



NATIONAL LEADERSHIP GRANTS FOR LIBRARIES

Sample Application LG-259108-OLS-25
Project Category: Planning

Curators of The University Of Missouri (School of Information Science & Learning Technologies)

Amount awarded by IMLS: \$149,994

The University of Missouri, in partnership with the University of Wisconsin-Milwaukee, will plan for the investigation, exploration, and testing of the capabilities of Artificial Intelligence to map Knowledge Organization Systems in the form of application programming interfaces to make data search and retrieval easier. The project features the production of several digital products including project webpages at each institution; GitHub repository for code storage and documentation; and Jupyter notebooks usage for code samples and testing. The project outcomes will include: a series of methods including prompting scripts to engage in conversation with publicly available generative artificial intelligence to map, reference, and learn to query other open application program; documented Python code for interacting with application programming interfaces for artificial intelligence platforms, with maps in the form of tables and diagrams for currently available data in Census, and PubMed Entrez application program interfaces; an expandable library and information science-driven search for Census.gov and PubMed data; and lookup tables and code for translating Census JavaScript Object Notation outputs to usable data.

Attached are the following components excerpted from the original application.

- Narrative
- Schedule of completion

When preparing an application for the next deadline, be sure to follow the instructions in the most recent Notice of Funding Opportunity for the grant program to which you are applying.

Leveraging AI for Documenting, Developing Understanding, Use, and Retrieval of Open Government Knowledge Organization Systems (AI-for-KOS)

Introduction

The University of Missouri's (MU) School of Information Science & Learning Technologies, in partnership with the University of Wisconsin-Milwaukee's (UWM) School of Information Studies, seek \$149,996 in support of a two-year **planning grant** from the IMLS National Leadership Grant for Libraries (NLG-L) program. We are investigating, exploring, and testing the capabilities of Artificial Intelligence (AI) to map Knowledge Organization Systems (KOSes) in the form of Application Programming Interfaces (APIs) to **make data retrieval easier**. This project aligns with NLG-L Goal 3 and Objective 3.2 to **promote preservation and access**. It addresses a gap in the **availability** and **accessibility** of **open government data** (OGD) resources (a **public good**), and seeks to simplify the technical know-how required for **accessing, retrieving, and using big data** for librarians, researchers, and members of the public by simplifying querying APIs.

Internet APIs have not traditionally been considered a part of library and information science education. APIs are written for programmers, by programmers, and often have limited documentation and an arcane structure. RESTful APIs, which result in the creation of a special URL with an embedded query, are common on the web. Advances in generative artificial intelligence (GenAI) have made access to APIs potentially easier for non-programmers, but GenAI will still need to be prompted to perform this task by users. Users who are unfamiliar with the process will need guidance and an interface to generate the correct URLs based on their querying needs.

Project Justification

Vignette 1: A first year master's student at a university needs MeSH data associated with a PubMed Central search on a gene to examine MeSH categories for corresponding publications. PubMed's interface allows for downloading only basic metadata. To retrieve associated MeSH terms with the search, the student needs to extract the PMIDs and query PubMed Entrez, which is available as a REST API, where a URL is formatted as a string, or through a [programmatically interface such as BioPython](#). The health librarian who helps them does not know Python, and has struggled to be approved for access to [PubMed Entrez Direct](#) due to restrictions put in place by campus IT. The student is frustrated and gives up, choosing a private vendor product that doesn't quite meet their needs because they can't otherwise search for and retrieve the data they need.

Vignette 2: A public librarian follows the [Webinars for Census.gov](#), where the videos make using the API look easy. She takes notes, [references the API documentation](#), and succeeds with provided sample queries. When she tries to create her own queries for her library service area, formatting the strings with the appropriate flags proves frustrating and the time expenditure involved to create a query that drills down into the data she needs for effective community programming (e.g., languages spoken at home, renters vs. homeowners in a certain school district) is too great. Her queries do not work, "getting anything feels like winning a jackpot," and she "[doesn't] have all week to figure it out" with her reference desk shift schedule and other duties.

We see a better way.

We are working toward creating (1) a simple web interface to Python code, which will allow users to query official APIs for Census.gov and PubMed using natural language, and to copy and paste the output to query the API with their individual key. The intended primary audience of this tool will be librarians and library students seeking information necessary for the running of their current or future libraries. (2) We also see a place for the development of the kind of directed AI-prompting scripts based on Retrieval Augmented Generation (RAG)-style prompts we are developing with the use of an API such as OpenAI, or using a chat-style LLM such as Microsoft Copilot, as having a role in public data access and preservation—as such scripts will also be applicable to other APIs, data, and documentation in the future. We are not training our own model, but "teaching" publicly accessible GenAI to go to the actual, relevant data, rather than open web chatter, for answers.

Research Question: How can AI and other tools be leveraged to map, document, understand, and simplify querying KOSes, especially APIs?

