they need and can search through a large body of data from one location with one query, reaching their goal with fewer efforts.

Architecture In recent years, the Sqlite database has become more and more common as a way to share research data. For example, the website of the Texts Surrounding Texts Project is a front-end that queries a read-only Sqlite database hosted on GitHub. This architecture means that the project automatically has its own, open API — any researcher, any website can also access the database using SQL queries, without requiring any authentication. A cross-cultural bookbinding information system takes advantage of this openness by connecting directly to the Texts Surrounding Texts Sqlite database and extracting bookbinding data from it, which is then collated with bookbinding data from the Ethiopian, Eritrean, and Coptic databases. In demonstrating our federated search application, we hope to encourage more and more research projects to make their databases openly accessible in this way, so that researchers can more easily cross-reference data from multiple sources.

Querying The use of natural language queries instead of SQL for accessing databases has been an area of active research in recent years. One approach involves the use of transformer models such as GPT to generate SQL queries from natural language queries. We used the GPT4All[6] library with the "wizardlm-13b-v1.1-superhot-8k"[7] model to generate SQL queries from natural language queries. The source code for this implementation is based on the code of "soumyansh" on GitHub [18].

The basic idea of this querying approach is to pass information about the database, in our case the names of the table together with the column names, together with the prompt given by the user. An example prompt can be seen in Figure 2. Since we only want to allow "SELECT" statements to be executed automatically on the databases, it is given as part of the prompt to the GPT. After the prompt has been generated, it is passed to the chosen GPT model using the GPT4All library (see Figure 3). Depending on the hardware resources, chosen model and query, execution time is around 45 to 60 seconds. Once the GPT has generated an output, it is further passed on to the functions responsible to generate the webpage. This querying approach can be generalized to n databases by repeating the process n times. While this makes it easy to apply the same principle to an undefined number of databases, it also increases execution time per database added. Further work needs to be done to make the process faster when using a large number of databases to query.



```
def combine_prompts(df, query_prompt):
    definition = create_table_definition_prompt(df)
    query_init_string = f"### A query to answer: {query_prompt}\nSELECT"
    return definition+query_init_string
```

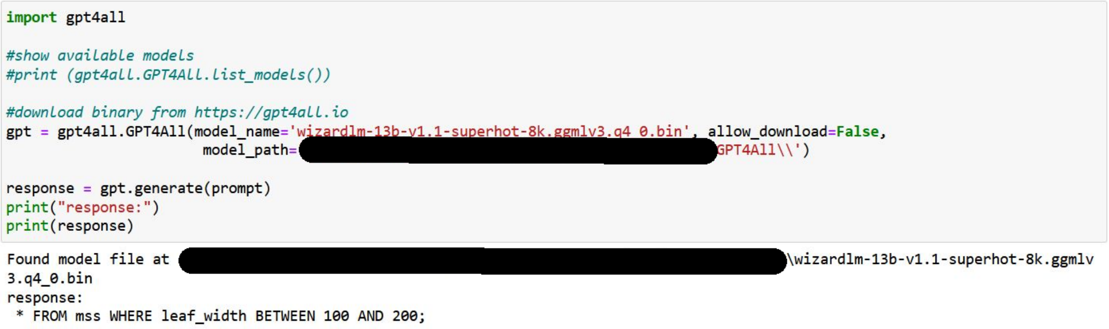
```
prompt = combine_prompts(result, nlp_text)
print(prompt)
```

```
### sqlite SQL table, with its properties:
#
# mss(id,shelfmark,leaf_width,leaf_height,sewing_stations,sewing_notes,cover_notes,link)
#
### A query to answer: Give me all the elements with a width between 100 and 200mm
SELECT
```

Figure 2: Method combining the information about the database table with the user input

[6] https://gpt4all.io/index.html

[7] https://huggingface.co/TheBloke/WizardLM-13B-V1-1-SuperHOT-8K-GGML/resolve/main/wizardlm-13b-v1.1-superhot-8k.ggmlv3.q4_0.bin



