2024 IEEE/ACM International Conference on Big Data Computing, Applications and Technologies (BDCAT) | 979-8-3503-6730-0/24/$31.00 ©2024 IEEE | DOI: 10.1109/BDCAT63179.2024.00044

MAGELLAN: Enabling Effective Search Over Voluminous, High-dimensional Scientific Datasets

Federico Larrieu*, Tyson O'Leary*, Sangmi Lee Pallickara*, and Shrideep Pallickara*
*Department of Computer Science, Colorado State University, Fort Collins, CO, USA
Email: flarrieu, tyson.oleary, sangmi, shrideep@colostate.edu

***Abstract*—As high-dimensional, voluminous datasets continue to become available, they present opportunities for users to perform richer explorations that lead to insights. Most explorations are however limited by the query semantics enforced by the underlying storage system. This precludes identification of connections that exists within and across datasets. This study describes, MAGELLAN, a system that is designed for richer, iterative explorations that allow users to explore connections within and across datasets. Our methodology combines aspects of ontologies and metadata to support analysis that are domain informed and statistically richer. Our performance benchmarks demonstrate the suitability of our methodology to inform explorations interactively and at scale.**

***Index Terms*—knowledge graphs, ontology, semantic web, trees**

I. INTRODUCTION

Datasets continue to be made available in several domains. These data often encapsulate phenomena from diverse domains. The data are often structured or semi-structured with the schema either rigorously enforced (as in relational stores) or implicitly inferred alongside lax enforcement (document or NoSQL stores). The datasets are expressed as a collection of records with values associated with individual data items. Variable names within these records are chosen to signify what it represents, and the data values themselves may be numeric, categorical, or ordinal.

Users interested in performing analysis must go through a data explorations phase that involves 3 broad steps. This includes identifying (1) datasets that may contain records of interest. (2) variables that might potentially be of interest, and (3) other ancillary datasets to be considered during analysis.

Data storage frameworks – be it relational, document, or NoSQL stores – take a more data-centric and necessarily, analysis-agnostic approach to storing the data. In this view, the datasets are simply a collection of variables with schemas that enforce type constraints. This view of the dataset provides a separation of concerns for the datastore administrators by simplifying how data are managed, indexed, and stored. Further, each variable is treated in a standalone fashion with little information about how they are related to each other. The connections between these variables are latent even to those who formulate queries or maintain these datasets within datastores. These problems are exacerbated as new datasets (and variables) continue to become available.

Users performing data explorations are however stymied by this approach. They are unaware to connections within the datasets. A variable, no matter how explicitly named, can have limited value and are often encoded. For example, a dataset that measures lead concentrations in bodies of water would benefit from information identifying other chemicals that are known to be reactive agents that can exacerbate the hazards of lead. Similarly, within the datasets, relying only on enforcement of schemas and type constraints can be limiting and difficult at scale. This is because researchers' analyses often span multiple datasets, layering diverse data sources to enrich their findings. These issues are particularly pressing in scientific domains.

The crux of this study is to design a framework, codenamed MAGELLAN, that facilitates richer and more comprehensive data exploration over highly heterogeneous scientific data. We postulate that in order to extract significant value from the data, searches both *for* and *in* the datasets must be performed.

A. Challenges

There are several challenges to enabling effective searches over scientific data collections. These include:

1) The data we consider are high dimensional. Data storage frameworks focus on type/schema enforcement for individual variables, and the data storage administrator may set up ancillary data structures such as indexes for some of these variables.
2) The datasets describe individual variables, but not necessarily the connections between them. These connections are often latent and implicitly available only to domain experts.
3) Because the data are voluminous, users launching custom jobs to build expansive views of the data might result in multiple, repeated sweeps of the data triggering both disk and network I/O.

B. Research Questions

We explore the following research questions:

RQ-1: How can we leverage domain knowledge to support effective search over datasets?
RQ-2: How can we support rapid explorations of features within a dataset?
RQ-3: How can we support declarative queries that allow explorations of the dataspace?