AI has been recommended for and used to enhance the engineering of knowledge representation ([Qian, 2020](#)) and to assist in mapping and engineering interdisciplinary vocabulary ([Greenberg et al., 2021](#)). Thus far, no work has applied the potential of AI to map existing KOSes to make end-user information retrieval easier, but

promising methods exist (e.g., RAG [de Boer & Stork, 2024]). KOSes serve as the backbone of information organization for facilitating search and information retrieval, usually in a graphical interface (e.g., documents, metadata, data), and **provide access to OGD** (Hodge, 2015). APIs take a step to the more technical, map to a URL, and include a method for accessing and retrieving the data. APIs reflect the state the data was saved in, and not necessarily the most logical way of retrieving data (under review, 2025). In some cases this is a literal translation of paper-based tabulation and storage methods to the electronic, as in the Census. We note that census.gov has recently added [a better web interface for retrieval](#); however, it poses a number of issues. These issues include that identifying which data table to use is difficult; data retrieved in the search interface is organized into pivot tables (which are **not** machine readable); and directly downloading the data from the tool is less than ideal as the headers use the API names (e.g., “B16001_002M”), with corresponding details (e.g., “Margin of Error!!Total!!Speak only English”). This notation forms a barrier for the average person wanting to access, download, clean, and use this data outside of the data.census.gov interface.

Projects supporting the use of OGD platforms, census.gov, and PubMed Entrez exist in various forms, some more official than others. However, these workarounds require IT support as well as sufficient familiarity with various programming languages and environments to navigate the API. This widens the digital divide between lay users who just need to access OGD and the system. It shouldn’t be this hard, but it is.

In one example, the US Census Bureau partnered with data.world to create the American Community Survey Linked Open Data (data.world, 2017) project. Using this project requires knowledge of SPARQL and use of Amazon Web Services, and, notably, has not been updated since 2017. Other projects require knowledge of Ruby, Python, R, and various dependencies, creating a steep learning curve for most potential users.

The current PubMed Entrez support and workarounds are largely created for users in the biomedical sciences, rather than our proposed users. BioPython’s Bio.Entrez package relies on the Document Type Definition, or DTD, for “the most commonly used Entrez Utilities” (The Biopython Contributors, 2020). Others require knowledge of PHP, Rust, Python, BioPython, Prolog, Flask, JavaScript, R, and assorted dependencies. Our PI would like to note that in addition to being inaccessible to the average user, these utilities do not include extensive support for bibliographic information from PubMed and PubMed Central.

This project’s use of publicly available AI models will allow for easier and faster updating and generation of tables, mappings, and new data access features than if we were to develop and support a custom model with an end-of-life date at the end of the grant. This will also allow for updating to be done by general users rather than IT specialists, while minimizing cost.

Why Such Tools are Needed

Data storytelling for libraries (McDowell, 2023) is recommended to provide a direction to data, giving context and meaning to help guide the cognition of the users of a visualization or presentation. Sometimes libraries **do** have the data, as McDowell claims that libraries are often “awash in data,” but, when that isn’t the case, where is the data supposed to come from? How do you find out how many library patrons in a given user base commute by car or by bus? Or would it be helpful to also have on-site bike storage? What if the data is out there, but complicated to the point of being inaccessible (See Figure 1, which has an additional 260 columns to examine languages spoken at home)?

Figure 1

Snippet of Census data from data.census.gov

	A	B	C	D	E	F	G	H	I	
1	GEO_ID	NAME	B16001_001E	B16001_001M	B16001_002E	B16001_002M	B16001_003E	B16001_003M	B16001_004E	B160
2	Geography	Geographic Area Name	Estimate!!Tot	Margin of Error!!Tot	Estimate!!To	Margin of Error!!Total!!	Estimate!!Total!!Spanish:	Margin of Error!!Total!!Spanish:	Estimate!!Total!!Spanish!!Speak Enj	Marg
3	0100000US	United States	316581199	19409	245472067	221387	43369734	134347	25773177	
4	0400000US	Wisconsin	5603081	2354	5084548	11548	281262	8643	172874	
5	0500000US	Milwaukee County, Wisconsin	858866	363	695067	5779	105633	4259	65499	
6										

260 additional columns →

Our project will help users retrieve otherwise inaccessible (or difficult to access) data in formats that can be used to address such questions. We will create a RAG system consisting of a retriever and a generator. After a user inputs a query or prompt, the retriever is responsible for searching a set of predefined documents, databases, or a corpus of text for query-relevant information (Gupta et al., 2024; Shahade & Deshmukh, 2024). The user’s query is transformed into a vector representation, and similarity scores are computed between the query and vector representations of chunks of text from external sources. Multiple documents or relevant

pieces of text are returned and ranked by relevance or similarity with the query (Gao et al., 2023). This information is passed into the generator with the query. The generator, essentially an LLM, takes the contextual information from the retriever and the input from the prompt to generate an output rooted in the retrieved content (Gupta et al., 2024; Shahade & Deshmukh, 2024). This ensures that the generated text is not only contextually accurate but also grounded in factual information from the external knowledge base. RAG helps address common limitations of traditional LLMs, such as hallucinations—where models generate plausible but factually inaccurate or misleading information—by anchoring responses in verifiable external sources (Gupta et al., 2024; Ji et al., 2023). The seeming magic in RAG derives from the identity given to the AI as part of the prompt, which better directs the AI to answer questions using the specific data.

