Knowledge Base for ESEA (East-and-Southeast-Asian) Traditional Music and its NLQ2SPARQL Intelligent Question-Answering System Development

Authors

Junjun Cao, Postdoctoral Researcher, Distributed Digital Music Archive and Library Lab, Schulich Music School, McGill University, Montreal, Canada

Ichiro Fujinaga, Director, Distributed Digital Music Archive and Library Lab, Schulich Music School, McGill University, Montreal, Canada

Xiaodong Fu, Head Librarian, Library of China Conservatory of Music, Beijing, China

Qianping Peng, General Manager/CTO, AgentBoosty Technology Co. Ltd., Hangzhou, China

Key Words

Schema; Shapes; Large Language Models (LLMs); Sub-graph Extraction from Ontology; Question-Answering Systems; Natural Language Queries (NLQ) to SPARQL

Abstract

The East-and-Southeast-Asian (ESEA) traditional music cultural heritage has long been underrepresented and under served in digital knowledge. We introduce an intelligent NLQ2SPARQL question-answering system, leveraging an RDF knowledge base for ESEA Traditional Music, which is part of the LinkedMusic (https://linkedmusic.ca/) project. This knowledge base focuses on the cataloging and classification of controlled vocabularies, such as music types and instruments, alongside first-hand audio-visual resources.

This knowledge base is primarily built upon ontology engineering and controlled vocabularies. The ontology reuses existing vocabularies including Wikidata, CIDOC-CRM, BibFrame, The Music Ontology and SKOS. We use Wikidata for reconciliation. Using cidoc-crm:E55_Type, the ontology incorporates a thesaurus for core classes such as bf:MusicInstrument and :MusicType, representing tangible and intangible cultural heritage, respectively. Inspired by the 3-layer (Work, Instance, Item) core elements from BibFrame, we sort the instrument entries into corresponding layers: Instrument Works relates to anthropological classification views, instrument Instances to "standardized" acoustic views, instrument items to individual physical instruments. The thesaurus employs a directed acyclic graph structure for classification. Additionally, a refined ontology snippet is incorporated as context for prompt engineering with large language models (LLMs, we use ChatGPT4o) to extract structured RDF triples from an Oriental Instruments Dictionary. This is to showcase leveraging AI for knowledge extraction.

The knowledge base is stored in OpenLink Virtuoso, and the Ontology is finalized in 2 versions: (1) an intact version for knowledge inference and data supplementation, and (2) a particular version for sub-graph extraction (refer to the following abstract). Both versions are hosted in Virtuoso.

We demonstrate **SPARQL query examples** for traditional music domain knowledge, and particularly for another 3 aspects: (1) query for specialized data types such as xsd:datetime, geographical query, blank nodes and rdf:Bag; (2) rule-based reasoning query e.g., partOf relationships; (3) ontology-based knowledge inference for data supplementation within Virtuoso by combining the ontology OWL file with RDF instance data. The natural language questions and corresponding SPARQL examples are explored in a vector database to enhance usability.

In terms of AI, the core research focuses on developing workflows for "NLQ2SPARQL" by invoking LLMs API for prompt engineering: Convert any natural language question to the SPARQL query language against the RDF

knowledge Base, which is foundational for the intelligent question-answering system.

Prior Experiment and Literature Reviews: (1) Initial experiments combined ontology with "shapes" (serving as schema for RDF database) rendered by SHACL (Shapes Constraint Language) to generate SPARQL queries, proving effective. (2) We found very few previous studies advocating straightforward use of ontology for NLQ2SPARQL; most approaches rely on matching existing SPARQL query examples. (3) An alternative method, using the concise ShEx (Shape Expressions) to support SPARQL generation, proving effective, still have limitations: (a) Excessive properties in classes can complicate shapes; (b) It can not capture and describe the triples supplemented by ontology-based reasoning. (4) Additionally, regarding a better context for NLP2SPARQL prompt engineering, ChatGPT4o gave a comparison between Shapes and Ontology to highlight their respective advantages. Above all, we applied these ideas to the ESEA Traditional Music Knowledge Base to craft a more tailored methodology.