C. Approach Summary

Our methodology supplements existing datasets with ancillary information that facilitates data explorations, provided by ontology and our metadata tree. We define this additional information as the knowledge graph. The knowledge graph sits alongside the raw dataset and entails no modifications to the original data, or the indexing structures maintained by the data

979-8-3503-6730-0/24/$31.00 ©2024 IEEE
DOI 10.1109/BDCAT63179.2024.00044

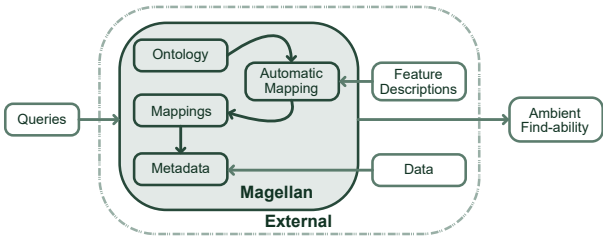


Fig. 1: **Provides translation from an ontology to underlying metadata, by automatically mapping of concepts to variables using variable descriptions and natural language processing. Additionally, the metadata tree is incrementally constructed via a stream of raw data.**

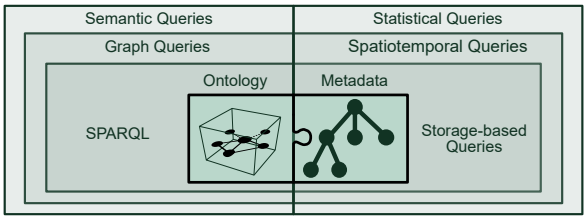


Fig. 2: **The MAGELLAN knowledge graph allows search for and in the data. The knowledge graph supports a federated query infrastructure encompassing semantic, graph characteristics, data store specific, and statistical queries. Together, this allows users to search expressively and alleviate blind spots.**

store. This allows our methodology to be independent of the mechanisms – relational databases, document storage, graph storage, etc. – used to manage the raw dataset.

Our knowledge graph encapsulates information about connections between variables within and across datasets, and richer metadata derived from each variable. The knowledge graph encapsulates two key pieces of information: the ontology and metadata. Ontology encapsulates information about concepts and relationships via a rich set of edge labels that exist between them. Ontologies are powerful for three key reasons: (1) they encapsulate domain knowledge in ways that are amenable to traversal and querying rigorously and programmatically, (2) ontologies represent agreement within the community about key concepts alongside rich contextual information that are conducive to human exploration, (3) their representation as a multigraph alongside specification of rules and heuristics are themselves amenable to inferencing and assertions. Our methodology overview is depicted in Fig. 1.

The metadata tree encapsulates rich statistical information for individual variables. This includes per-variable summary statistics and kernel density estimates, and pairwise intra-dataset covariance and correlation. Our metadata tree complements queries supported by the data store, informing queries that can be executed with finer grained analysis. For instance, the metadata tree can be used to perform multi-dimensional queries, formulate queries while understanding the number of records likely to be included or excluded (via our kernel density estimates), aggregation queries and conditional queries.

Our knowledge graph connects concepts in the ontology with the metadata associated with individual datasets. Making connections between the ontology and the metadata tree allows users to search *for* and *in* the data, as illustrated by Fig. 2. The explorations are transformed because the user can understand connections between concepts, access richer descriptive informational variables that reflect current domain knowledge and explore individual variables at the aggregated scale. This allows us to surface other connections within the data set. We also support the concept of ambient findability, where a researcher can explore related concepts and variables.

Rather than support imperative queries where every aspect of the query is explicitly specified, we rely on declarative queries. This allows us to support richer, expressive, and iterative explorations of the entire data space. Our queries leverage a mix of the capabilities available for querying ontologies (concepts, relationships, etc), statistical probing of the data space, and storage-specific queries such as those based on NoSQL, graphs, and relational algebra.

Our methodology is amenable to continuous, incremental updates to the data space i.e., new variables or records may be added to existing datasets and entirely new datasets may be added. Our knowledge graph is designed to minimize the number of updates that need to be performed. This allows our framework to cope with evolution of the data space at scale.