Our target systems, **census.gov** ([130+ sources](#)) and **PubMed Entrez** ([30 total databases](#)), were selected due to the frequency of their use in the library profession ([Bonnell, 2023](#); [Wessel et al., 2003](#)). Of these resources for the Census, we plan to start with census.gov's American Community Survey and Decennial Census, mapping backwards by year for 3-5 years. We will prioritize other Census Surveys and years based on input from the library community collected via a survey in Phase 0. For PubMed Entrez, our focus is providing support to access PubMed and PubMed Central, which are both bibliographic databases.

The Census Website houses several data tools to assist in locating and extracting relevant query data using a search engine-style keyword search or an API ([data.census.gov](#)). PubMed supports [PubMed Entrez Direct](#) (EDirect), which is accessed using a command line interface after installing EDirect via Terminal (Mac) or Windows using Cygwin (we suspect that PowerShell may also work on Windows 11, but PubMed's documentation is out of date). However, there are multiple issues with the extant tools; they are not user-friendly, particularly for the lay user, and are frequently ineffective in providing access to desired OGD (see *Vignettes 1 & 2* for examples).

More specifically, two points are barriers to OGD access. First, translating the documentation for a KOS into a method for retrieving data from an API is quirky, at best, and API-only access to downloadable OGD is cited as a barrier ([Crusoe & Melin, 2018](#)). Second, work-arounds to avoid direct use of APIs require specialized knowledge of third-party tools (see *Vignettes 1* and *2*), knowledge of programming, and often substantial guesswork cross-referencing documentation, examples, and help boards. Third, information- and IT-centric organizations can fall into the trap of relying on one person for their expertise (for example, in the use of these tools), and do not necessarily have good practices for encoding knowledge “in one person's head” so that their experience is not lost when they leave an organization ([Del Canele, 2013, p. 24](#)).

In addition to the issues with existing tools, a key consideration in the proposed project is that *AI systems learn from how they are queried* ([Nah et al., 2023](#)); we stress that if these systems are not tested and used for library-specific applications now, they will not support them in the future. Importantly, the application of this work is to more than just the Census and PubMed—it can be applied to other open systems using the techniques we will refine and discuss in our publications. By using publicly available models, we will help to shape how their future versions will interact with and present data. If AI systems are trained to map data and documentation, these systems may be able to help with digital preservation by supporting access to sense-making even if interfaces become unavailable in the future (e.g., BLAST or PubMed Central going down).

Project Work Plan

TEAM: PI Laura Ridenour (LR) of MU is an expert in data, information organization, linked data, and emerging AI technologies; has a background in Information Technology and systems documentation, and often translates between IT, students, researchers, and administrators; she holds certification in Digital Accessibility; teaches Python, data analytics and storytelling, collection development and management, and Systems Librarianship to MLIS students and LIS researchers. She worked in corporate IT supporting specialized and legacy hardware and software. She was also a member of the 2018 LEADS-4-NDP data science cohort.

Co-PI Margaret Kipp (MK) of UWM is an expert in data, metadata and social tagging; has a background in computer science; and teaches data science, data analytics, metadata, databases, XML, programming, and reference. She worked as a programmer in Geographic Information Systems (GIS) and as a Systems Librarian. She is the Knowledge Organization Research Group coordinator at UWM.

One hourly graduate research assistant (GRA) will be hired from each institution (MU and UWM) to bring strengths from each program to the project. Both GRAs will learn about working on an inter-institutional team

and develop skills in universal design, project planning, user study design and implementation, Python, regular expressions, expanded API programming, and working with generative AI.

ADVISORY BOARD: Cindy Dudenhoffer, Associate Teaching Professor at MU and AI Teaching Fellow, has committed to serve on the board as a volunteer. We plan to invite the new AI hire at MU to serve, and will recruit two others (with current involvement in AI, either producing or developing models) for a total of four members. We will solicit input from the board on choice of model(s), as well as feedback on the tools as we proceed in their development.

DATA SHARING: This project will be documented and posted to GitHub (for coding, testing, versioning, and documentation) during development, which will [allow for other interested individuals to contribute in a structured way](#). We plan to start by obtaining feedback from a relatively technical venue (JCDL) and then move to a more librarian-centric conference when the project is further along.

