Semantic Search for Self-describing Scientific Data Formats

Chenxu Niu[1], Wei Zhang[1], Suren Byna[2], Yong Chen[1]
[1]Texas Tech University, [2]Lawrence Berkeley National Laboratory
{Chenxu.Niu,X-Spirit.zhang}@ttu.edu, sbyna@lbl.gov, Yong.Chen@ttu.edu

***Abstract*—It is often a daunting and challenging task for scientists to find datasets relevant to their needs. This is especially true for self-describing file formats, which are often used for data storage in scientific applications. Existing solutions extract the metadata and process search queries with matching search keywords in metadata via exact or partial lexical match approaches. However, they are hindered by an inability to capture the semantic meaning of the content of the metadata, and are therefore precluded from performing queries at the semantic level. We propose a novel semantic search solution for self-describing datasets, which captures the semantic meaning of dataset metadata and achieves search functionality at semantic level. We have evaluated our approach and compared against the existing solutions. Our approach demonstrates efficient semantic search performance.**

I. Introduction

Large-scale scientific applications typically store their experimental, observational and simulation datasets in self-describing data formats, such as HDF5 [1] and NetCDF [2]. Finding relevant scientific datasets is critical to data discovery, data reuse, and data management. However, scientists are often limited by the capabilities currently available for finding relevant desired datasets. They often find it is a daunting and time-consuming task for them to locate datasets relevant to their needs.

The metadata in self-describing data formats provides users with sufficient descriptive information about the scientific dataset and therefore is prevalently used for finding required data. The metadata can be seen as a collection of attributes. Each attribute can be represented as a key-value pair, where the *key* represents the attribute name of the metadata and the *value* represents the attribute value. Most attribute names are of the string type, and most attribute values are of either the string or the numeric number type. Existing metadata search solutions tend to extract the metadata attributes and match the attributes with search keywords.

However, in the existing dataset search solutions for self-describing file formats, search functionality was only achieved by conducting exact or partial lexical match on the metadata attributes, without considering the semantic meanings of the metadata. With such metadata query and data search capability, the result is always limited by the exact naming of metadata attributes (i.e. the naming of the "keys"). Such an approach often needs scientists to navigate through dataset schema and understand the metadata attributes to be able to query and search interested datasets. It also results in a barrier for scientists to mine a large number of datasets as these datasets often have inconsistent metadata attributes and naming schema (e.g. two datasets can describe the exactly same observation but with two different metadata attributes as "speed" and "velocity", respectively). Even though the semantic search in natural language processing and web domain has been well studied, these semantic solutions in information retrieval and semantic web are not directly applicable (or too complex in many cases) to scientific datasets because metadata attributes in the self-describing data formats are in the form of key-value pairs, not in sentences, paragraphs or articles. The objective of this research investigation is to validate the hypothesis that semantic meanings of metadata can be used to optimize dataset search solutions and to design and develop an efficient semantic search solution for scientific datasets.

II. Methodology

Figure 1 shows an overview of our proposed approach. There are three key components in our design: metadata semantization, semantic index construction, and semantic search processing. These components are designed to provide metadata semantization, build semantic metadata indexes and provide semantic search functionality.

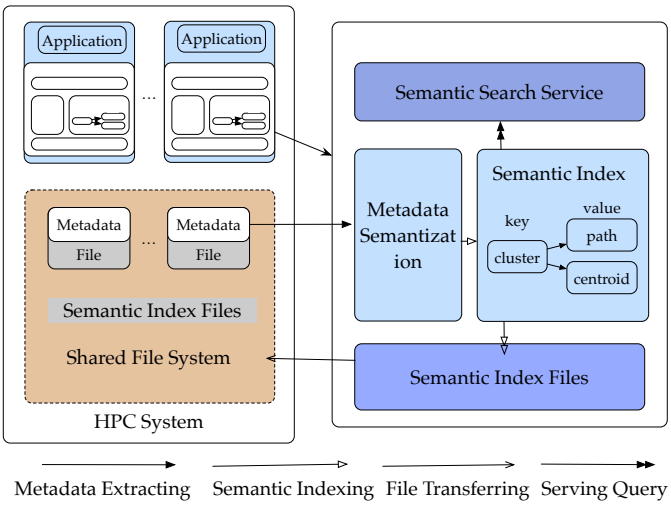

Figure 1: Overview of semantic search for self-describing scientific data formats

We illustrate the procedure with a typical self-describing format example, HDF5 dataset. HDF5 file format is designed

to store different types of data within the same file. For example, one group may contain a set of datasets comprised of integer (numeric) and text (string) data. Or, one dataset can contain heterogeneous data types (e.g., both text and numeric data in a single dataset). It is one of the data formats most frequently used by scientific applications. HDF5 uses a "file directory"-like structure that allows users to organize data within the file in many different structured ways, as they might do with files on their own computer. The HDF5 format also allows the embedding of metadata, making it self-describing. Within each HDF5 file, there are three main elements of the HDF5 data model: datasets, array-like objects that store your numerical data; groups, hierarchical containers that store datasets and other groups; and attributes, user-defined metadata that can be attached to datasets and groups.