D. Paper Contributions

In our study we describe our methodology and performance benchmarks to facilitate effective searches over voluminous, high-dimensional data. Our specific contributions include:

1) Our methodology is independent of the underlying framework used for storing data. MAGELLAN only necessitates a stream of data as input.
2) MAGELLAN allows a richer exploration of the data space that goes beyond queries supported by the underlying data storage framework. The framework allows users to identify connections between variables that span data sets alongside richer contextual information associated with individual variables.
3) Our methodology is amenable to continuous updates (and additions) of datasets in the ecosystem.
4) Our methodology allows users to make richer explorations within/across datasets while supporting iterative explorations of the data space.

E. Paper Organization

The remainder of the paper is organized as follows. Section II outlines background and related work. Section III describes several key aspects of our methodology and system architecture. Section IV includes a discussion of our performance benchmarks and profiling. Finally, section V outlines our conclusions and future work.

II. Related Work

Ontology-based data access (OBDA) systems allow users to access external databases through a conceptual domain view, which directly aligns with our approach in MAGELLAN. There are differences in the underlying data stores that are used, and the mapping generation and storage techniques. There are applications using semantic technologies, particularly OBDA, to address the data variety challenge in complex, distributed, and heterogeneous environments [1], [2]. In which automation and the translation of user queries into optimized SQL or SPARQL queries over distributed databases is the primary focus. Rather than leveraging pre-existing ontologies in their raw form, a common approach is to define and construct a custom ontology that is based on both the semantics in the database and augmented with notions from foundational and related domain ontologies [3], [4]. This differs from MAGELLAN as we aim to leverage already existing ontologies.

One of the necessary components of using ontologies for data exploration involves mapping concepts to feature space. Managing OBDA systems also requires significant overhead. Specifically when altering mappings. MASTRO uses description logics to manage ontologies, ensuring efficient query answering even in large-scale data environments [5]. Previous work has focused on managing and debugging ontologies and mappings to ensure that OBDA systems remain efficient and adaptable [6] . Approaches for annotating data involve natural langauge processing and computing semantic similarity. One approach explores ontology-based annotations and semantic relations in large-scale epigenomics data [7]. Another approach, which is a tool used in this paper, is the NCBO annotator. This tool leverages mgrep and semantic expansion to map annotated text to concepts within an ontology [8].

There are many different techniques to construct and utilize knowledge graphs. Chavas et al. [9] presents a systematic workflow for constructing knowledge graphs from existing information systems in research-performing organizations. These utilize systems such as R2RML and YARRRML.

Lin et al. [10] propose a semantically enhanced catalogue search model to extend existing catalogue services to allow more searchable parameters without changing the underlying metadata database. In particular, the focus is also on a semantic query for collection-level discovery search, and a catalogue service for granule-level inventory search. MAGELLAN, on the other, targets collection level and feature level search.

Tree structures are able to effectively index and support search. There are many studies surrounding the utilization of tree structures for spatiotemporal data. In one case, Delta-tree and Delta+-tree index structures are proposed to efficiently index and search high-dimensional data in main memory. Efforts have also leveraged PCA analysis to create a multilevel tree, where each level represents the data space at different dimensionalities [11]. Another, proposes a set of algorithms that use a quadtree index to enable real-time generalization of large point datasets [12].

One aspect of our data is the spatiotemporal dimension. Here, each data item includes both the location and the time at which the observation was captured. In MAGELLAN, we focus primarily of providing a conceptual search layer. On the other hand the spatiotemporal attributes are not used as concepts but rather indexed in the underlying metadata tree structure. There are approaches that leverage geographical semantics for the conceptual layer. One approach, works on expanding geographic queries to improve the performance of geographic information retrieval systems [13]. It accomplishes this by augmenting the geospatial part of the query by adding related geographic terms or entities or by incorporating synonyms and related terms that are contextually relevant. Another, involves expanding a query by deriving its geographical query footprint for queries that involve spatial terms [14]. AnnoTerra searches on NASA resource catalogs using earth science concepts and relationships [15]. Additionally, in a different approach, queries to address questions related to the dimensions of "when," "what," and "where" leveraging spatiotemporal ontology [16]. Another approach leverages concepts in natural language queries to find data sources that would be useful for the user [17].