We will release software produced by this project under a GNU General Public License, and release documentation, websites, and other products not subject to publisher copyright agreements under a Creative Commons Attribution License CC-BY 4.0. We will prioritize open access (OA) publications for our published papers and other scholarly outputs. Otherwise, such outputs will be released under the publisher's approved licensing terms, with preprints made publicly available as allowed. Digital products, such as survey results or results of user testing that require IRB approval, will not be released except as aggregated or anonymized results in scholarly publications.

ACCESSIBILITY: The project will be digitally accessible from the start, implementing guidelines from the [2024 updates to Title II](#) following the WCAG 2.1 guidelines, and will be tested using edge cases suggested by librarian and researcher participants as well as the open source community.

DIGITAL PRODUCTS: We will create the following digital products during the course of the project:

1. Project webpages at MU and UWM.
2. A project Github site used to store code and documentation. The code will mainly be in Python, but may include other languages or markup languages such as XML/JSON as required by the project. Documentation may be in word processor formats such as Microsoft Word or in ASCII text formats such as Markdown.
3. Jupyter notebooks will be used for code samples and testing. We will use local installations of Jupyter notebook for initial code samples and testing. Later Jupyter notebook code may use a Binder account linked to the project Github site to allow for remote user testing in the second phase of the user study.
4. Qualtrics will be used for the initial user survey and Phase 1 of the user study; results from the survey will continue to be stored in Qualtrics format and may also be downloaded in Excel or other formats for analysis and preparation of manuscripts.
5. Binder will be used to allow for live demos of Jupyter notebooks for user testing.
6. AI prompts and AI chat transcripts.
7. User interface/testing data consisting of screen captures and transcripts of users testing the created tools. This information will not be released; screen capture videos and transcripts will be retained as per IRB requirements at the two institutions.

PROJECT PHASES

Because we will use an agile approach to this project, there may be some overlap between phases. Each phase is iterative and mapping each API subset can be extended to others. While developing code and prompts, testing, and training is time intensive, we have the expertise to accomplish bridging KOSes and AI to simplify data retrieval.

Phase 0: Preliminary Investigation of Systems and Capabilities [in progress – Jul. 2025]

Thus far, LR, LR's current GRA, and MK have piloted a method using Microsoft Copilot's web interface to walk the AI agent through a domain analysis of Census data, to great success (research paper currently under review for NASKO 2025). The method, based in RAG, has already mapped the American Community Survey to its corresponding tables for the year 2023, as well as several lookup tables (see Table 1). When walked

through a KO-style domain analysis with an appropriate RAG identity (e.g., an economist interested in supporting public libraries), we were able to have the system elucidate more about the data and how it was organized. Output Tables 1-5 contain outputs from this proof of concept consisting of metadata, data definitions, and variable codes. In particular, Tables 4 and 5 contain examples of Census-designated variable names that map back to what are essentially lookup tables (<https://api.census.gov/data.html>). Some codes are relevant to one Census survey and not another; the Census instruments are dynamic and change with changes in priorities at the Census Bureau.

Table 1

Copilot Output: Method Sample Survey-to-Data Table Mapping Output and Descriptions from Bing

The American Community Survey (ACS) for 2023 organizes its data into several types of tables, each serving a specific purpose and often interrelated. Here are the main types of tables and their relationships: 1. Data Profiles (DP): These tables provide broad summaries of social, economic, housing, and demographic data for a specific geographic area. They are often used as a starting point for understanding the general characteristics of an area[1]. 2. Detailed Tables (DT): These tables offer more granular data on specific topics covered in the Data Profiles. They provide detailed cross-tabulations and are useful for in-depth analysis[2]. 3. Subject Tables (S): These tables focus on specific subjects, such as education, employment, or housing. They provide detailed data similar to Detailed Tables but are organized by subject area[2]. 4. Comparison Profiles (CP): These tables allow users to compare data across different time periods or geographic areas. They are useful for identifying trends and changes over time[2]. 5. Geographic Comparison Tables (GCT): These tables compare data across different geographic areas, such as states, counties, or metropolitan areas. They help in understanding regional differences[2]. 6. Public Use Microdata Sample (PUMS): These tables provide anonymized individual-level data, allowing for custom tabulations and more flexible analysis. They are often used by researchers and analysts who need detailed microdata[1]. 7. Variance Replicate Estimate (VRE) Tables: These tables provide measures of sampling variability, which are essential for understanding the precision of the estimates in other tables[1]. The relationships between these tables are hierarchical and complementary. Data Profiles give a broad overview, while Detailed and Subject Tables provide more specific information. Comparison Profiles and Geographic Comparison Tables help in analyzing changes and differences across time and space. PUMS and VRE Tables offer additional flexibility and precision for advanced analysis[1][2][3].

Our next step is to test the output of machine-readable table formats using readily available models. Our top candidates include [OpenAI's GPT-3.5 Turbo model](#) and [Microsoft's TAPEX](#). GPT-3.5 Turbo is suitable for non-chat tasks and economical for testing at \$0.50/\$1.50 (input/output) per million processing tokens. TAPEX, or Table Pre-training via Learning a Neural SQL Executor, is based in SQL and may work well for the Census's tabular format. TAPEX is a huggingface transformer model and is covered by the \$9 a month premium membership. The choice of model(s) will be influenced by input from our Advisory Board.