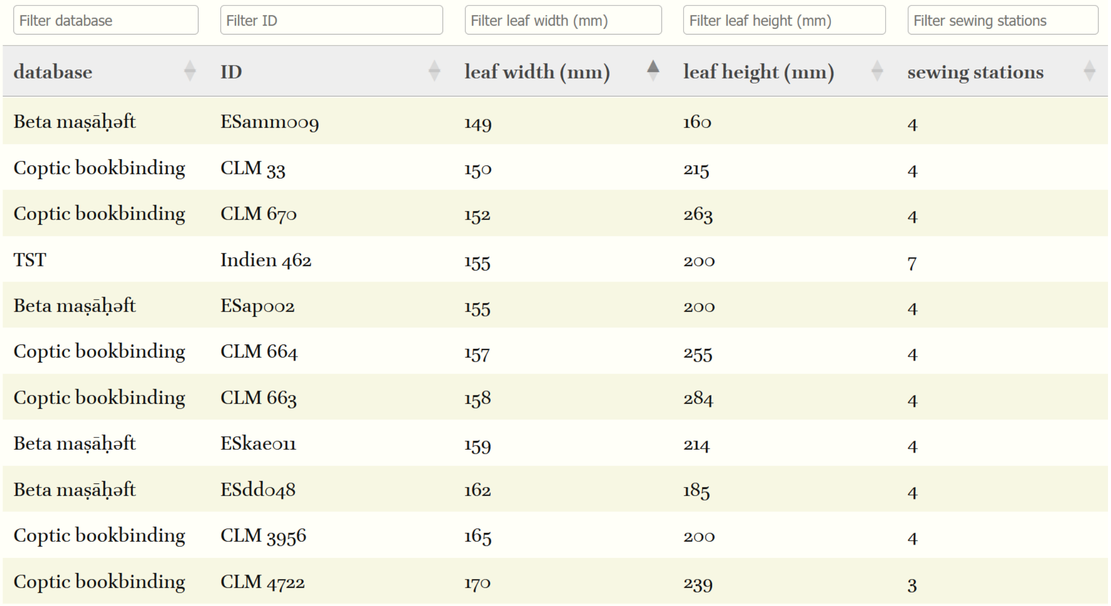
```
import gpt4all

#show available models
#print (gpt4all.GPT4All.list_models())

#download binary from https://gpt4all.io
gpt = gpt4all.GPT4All(model_name='wizardlm-13b-v1.1-superhot-8k.ggmlv3.q4_0.bin', allow_download=False,
                      model_path=[redacted]GPT4All\\')

response = gpt.generate(prompt)
print("response:")
print(response)
```

```
Found model file at [redacted]\wizardlm-13b-v1.1-superhot-8k.ggmlv
3.q4_0.bin
response:
 * FROM mss WHERE leaf_width BETWEEN 100 AND 200;
```

Figure 3: GPT4All result

Federated Bookbinding Information System By extracting bookbinding data from the three databases pertaining to three different manuscript traditions, we can begin to compare how the codex format was adapted by different cultures at different periods of time.

In our prototype (see Figure 4) it can already be seen that desired database entries from different database instances can be viewed in one view. This joint representation makes it easier to answer the research question and to prove it with concrete values. The additional linking to similar documents through the matching algorithm improves the overview of information. The prototype still needs to be further refined over time, as not all queries have been answered correctly so far. Additionally, it takes a few minutes to execute queries. While a user only sends one natural language query to the system, it internally generates a separate SQL query



Filter database	Filter ID	Filter leaf width (mm)	Filter leaf height (mm)	Filter sewing stations
database	ID	leaf width (mm)	leaf height (mm)	sewing stations
Beta maṣāḥǝft	ESamm009	149	160	4
Coptic bookbinding	CLM 33	150	215	4
Coptic bookbinding	CLM 670	152	263	4
TST	Indien 462	155	200	7
Beta maṣāḥǝft	ESap002	155	200	4
Coptic bookbinding	CLM 664	157	255	4
Coptic bookbinding	CLM 663	158	284	4
Beta maṣāḥǝft	ESkae011	159	214	4
Beta maṣāḥǝft	ESdd048	162	185	4
Coptic bookbinding	CLM 3956	165	200	4
Coptic bookbinding	CLM 4722	170	239	3

Figure 4: Collating data on sewing stations across databases (https://uhh-tamilex.github.io/bookbinding/)

per database. This makes it possible to generate queries to databases with different table as well as column names. During our testing, the system seemed to give reasonable results. The performance will be formally evaluated at a later stage, which presents an opportunity for improvement. This *cross-cultural bookbinding information system* was created with little effort. Although work still needs to be put into a productive system for correctly responding to all user requests. Using the indexing process, we can now offer similar documents to each record. Which they are for our example will be presented in the next subsection.

Federated Search Results In an offline process, the bookbinding matching process can be applied. In Table 5 you can see the results if the three columns width, height, and number of sewing stations (cf. Figure 4) are defined as the relational structure. According to Equation 4, we receive the data sets that fulfil $1.5 \leq \theta < 3$.