Frameworks have explored support for ad hoc queries [18], [19] over spatiotemporal data alongside content dissemination [20] including in the context of peer-to-peer systems and grids [21]. These are complementary to the MAGELLAN queries .

III. Methodology

Our methodology encompasses a set of key tasks to accomplish effective search over voluminous datasets. This includes (1) leveraging ontologies as a stand-in for domain knowledge, (2) effective metadata generation alongside statistical information at different spatiotemporal scopes, (3) constructing a knowledge graph through a lightweight, information-based connection between the ontology and metadata tree, (4) support for declarative queries that leverage SPARQL, statistics, traditional database queries, and graph properties, (5) ambient findability that leverages semantic expansion and graph characteristics, and (6) the ability to support continuous, incremental updates to the knowledge graph.

A. *The* MAGELLAN *Knowledge Graph* [RQ-1]

The knowledge graph in MAGELLAN, visualized in fig. 3, integrates the semantic layer (ontology) with the data layer (variables). We define the knowledge graph as a property graph $G = (E, R)$ with entities $e \in E = \{E_1, ..., E_n\}$ from the ontologies and metadata graph. The relations R inherit transitive relationships defined in the ontology and metadata graph and are further enhanced based on the integrated network structure.

Searching *for* the data can be a difficult process when dealing with a diverse set of high dimensional datasets. Often, the name of variables are encoded and unclear.

MAGELLAN supports search *for* the data by integrating a semantic layer in addition to the data operations. This semantic layer in our case is the Agriculture and Forestry Ontology (AFO), which provides concepts and context surrounding the

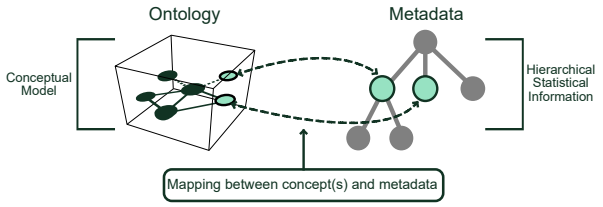


Fig. 3: **The MAGELLAN knowledge graph fuses the ontology graph with the metadata tree allowing information linkage between concepts and metadata. The metadata tree maintains hierarchical statistical information, about the connected variable, that is amenable to queries and incremental updates.**

agricultural domain. If a user is interested in looking at a concept, MAGELLAN will provide the columns of interest along with the following contextual information: (1) the dataset it resides, (2) other concepts of interest (semantic expansion) and the centrality of their relationships, and (3) the importance of the concept within the ontology.

Searching *in* the data involves uncovering patterns, trends, and insights that may not be immediately apparent. MAGELLAN aims specifically to help researchers analyze relationships between variables, examine correlations, and understand distributions. Furthermore, this is paired with the ability to specify space, time, or space and time together.

MAGELLAN supports operations for searching *in* the data using the metadata tree structure. This structure is optimized using spatiotemporal indexes, for efficient data search.

*B. **Knowledge Graph: Leveraging Ontologies*** [RQ-1, RQ-2]

Accounting for domain-knowledge is a key aspect in MAGELLAN. Knowledge of data alongside its context is beneficial for users as they search for datasets and variables that are relevant to their task. Results of data analysis are often more interesting when many variables are combined and explored together. To support this, knowledge of relationships across variables provides the necessary underpinnings for users to expand their analysis. We provide domain knowledge via the use of ontologies that rigorously define *concepts* and the *relationships* that form the basis of connections between them. The logical representation of an ontology consists of nodes and edges in a multigraph structure, meaning concepts can be related to any number of other concepts and any given pair of concepts can have any number of relationships between them.