Additional activities in Phase 0 in progress include assembling the rest of our Advisory Board, piloting mapping of the PubMed API for PubMed and PubMed Central (in progress). LR and current GRA are documenting and analyzing existing API workarounds for Census.gov and PubMed (discussed above).

Later this spring, we will connect with library communities (LR&MK + university and local public library communities) to generate interest in the project and recruit users for the survey on what librarians want from OGD, with a \$10 gift card for up to 40 participants as an incentive. Our first Advisory Board meeting will be in the summer of 2025, and sharing the grant will be done on the MU and UWM websites by IT, as well as social media websites by LR & MK, upon receiving confirmation of the grant.

Phase 1: Environmental Scan [Aug. 2025 – Feb. 2026] Assessment of available AI tools and their capabilities (TEAM) and available AI-interfacing packages (GRAs); research and testing of new methods with sample API data (GRAs); and verifying API documentation and query structure (LR & MK).

During Phase 1, we will set up the GitHub site including the repository, read.me, and additional documentation. In August 2025 we will share the project to appropriate open source communities and social media including BlueSky (LR & GRAs). A protected data and document repository for private data will be set up in OneDrive (LR&MK). Initial accessibility checks of the GitHub site will be conducted by LR, who will teach the GRAs how to conduct such checks. GRAs will complete their IRB certifications by the end of August 2025.

An environmental scan of available AI models, including relevant pre-testing to verify the models work as advertised, will be conducted by the team with the guidance of the advisory board. Thus far, we have identified TAPEX and ChatGPT 3.5 Turbo as possibilities but will continue to monitor for new developments, as GenAI is a rapidly changing technology. A writeup of the project will be produced by the team (drafted by LR, edited and contributed to by MK and GRAs) and submitted to JCDL (conference in December 2025).

As we are embracing the open source spirit of inclusion in this project, we will host a Zoom Meet and Greet with the respective MU and UWM iSchool (and other departmental) student and faculty communities who may be interested in the project in September 2025. This will be advertised with flyers and over student listservs and departmental mailers.

In Phase 1, we will distribute our survey to librarians and library students to solicit their input on working with APIs and AI tools. We will ask users what features they would want or need from a web form that would allow them to work with the APIs supported by generative AI, what past experiences users have had working with APIs, and what type of tools would have improved the experience. We will use an Internet survey to collect information from users about what they would do with the APIs if they had a tool to make searching with APIs easier. Participants, who will be librarians or library students, will be recruited via existing mailing lists and social media. We will use a snowball sampling methodology to increase the size of our participant pool for the survey. In this phase, we will also start recruiting participants for Phase 3 by asking if they would like to participate in testing the tool when it is completed.

A publication, preliminarily titled “What do librarians want from OGD?” will be the scholarly output from the Phase 1 survey, and submitted to [ITAL](#) or [Information Research](#) (both OAJs) by October 2025. An Advisory Board meeting will be held in October or November 2025, and the IRB for the user study in Phase 3 will be completed and submitted by the team in December 2025.

Phase 2: Training and Testing of AI-informed Query Methods, User Study Recruitment [Mar. 2026 – Sept. 2026] Scrape data and train selected AI models using identified methods to map PubMed and Census.gov API data (LR & MK); manually verify generated mappings with human-created REST API queries (GRA), testing of API queries using natural language and AI methods.

During Phase 2, we will begin by producing diagrams and documentation of Census Surveys and mappings to data tables and API variables using AI tools. We will also create flowcharts of PubMed structures and mappings to the API with AI tools. The project team will manually verify and manage these digital mappings. We anticipate completing this work by the end of May 2026. Beginning in parallel with the mapping work, the project leads will teach the GRAs to build regular expression templates and map flowcharts to regex query templates. Testing will be done in parallel with the development work. We anticipate completing this stage of the project by August 2026.

In the second half of the year, we will begin additional recruitment for our Phase 3 user study to ensure a suitable pool of participants for user testing. Simultaneously, we will build additional Python structures for our AI-mediated web-based search tool, such as dictionaries (regex to surveys, surveys to variables, etc.) and lookup tables based on flowcharts and data documentation (LR lead, TEAM assists). We anticipate the work will last from February to October 2026.

Our major project in the second half of the year will be testing the tools and expanding their scope with additional data sets before user testing. The team will begin testing more complex queries and adjusting the underlying Python code and web-based interface as required. Additional data from the Census and PubMed Entrez utilities will be added at this time, expanding our scope to the past 3-5 years of the Census and additional PubMed Entrez tools. We anticipate completing the work by end of year 2026 in preparation for user testing in Phase 3.

In September 2026, the project team leads (LR & MK) will host a second Zoom Meet and Greet with MU and UWM students, librarians, and members of the open source community interested in the project.