Table 4
Retrieval results of similar data sets with similarity score 2 (each)

database: data set ID	similar to database: data set ID(s)
Coptic bookbinding: CLM 33	Beta maṣāḥəft: SinaiEt001
Coptic bookbinding: CLM 34	Beta maṣāḥəft: GAet5, SinaiEt004
Coptic bookbinding: CLM 35	Beta maṣāḥəft: ESamm009, GAet1, SinaiEt006
Coptic bookbinding: CLM 40	Beta maṣāḥəft: ESamm002, GAet3, SinaiEt006
Coptic bookbinding: CLM 38	Beta maṣāḥəft: ESdd048
Coptic bookbinding: CLM 179	Beta maṣāḥəft: ESdd007, ESum058
Coptic bookbinding: CLM 185	Beta maṣāḥəft: ESum042
Coptic bookbinding: CLM 714	Beta maṣāḥəft: ESap016, ESgmg006
Coptic bookbinding: CLM 37	Beta maṣāḥəft: DSEthiop2
Coptic bookbinding: CLM 3956	Beta maṣāḥəft: ESagm002, ESamm011, ESap002
Coptic bookbinding: CLM 21	Beta maṣāḥəft: ESdd007, ESum042
Coptic bookbinding: CLM 64	Beta maṣāḥəft: BerOrOct555, DSEthiop1, GotD781
Coptic bookbinding: CLM 207	Beta maṣāḥəft: ESamm008, ESbgy004
Coptic bookbinding: CLM 213	Beta maṣāḥəft: ESgmg006
Coptic bookbinding: CLM 219	Beta maṣāḥəft: ESdd007, ESmqm002
Coptic bookbinding: CLM 239	Beta maṣāḥəft: ESdd007, ESmqm002
Coptic bookbinding: CLM 240	Beta maṣāḥəft: ESath007
Coptic bookbinding: CLM 254	Beta maṣāḥəft: ESamm008
Coptic bookbinding: CLM 255	Beta maṣāḥəft: ESdd007, ESmqm002
Coptic bookbinding: CLM 39	Beta maṣāḥəft: GAet5, SinaiEt004
Coptic bookbinding: CLM 667	Beta maṣāḥəft: ESamm009
Coptic bookbinding: CLM 668	Beta maṣāḥəft: ESamm003, ESamm005
Coptic bookbinding: CLM 670	Beta maṣāḥəft: GAet7
Coptic bookbinding: CLM 662	Beta maṣāḥəft: PetermannIINachtrag42, SinaiEt001

For the “number of sewing” category, the precision score is perfect, with a value of 1. This implies that all the instances identified as belonging to the “number of sewing” category were indeed accurate, leaving no room for false positives. In contrast, for the “width” category, the precision score is 0.535, indicating that approximately 53.5% of the items classified as “width” were true positives, while the remaining 46.5% were false positives. This suggests some room for improvement in reducing false positives within the “width” category. The “height” category exhibits a precision score of 0.465, indicating that about 46.5% of the items identified as “height” were true positives, while 53.5% were false positives. Similar to the “width” category, there is potential for enhancing precision within the “height” category to reduce false positives.

Even though the values for precision are not very high for some values, it can be seen in the following that the choice for θ in this example is well chosen. If a different value is taken for theta, the results are much worse. That is for $1 \leq \theta < 3$ as follows: For the “number of sewing stations,” the precision score is 0.99 and therefore high. This implies that the process for determining the number of sewing stations is remarkably accurate, with only a 1% margin for error. However, the precision values for “width” and “height” paint a different picture. The precision score of 0.001 for “width” indicates a notable lack of precision in this measurement. Similarly, the precision score of 0.007 for “height” also suggests a measurement process that falls short in terms of precision.

In data analysis and research, the choice of parameters like θ is just one piece of the puzzle. Equally important is the alignment of these parameters with the overarching research question. In the context of the Coptic Bookbinding project, it is obvious that expanding the data set and considering additional suggestions has proven beneficial.

Researchers can improve the robustness of their analysis and enhance the overall quality of results. Incorporating more data points and seeking suggestions from similar data sets can provide a broader context and lead to more meaningful insights. This approach not only helps in fine-tuning the parameters but also contributes to a deeper understanding of the subject matter and the research objectives.

6. Conclusion and Outlook

In this article, we present a cross-cultural bookbinding information system that supports FIR with low effort. We demonstrate how federated search can be used to retrieve information from various digital resources produced by research institutions, museums, or libraries to answer cross-domain research questions. Our system integrates GPT4All to generate SQL queries from natural language queries, enabling users to search for similarities between the binding techniques, object size, or written area dimension of manuscripts from Ethiopia, Eritrea, early Egypt, and South India. We use a data matching method to make the search for finding similar data sets more efficient. With the bookbinding information system, we have succeeded in substantiating statements with numbers by combining different sources.

In the future, we plan to expand the system to include more digital collections and data sources. We also plan to improve the system’s search capabilities by integrating the similarity score calculation in our prototype. Additionally, we aim to integrate the system with other research tools and platforms to provide a more comprehensive and seamless research experience for scholars in the humanities.