MAGELLAN utilizes an ontology as the foundation for finding useful and relevant data. The ontology sits alongside data collections informing search and retrieval operations. Choosing the correct ontology is important as it informs how MAGELLAN supports searching for relevant data. There are a large variety of ontologies available across various domains. The Linked Open Data Cloud provides vast amounts of semantic knowledge and is a good starting point for locating an ontology for the desired domain of study.

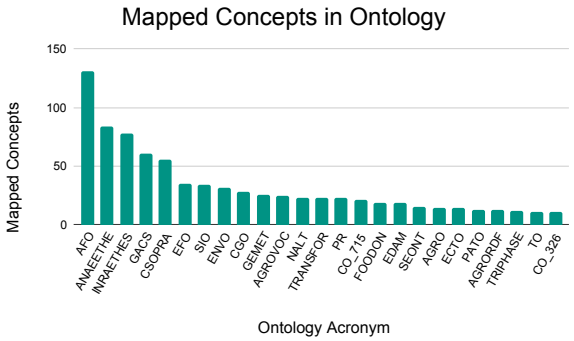


Fig. 4: **Distribution of concept connections across the top 25 ontologies, illustrating their coverage of the agricultural scientific datasets in our use case.**

The scientific datasets that we consider explore dimensions of mitigation and adaptation to climate change in agricultural settings; specifically, relating to soil organic carbon. We use AgroPortal [22]: a portal that hosts a wide variety of agricultural ontologies in agronomy, food, plant sciences and biodiversity. Similar portals exists for other scientific domains —our methodology would work with other ontologies as well. Next, we leveraged their natural language processing recommender that assigned evaluation scores (relevance) for individual ontologies. Crucially, this precludes the need for creating complex local data models to evaluate ontologies that best fit our data.

Our evaluation encompassed scoring 1144 concepts, across 175 ontologies, based on their evaluation score. The counts of concepts linked per ontology is depicted in Fig. 4. With an average evaluation score of 0.4426, and 131 recommended concepts, the Agriculture and Forestry Ontology was selected. There is nothing preventing the use of multiple ontologies with Magellan, but the AFO will be used alone in our experiment for tractability. AFO is based on the Agriforest thesaurus maintained by the Viikki Campus Library, University of Helsinki. This allows equivalent and differing concepts, and their links to more general concepts, to be explicitly described. Notably, such a combination of ontologies can be used for describing resources especially in the domain of agriculture, forestry, veterinary medicine, food science, environmental science and biology. The ontology contain 32,848 nodes and 203,391 edges. The top 20 linked concepts in AFO, based on evaluation scores, are shown in Table I.

Concept Mapping: MAGELLAN employs an RDF graph to map dataset variables to domain-specific concepts, supporting scalable and dynamic updates. This graph-based approach aligns well with semantic web standards such as RDF, SPARQL, and OWL, enhancing interoperability and flexibility compared to traditional YAML-like mapping languages. Additionally, MAGELLAN ensures that relationships can be easily updated and expanded as new ontologies or concepts are integrated. It ensures this by storing concept-variable

TABLE I: Top concepts linked in AFO with evaluation scores.

Concept	Evaluation Score
layout	1.000000
crop_rotation	1.000000
information	1.000000
animal_species	1.000000
being	1.000000
practice	1.000000
tillage	1.000000
identity	1.000000
potassium	0.987544
equipment	0.969751
hay	0.957295
depth	0.944840
harvest	0.937722
spring	0.935943
crops	0.935943
frequency	0.934164
seed	0.934164
leaching	0.923488
SURFACE	0.923488
litter	0.923488

mappings in an RDF graph that is stored using Apache Jena and backed by a persistent data store, TDB2.