Our scholarly output from this phase of the project will be a publication on AI mapping of OGD for data retrieval. We anticipate submitting to either IPM or JASIST (both gold, or paid OAJs) between August and December of 2026.

Phase 3: Tool Production [Oct. 2026 – Jul. 2027] A Python-based query tool will be developed (LR & MK), tested (GRAs), and posted to GitHub (LR) with documentation (LR & GRAs). A read.me file for the project (GRAs) and final publications (TEAM) will be developed.

This phase of the project will be heavily focused on our user study, which will involve testing of our web-based tool for AI-mediated searching of OGD. We will continue to develop our web-based tool as the user study progresses, based on feedback from users.

We will begin the year with an Advisory Board meeting in January 2027.

During this phase of the project our focus will be continued development of the underlying Python-based query builder tool (which will be built on regex templates). We will perform continuous testing of this tool via user testing in our user survey and testing by the project team members. We will update our documentation and read.me files as required during this period in anticipation of the completion of the project and perform web accessibility checks. This work will be completed by July 2027.

Our user study, which will run in parallel with the development work described above, will involve user testing of our web-based user interface to determine how users use the interface and evaluate their responses to the interface. Participants for this phase of the user study will be recruited from the initial survey and additional librarians and library students who will be contacted via mailing lists, social media, and classroom announcements. We will expand this list of interested participants with snowball sampling, offering \$20 gift cards as a participation incentive for up to 20 participants.

We will provide a web form that allows users to search from a list of pre-selected datasets and enter their topic plus any further refinements of the topic based on the API they choose. Our web tool will provide them with a URL and instructions on how to create an API key. (Note: users of these APIs need to request a free API key from the organization. This key must be their own; we cannot share ours.) At present, we intend to provide support for this process within our tool (i.e., the tool will walk them through the process of obtaining the key). Given their key, we can generate the full URL or we can provide instructions to the user on how to insert the key themselves. The web form will generate the URL, which can be copied and pasted to a browser or sent programmatically to the API. Users will be able to select from a variety of output formats that will be generated by the underlying AI technology. The user will be able to download the file and open it in Excel or other appropriate software for formatting or analysis.

We will ask users to perform the above actions using our web-based tool and evaluate the efficiency and effectiveness of the retrieval process from the API using the natural language interface as well as the user interface itself.

This phase of the study will include a scholarly publication describing the user study and its results, which we intend to submit to KO (OA after 1 year) or another appropriate journal.

Final publication about our development process, user motivations for using the tool, and the user study are anticipated to be submitted to LRTS (OA) or other journals read by professional librarians in Summer 2027.

Project Results

Results will be implemented according to WCAG 2.1 digital accessibility guidelines and tested for usability using a screen reader such as JAWS.

Our project outcomes will include: (1) a series of methods and code notebooks (such as Jupyter Notebooks) for mapping KOSes and APIs using AI and other tools; (2) a series of methods including prompting scripts to engage in conversation with publicly available GenAI to map, reference, and learn to query other open APIs with code books (likely Jupyter Notebooks) and Knowledge Organization Systems (KOSes) using GenAI web interfaces such as Bing CoPilot and ChatGPT; (3) documented Python code for interacting with APIs (OpenAI and its models, huggingface transformers, etc.) with maps in the form of tables and diagrams for currently available data in Census, and PubMed Entrez APIs; (4) an extensible LIS-driven search for Census.gov and PubMed data; and (5) lookup tables and code for translating Census JSON outputs to usable data.

The following are example preliminary mappings using Microsoft’s Bing Copilot to map the American Community Survey (ACS) from the US Census API. Results included descriptions of “social characteristics” from the ACS and their definitions (Table 2) and Survey-to-Table summary and output, the main types of variables and their relationships (Table 3).

Table 2
User-verified Sample Social Characteristics Included in the ACS

LLM Output
1. Health Insurance Coverage: Information on the type of health insurance coverage individuals have, including private and public insurance.
2. School Enrollment and Educational Attainment: Data on the current school enrollment status of individuals and the highest level of education they have completed.
3. Residence 1 Year Ago: Information on where individuals lived one year prior to the survey, which helps track migration patterns.
4. Household and Family Type: Data on the composition of households and families, including the presence of children and the relationship between household members[1][2][3].

Generating Survey-to-Table summaries took additional prompting for full results. Table 2 contains sample output.

Table 2
User-verified Sample CoPilot Survey-to-Data Table Summary and Mapping Output

LLM Output
The American Community Survey (ACS) for 2023 organizes its data into several types of tables, each serving a specific purpose and often interrelated. Here are the main types of tables and their relationships:
1. Data Profiles (DP): These tables provide broad summaries of social, economic, housing, and demographic data for a specific geographic area. They are often used as a starting point for understanding the general characteristics of an area[1].
2. Detailed Tables (DT): These tables offer more granular data on specific topics covered in the Data Profiles. They provide detailed cross-tabulations and are useful for in-depth analysis[2].
3. Subject Tables (S): These tables focus on specific subjects, such as education, employment, or housing. They provide detailed data similar to Detailed Tables but are organized by subject area[2].