Given a collection of HDF5 files, we extract metadata from the objects in the files first, including groups and datasets. Different from existing metadata search solutions, our approach conducts metadata semantization to represent each attribute by a multi-dimensional semantic vector with word-vector mapping. Word-vector mapping is a mapping of words into vectors of real numbers using a neural network, a probabilistic model, or a dimension reduction on a word co-occurrence matrix. It can be achieved by using various word-vector models available such as word2vec [3], [4], GloVe [5], and fasttext [6]. After the mapping conversion, we utilize the cosine distance between two semantic vectors to describe the semantic similarity. Depending on the cosine distances among vectors, our approach creates clusters for each dataset.

After the cluster construction, we build a semantic index for each dataset to maintain the correlation between semantic features and the files. For each index, the key is the cluster, while the value is the path of the dataset. The semantic indexing mechanism provides an efficient semantic search over large scientific datasets with millions of attributes. When serving search functionality, the keyword is converted into a semantic vector to determine the similarity between the vector and the clusters. Once the vector is located in clusters, the index is used to retrieve the cluster and the path of the corresponding file. Following the semantic index mechanism, it is handy to find and return the semantically relevant datasets as results for users. (Please see the poster for the illustration of the method too.)

III. Evaluation

We collected three sets of real-world HDF5 files: National Snow and Ice Data Center (NSIDC), Sea Ice Altimetry Data Center (SIADC), and the Baryon Oscillation Spectroscopic Survey (BOSS), and used in our current evaluation. We tested the index construction performance. The results show that the index construction is proportional to the dimensions of the vectors. We further tested the performance of queries. The results indicate that our approach is scalable, and the query throughput outperforms the existing MongoDB-based solution.

Compared with three existing metadata search solutions, including MongoDB-based method (metadata stored and queried on a separate MongoDB), MIQS [7], and a content-based information retrieval method, we measured the search accuracy by calculating query hits percentage and recall accuracy. Query hits percentage describes the effectiveness of search solutions. We first randomly selected keywords from science data website, related papers and Kaggle [8] to perform queries. We then calculated the query hits percentage for all different methods. The results show that our approach achieved about 2X better than the other solutions. After that, we choose the overlap of the valid keywords and perform queries. We further analyzed the recall accuracy, which is the rate of the number of retrieved relevant datasets and the whole relevant datasets. As shown in Figure 2 and Figure 3, our approach outperforms these solutions in terms of semantic precision and recall accuracy.

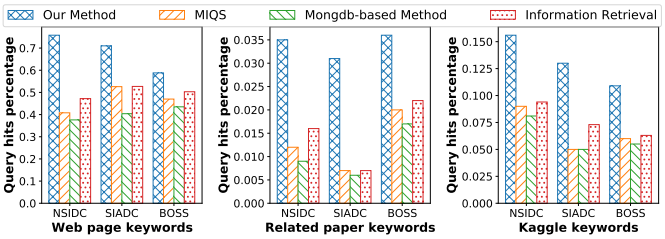

Figure 2: Query hits percentage comparison

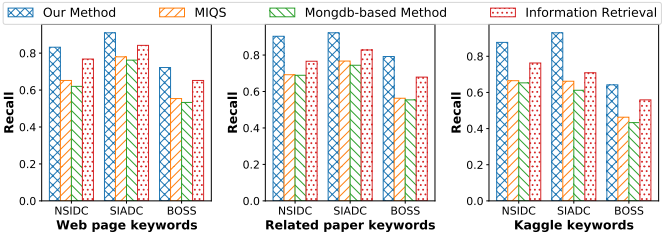

Figure 3: Recall accuracy comparison

IV. Conclusion & Future Exploration

Existing metadata search solutions for self-describing data formats process metadata queries without considering semantic meaning of the metadata and queries. In this research, we explore a methodology to achieve semantic search for self-describing datasets. We demonstrate the potential of this approach with an evaluation with real-world scientific datasets. Our approach outperforms existing solutions in semantic metadata search in terms of semantic precision and recall accuracy. We will continue improving the efficacy of the semantic search methodology by optimizing the metadata semantization and cluster construction.

References

[1] M. Folk, A. Cheng, and K. Yates, "HDF5: A File Format and I/O Library for High Performance Computing Applications," in *Proceedings of supercomputing*, vol. 99, 1999, pp. 5–33.

[2] R. Rew and G. Davis, "NetCDF: An Interface For Scientific Data Access," *IEEE computer graphics and applications*, vol. 10, no. 4, pp. 76–82, 1990.

[3] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient Estimation of Word Representations in Vector Space," *arXiv preprint arXiv:1301.3781*, 2013.

[4] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed Representations of Words and Phrases and Their Compositionality," in *Advances in neural information processing systems*, 2013, pp. 3111–3119.

[5] J. Pennington, R. Socher, and C. D. Manning, "GloVe: Global Vectors for Word Representation," in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 2014, pp. 1532–1543.

[6] A. Joulin, E. Grave, P. Bojanowski, and T. Mikolov, "Bag of Tricks for Efficient Text Classification," *arXiv preprint arXiv:1607.01759*, 2016.

[7] W. Zhang, S. Byna, H. Tang, B. Williams, and Y. Chen, "MIQS: Metadata Indexing and Querying Service for Self-describing File Formats," in *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, 2019, pp. 1–24.

[8] "Kaggle," https://www.kaggle.com/, 2020.