We introduce **using ShEx to express the shapes** for our ESEA traditional music database (based on the previous research by other scholars). The core workflow at this stage contains 2 steps: First, generate VoID (Vocabulary of Interlinked Datasets) information from the given SPARQL Endpoint; Second, generate shapes information from the VoID. Both steps are done automatically.

Due to the massiveness and the dynamic, flexible and expansive nature of the ESEATM Knowledge Base's ontology, it's impractical to feed the entire ontology as context in one go. **We introduce our original method featuring "Sub-graph Extraction from Ontology"**:

(1) **Ontology Segmentation**: Using the aforementioned particular ontology version for sub-graph extraction, we divide the OWL file, which mainly contains rdfs:domain, rdfs:range, rdfs:label, rdfs:comments etc. and trims off other semantic assertions, into evenly separate segments. (We regard an ontology as vocabularies list, so it can be segmented in a flexible and manageable way.)

(2) **Entity Extraction**: For a given natural language question, relevant entities (classes and properties) are extracted from the OWL file through prompt

engineering with LLMs. If the class of an instance is not explicit, its IRI and rdf:type are retrieved using SPARQL, supported by fuzzy queries for robustness.
(3) **Sub-graph Assembly**: Using SPARQL, we retrieve the ontology, which is structured with domains and ranges for properties, enabling it to be coherently viewed as a general graph. To address the efficiency limitations of LLMs (e.g., ChatGPT4o), we designed a SPARQL CONSTRUCT query to extract a sub-graph from this general graph. The parameters for the CONSTRUCT query are the identified classes and properties determined in the previous step.
(4) **Shape Snippet Extraction**: Extract corresponding shapes snippet based on the classes extracted in Step (2). Additionally, these shapes snippets also enhance the completeness of the sub-graph extraction approach.
(5) **SPARQL Query Generation Based on the Sub-graph**: Prompt LLMs to generate SPARQL queries from the sub-graphs. To further improve accuracy, we employ 3 reinforcement methods: (5.1) Supplement the context by retrieving a relevant SPARQL query example from the aforementioned vector database with embedding-based similarity search; (5.2) Verify the SPARQL query using the corresponding shapes snippet; (5.3) Iteratively refine the SPARQL query based on feedback from the SPARQL endpoint to correct any errors.
(6) **Retrieval-Augmented Generation (RAG)**: (6.1) Prompt LLMs to explain the generated SPARQL query and the semantic meaning of its corresponding ontology snippet. (6.2) We may also prompt LLMs to reflect upon the result, determining whether it aligns with the intention of the original question. (6.3) Leverage the fact that parts of the knowledge base instances are reconciled from Wikidata, which has been incorporated in the training of LLMs. This enhances the RAG process, yielding more comprehensive and accurate results.

Issues and Prospective Refinements Strategy: While the process is robust, it is time-consuming, leaving room for efficiency improvements and simplification. For instance, incorporating negative assertions to clarify the ontology could reduce reliance on ShEx. Additionally, further exploration of combining ontology with SHACL may enhance the process. Improving user-LLM interaction procedures could also increase precision in future question-answering interfaces. However, the increasingly hierarchical structure of classes and

properties poses a challenge to the accuracy of the current sub-graph extraction method, which requires careful consideration for future refinements.

Epilogue: We advocate an **ontology-driven NLQ2SPARQL system to emphasize the synergy between symbolic AI (knowledge graphs) and LLMs.** Ontologies enhance data interoperability, bridging user intent with LLM understanding. The ontology-based knowledge inference and SPARQL generation based on the ontology snippet also relieve the burden of LLMs operation for knowledge service. Ontology will also constantly play an essential role in the knowledge expression, automatic extraction, and intelligent retrieval for both tangible (e.g., instruments) and intangible (e.g., music types/forms) cultural heritages. This approach underscores the role of ontology in preserving and disseminating ESEA traditional music while paving the way for innovative AI applications.

Contact: Junjun Cao

Tel: (+1) 438 596 0579

Email:alienmusedh@gmail.com