Acknowledgments

The research for this contribution was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy - EXC 2176 'Understanding Written Artefacts: Material, Interaction and Transmission in Manuscript Cultures', project no. 390893796.

References

[1] E. O. F. N. Research, OpenAIRE, Zenodo, 2013. URL: https://www.zenodo.org/. doi:10.25495/7GXK-RD71.

[2] Universität Hamburg, Research Data Repository, Available: https://www.fdr.uni-hamburg.de/, 2022. Accessed March 09, 2023.

[3] Text Encoding Initiative, P5: Guidelines for Electronic Text Encoding and Interchange, Version 4.0.0, https://tei-c.org/Vault/P5/4.0.0/doc/tei-p5-doc/en/html/, 2020. Accessed 29 June 2022.

[4] S. Schiff, S. Melzer, E. Wilden, R. Möller, TEI-Based Interactive Critical Editions, in: S. Uchida, E. Barney, V. Eglin (Eds.), Document Analysis Systems, Springer International Publishing, Cham, 2022, pp. 230–244.

[5] Zwass, Vladimir, information system, Encyclopedia Britannica, https://www.britannica.com/topic/information-system, 2023. Accessed 28 July 2023.

[6] S. Melzer, M. Klettke, F. Weise, K. Harter-Uibopuu, R. Möller, EpiDoc Data Matching for Federated Information Retrieval in the Humanities, in: 1st International Workshop on AI in Digital Humanities, Computational Social Sciences and Economics Research at part of the 18th Conference on Computer Science and Intelligence Systems (FedCSIS), Proceedings of the 2023 Federated Conference on Computer Science and Intelligence Systems, 2023, pp. 1063–1068. URL: https://annals-csis.org/proceedings/2023/pliks/1515.pdf.

[7] T. Elliott, G. Bodard, E. Mylonas, S. Stoyanova, C. Tupman, S. Vanderbilt, et al., EpiDoc Guidelines: Ancient documents in TEI XML (Version 9)., Available: https://epidoc.stoa.org/gl/latest/., (2007-2022). Accessed January 22, 2022.

[8] M. Shokouhi, L. Si, Federated Search, Found. Trends Inf. Retr. 5 (2011) 1–102. URL: https://doi.org/10.1561/1500000010. doi:10.1561/1500000010.

[9] T. Demeester, D. Nguyen, D. Trieschnigg, C. Develder, D. Hiemstra, Snippet-Based Relevance Predictions for Federated Web Search, in: P. Serdyukov, P. Braslavski, S. O. Kuznetsov, J. Kamps, S. Rüger, E. Agichtein, I. Segalovich, E. Yilmaz (Eds.), Advances in Information Retrieval, Springer Berlin Heidelberg, 2013, pp. 697–700.

[10] W. Meihan, L. Li, C. Tao, E. Rigall, W. Xiaodong, X. Cheng-Zhong, Fedcdr: Federated cross-domain recommendation for privacy-preserving rating prediction, in: Proceedings of the 31st ACM International Conference on Information & Knowledge Management, CIKM '22, Association for Computing Machinery, New York, NY, USA, 2022, p. 2179–2188. URL: https://doi.org/10.1145/3511808.3557320. doi:10.1145/3511808.3557320.

[11] S. Melzer, S. Thiemann, R. Möller, Modeling and Simulating Federated Databases for early Validation of Federated Searches using the Broker-based SysML Toolbox, in: IEEE International Systems Conference, SysCon 2021, Vancouver, BC, Canada, April 15 - May 15, 2021, IEEE, 2021, pp. 1–6.

[12] Matt Marzullo , WHAT'S IN A BIND? 4 TYPES OF BOOK BINDING - PROS AND CONS, https://blog.ironmarkusa.com/4-types-book-binding, 2021. Accessed 28 July 2023.

[13] Wikipedia, Bookbinding, https://en.wikipedia.org/wiki/Bookbinding, 2023. Accessed 28 July 2023.

[14] Universität Hamburg , Background, https://www.betamasaheft.uni-hamburg.de/about/background.html, 2017. Accessed 28 July 2023.

[15] P. Christen, Data Matching: Concepts and Techniques for Record Linkage, Entity Resolution, and Duplicate Detection, Springer Publishing Company, Incorporated, 2012.

[16] J. Jacobs, Finding words that sound alike. The SOUNDEX algorithm., Byte 7 (1982) 473–474.

[17] F. P. Miller, A. F. Vandome, J. McBrewster, Levenshtein Distance: Information Theory, Computer Science, String (Computer Science), String Metric, Damerau-Levenshtein Distance, Spell Checker, Hamming Distance, Alpha Press, 2009.

[18] soumyansh, NLP-To-SQL, https://github.com/soumyansh/NLP-To-SQL, 2023. GitHub repository, Accessed 01 September 2023.