Further mapping of Tables-to-Related Tables, as a lookup table (Table 3), was successful for a descriptive output, but will require more refinement to be machine-readable. The table illustrates inter-Table relationships in the ACS for 2023. This table is illustrative of chat-based GenAI’s ability to be guided to output data mappings.

Table 3
Mapping of Tables-to-Related Tables as a Lookup

Table Type	Description	Related Tables
Data Profiles (DP)	Broad summaries of social, economic, housing, and demographic data.	Detailed Tables (DT), Subject Tables (S)
Detailed Tables (DT)	Granular data on specific topics with detailed cross-tabulations.	Data Profiles (DP), Subject Tables (S)
Subject Tables (S)	Focused on specific subjects like education, employment, or housing.	Data Profiles (DP), Detailed Tables (DT)
Comparison Profiles (CP)	Compare data across different time periods or geographic areas.	Data Profiles (DP), Detailed Tables (DT), Subject Tables (S)
Geographic Comparison (GCT)	Compare data across different geographic areas.	Data Profiles (DP), Detailed Tables (DT), Subject Tables (S)
Public Use Microdata Sample (PUMS)	Anonymized individual-level data for custom tabulations.	Data Profiles (DP), Detailed Tables (DT), Subject Tables (S)
Variance Replicate Estimate (VRE) Tables	Measures of sampling variability for understanding precision of estimates.	Data Profiles (DP), Detailed Tables (DT), Subject Tables (S), PUMS

Outputting mappings for variables was possible only after guiding the AI through the domain. Table 4 contains high-level mappings of Data Profile (DP) tables and variables (“characteristics”) by the characteristic ID.

Table 4

High-Level Mappings of Data Profile (DP) Tables and Variables by the Characteristic ID

Data Profiles (DP) Variables	Related Detailed Tables (DT)	Related Subject Tables (S)	Related Comparison Profiles (CP)	Related Geographic Comparison Tables (GCT)
Social Characteristics	B01001, B02001, B03002, B05001	S0101, S0201, S0501	CP02	GCT0101, GCT0201
Economic Characteristics	B19001, B23001, B24011	S1901, S2301, S2401	CP03	GCT1901, GCT2301
Housing Characteristics	B25001, B25002, B25003	S2501, S2502, S2503	CP04	GCT2501, GCT2502
Demographic Characteristics	B01003, B02001, B03002	S0101, S0201, S0301	CP01	GCT0101, GCT0201

Without prompting, Bing Copilot included some example mapping of variables including Social Characteristics and Economic Characteristics to DP Variables, Detailed Tables, Subject Tables, Comparison Profiles, and Geographic Comparison Tables plus samples of Census-designated variable IDs and descriptions for each (see Table 5). Elucidating more variables, and in a more structured output, is a matter of structuring the query to output mappings in a specific structure (e.g., .csv), and for specific data (e.g., named tables in relation to other tables).

Table 5

Piloted Plain Language Mapping Table

Characteristic	LLM Mapping
Social Characteristics	- DP Variables: Educational attainment, language spoken at home, disability status, etc. - Detailed Tables (DT): B01001 (Sex by Age), B02001 (Race), B03002 (Hispanic or Latino Origin), B05001 (Nativity and Citizenship Status). - Subject Tables (S): S0101 (Age and Sex), S0201 (Selected Population Profile), S0501 (Selected Characteristics of the Native and Foreign-Born Populations). - Comparison Profiles (CP): CP02 (Selected Social Characteristics). - Geographic Comparison Tables (GCT): GCT0101 (Age and Sex), GCT0201 (Race and Hispanic Origin).
Economic Characteristics	- DP Variables: Income, employment status, occupation, commuting to work, etc. - Detailed Tables (DT): B19001 (Household Income), B23001 (Employment Status), B24011 (Occupation). - Subject Tables (S): S1901 (Income), S2301 (Employment Status), S2401 (Occupation). - Comparison Profiles (CP): CP03 (Selected Economic Characteristics). - Geographic Comparison Tables (GCT): GCT1901 (Income), GCT2301 (Employment Status).

We tested the system’s ability to output a REST API query for the 2023 ACS after the initial domain analysis for this mapping, which it correctly did after an initial output in Python (see Table 6 for the REST API query and other generated information).

Table 6*REST API Query*

System Input	System Output
Using that information, write a census API query for the number of grandparents living with grandchildren in Philadelphia in 2023.	Sure! Here's how you can construct a REST API query to get the number of grandparents living with grandchildren in Philadelphia in 2023 using the US Census Bureau API: https://api.census.gov/data/2023/acs/acs1?get=B10001_001E&for=county:101&in=state:42&key=YOUR_CENSUS_API_KEY In this URL: • B10001_001E is the variable for the total number of grandparents living with grandchildren. • county:101 specifies Philadelphia County. • state:42 specifies Pennsylvania. • Replace YOUR_CENSUS_API_KEY with your actual Census API key. When you make a GET request to this URL, it will return the number of grandparents living with grandchildren in Philadelphia for the year 2023. If you need any further assistance or have more questions, feel free to ask!