Utilizing Protege [23], an open-source ontology editor, we specified the classes and properties for our internal mapping structure. The structure encompasses several key classes and properties to facilitate data organization and retrieval between the ontology and metadata tree. The primary classes, blueprints for instances in the graph, include Dataset, Variable, Concept, and Ontology. Dataset and Variable classes are used to match top-level indexes in the metadata tree. Concept and Ontology classes map concepts to ontologies for search. To establish relationships among these classes, MAGELLAN's ontology introduces several new object properties:

- `annotates variable` - links domain concepts to specific variables
- `belongs to dataset` - denotes dataset a concept relates to
- `belongs to ontology` - denotes overarching ontology a concept is part of

MAGELLAN integrates and extends concept-variable mapping by leveraging the NCBO Annotator framework [23], which employs Natural Language Processing (NLP) techniques combined with semantic expansion to annotate dataset variables with relevant ontological concepts. The NCBO Annotator uses mgrep (Multi-Granular Entity Recognition) [24] to perform precise string matching on variable names, descriptions, and metadata, identifying potential matches from a vast set of ontologies.

Semantic expansion is critical for mapping variables to concepts within the ontology, as it establishes contextual relationships between the original dataset and user annotations. It expands vocabulary used to describe an object in order to better represent its meaning [25]. MAGELLAN implements its own semantic expansion techniques by leveraging preexisting relationships that are in AFO, and defined by SKOS (simple knowledge organization system). SKOS is a common data model for sharing and linking knowledge organization systems via the web. Relationships used in semantic expansion include: (1) broader, (2) narrower, (3) related match, and (4) exact match. We also support expanding concept searches using graph-based breadth first search. Finally, these semantic expansion queries are executed through SPARQL.

The RDF mapping graph can be queried and modified dynamically using SPARQL, providing a more flexible and responsive mapping solution that evolves with the data and ontological landscape. MAGELLAN facilitates dynamic integration of diverse datasets and conceptual models. This allows MAGELLAN to handle a large number of ontologies.

C. *Knowledge Graph: metadata generation* [RQ-1, RQ-2]

While the ontology is useful when searching *for* datasets and variables to analyze, the metadata tree structure supports searches *in* the datasets. The metadata tree encapsulates statistical information about variables, supporting efficient data retrieval, analysis, and visualization. Statistical information includes mean, variance, min, max, covariance, and density. It provides a rich query interface for locating data based on time and space. It also leverages the inherent indexing to provide rapid exploration.

Tree Construction: Our approach to metadata generation utilizes a hierarchical tree structure organized as root/dataset/variable/temporal-block/spatial-block. All child nodes under the "dataset" node exclusively contain metadata pertaining to the dataset. The scope of a subtree is constrained by the path from the highest common ancestor node to the root node. For instance, a subtree under a specific variable encompasses metadata related to that variable within the dataset. The metadata structure includes temporal and spatial blocks. The temporal blocks are specified using a range (e.g., 1/1/2023 - 12/31/2023), shown in Fig. 5.

To specify spatial range, we leverage a quadtree data structure that allows specification of nested spatial coverage with configurable extents for spatial indexing. A leaf node then holds the summary statistics calculated for data that corresponds to the path defined by its spatial extent.

One of the core strengths of our approach is the ability to incrementally update the metadata tree as data arrives. To achieve this, we use Welford's method [26]: a method for calculating summary statistics in an online, incremental fashion. This allows for efficient incremental updates to the mean μ_n and variance σ_n^2 as new data points are added. This is especially important in scenarios where data size is large and continually evolving because such updates effectively reduce computational workloads and summarize distribution of data.

We also track covariance between all pairs of variables within a dataset in our metadata tree. A similar online approach [27] is employed for incremental pair-wise covariance calculation that builds on Welford's method. The covariance scores are organized in the tree in the same way as other statistics. Using these covariance scores, the count of records, and the variances of the pair of variables, the Pearson correlation coefficient can be calculated interactively.