Targeted venues for dissemination of results include **open access journals** (OAJ) including [Knowledge Organization](#), [Information Technology and Libraries](#), [Library Resources & Technical Services](#); and **hybrid OAJs** including [Cataloging & Classification Quarterly](#). Conference venues include the [Joint Conference on Digital Libraries](#) and the American Library Association's [Library Research Seminar](#) series. Outcomes of this project build capacity for public libraries to better understand and serve their specific user populations; for researchers to better understand their communities; and text scripts for **methods to use AI for mapping other APIs and data backups as available**.

In summary, we aim to address a gap in the accessibility and availability of OGD, a public good. We will do this by using GenAI and methods we develop to examine the KOS structures of OGD and how it is accessed in APIs, building a simple tool to query it. This promotes **data access and preservation** by illuminating the structures of OGD KOSes and how they map to the more technical APIs that govern data exchange, and we see it having a place in making accessing and using big data backups easier. Using GenAI now, according to library values and interests, will ensure that they have a place in future versions of such systems, as future GenAI versions are shaped based on how they are used in the present.

Schedule of Completion

Phase	Objective
0 1/2025 – 7/2025	- Pilot methods using AI to map APIs: Census.gov API (LR) → 2/2025 submitted to NASKO 2025 (OA); PubMed API (LR, LR-GRA) <i>in progress</i> - Test simple generated queries: <i>in progress</i> - Examine, document, and analyze existing API workarounds for Census.gov API, Pubmed API (LR-GRA): <i>in progress</i> - Sample API testing of plain language queries: <i>in progress</i> - IRB for Qualtrics survey – user study of what librarians want from OGD (LR&MK): 5/2025 - Connect with library communities (LR&MK + university communities): Late spring – early summer 2025 - Advisory board meeting: Summer 2025 - Share grant confirmation on MU & UWM websites (MU IT, UWM IT): Summer 2025
1 8/2025 – 12/2025	- GitHub project setup (repository, read.me, documentation) (LR): 8/2025 - Share project to open-source communities & social media (LR & GRAs): 8/2025 - Protected data/document repository setup (OneDrive) (LR&MK): 8/2025 - Initial accessibility checks (LR teach, GRAs) - IRB certifications (GRAs): 8/2025 - Environmental scan (LR&GRA): 8/2025 - Writeup of project for JCDL submission (TEAM): 8/2025 sub, conf 12/2025 - Mapping Census surveys & tables, Pubmed API structures (TEAM): 8-12/2025 - Zoom meet and greet with MU and UWM students interested in project (LR&MK): 9/2025 - Publication: What do librarians want from OGD? From survey data (TEAM) → ITAL or Information Research (both OAJs): 10/2025 - Advisory board meeting: 10 or 11/2025 - IRB for user study for Phase 3 (LR&MK lead, teach GRAs): 12/2025
2 1/2026 – 12/2026	- Flowcharts of Census surveys and mappings to data tables, API variables: 1-5/2026 - Flowcharts of PubMed structures and mappings to API: 1-5/2026 - Manually verify mappings: 1-5/2026 - Digital management of mappings: 1-5/2026 (TEAM) - Build regular expression templates (LR&MK lead, teach GRAs): 3-6/2026 - Map flowcharts to regex query templates and test (LR&MK start, teach GRAs): 3-8/2026 - Recruit users for user study: 5-12/2026 - Build additional Python structures (dictionaries (regex to surveys, surveys to variables, etc.), lookup tables) based on flowcharts and data documentation (LR, TEAM assists): 2-10/2026 - Test more complex queries (TEAM): 8-12/2026 - Work in additional data (past 3-5 years in Census, additional PubMed Entrez utilities): 8-12/2026 - Zoom meet and greet with MU and UWM students, open source community interested in project (LR&MK): 9/2026 - Publication on AI, mapping OGD for data retrieval → IPM or JASIST: 8-12/2026
3 1/2027 – 7/2027	- Advisory board meeting: 1/2027 - Python based query builder tool development (built on regex templates) (TEAM, community): 1-7/2027 - Python based query builder tool testing and documentation (TEAM, community): 1-7/2027 - Update read.me file as needed (TEAM, community): 1-7/2027 - Accessibility checks (GRAs): 1-7/2027 - User study of select surveys (2 examples of build): 2-3/2027 - User study publication: 3-6/2027 → KO or other appropriate journal - Work in additional data: 1-7/2027 - Final publications about what we did and why they might want to use it, user study → LRTS (OAJ), other journals read by professional librarians: Summer 2027