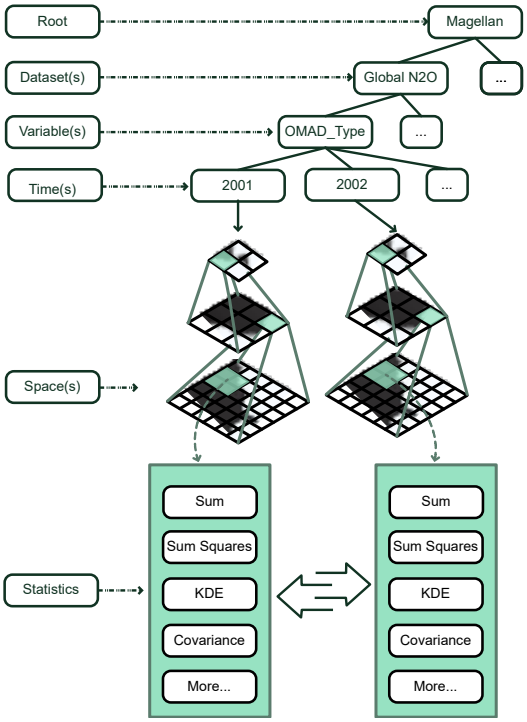


Fig. 5: **Hierarchical structure for complex datasets: optimized for efficient metadata searches, especially where data is conditional on factors like time and location.**

D. Enhancing the Knowledge Graph [RQ-3]

We enhance the knowledge graph by supplementing it with measures of graph importance, as shown in Fig. 6. In graph theory, centrality analysis is broadly used to rank nodes according to their position in the network structure. In our study, we employ betweenness centrality and PageRank to assign weights to the nodes in the knowledge graph in order to prioritize paths and subsets within the graph.

Betweenness centrality is measured based on the number of shortest paths that pass through a vertex. In our knowledge graph, betweenness centrality analysis helps identify critical connections between nodes that change the flow of information in the graph. We calculate all shortest paths in our knowledge graph for all beginning and ending nodes $k, j \in E$. If $P(k, j)$ denotes the total number of shortest paths between k and j, and $P_e(k, j)$ specifies the number of shortest paths that pass through e, the betweenness score $bc(e)$ is calculated using eq. (1) where n denotes the number of entities in the knowledge graph and $P(k, j)$.

$$bc(e) = \sum_{k \neq j, e \neq k, e \neq j} \frac{P_e(k, j)}{P(k, j)} \cdot \frac{2}{(n-1)(n-2)} \qquad (1)$$

PageRank centrality (or eigenvector centrality) represents the likelihood that a trajectory, which randomly follows links, arrives at any particular node in the graph. This centrality not only classifies the level of influence of a node but also ranks

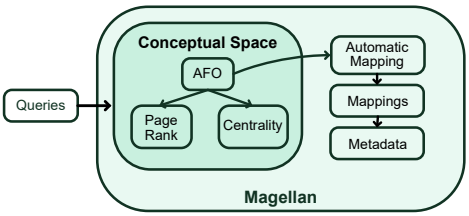


Fig. 6: **The Ontology component includes the following subcomponents: the ontology provides concepts to be mapped via NLP and Semantic Expansion, (2) mappings are stored in a structure for translation, (3) node importance and edge centrality are pre-computed and stored in hash maps, (4) providing the overall conceptual search space.**

the indirect possibility of influencing the entire graph. In our knowledge graph, PageRank centrality classifies the nodes that have a high probability of appearing as part of a path from the root and significantly influence query evaluations.

The page rank and betweenness centrality are stored in maps. The page rank for a concept can be indexed using the concept URI as the key. The betweenness centrality of an edge between two concepts can be indexed using the first concept's URI followed by the second concept's URI as the key.

There are overheads associated with computing these scores, but because concept-evolution occurs at much slower timescales compared to data arrivals, these scores need to be recomputed only at very long periodic intervals. Further, the computed scores serve as the initial starting point during recalculation; this warm starts the iterative computation process which allows these calculations to converge much faster.

E. Support for Expressive Queries [RQ-3]

MAGELLAN leverages hierarchical data structures and ontological representations to support queries. Our metadata tree structure provides a rich set of querying capabilities that allow for exploration across dimensions alongside indexing on space and time. The ontological representations allow for conceptual queries to explore the surrounding hierarchy or relationships to other concepts. For example, if a user is interested in the concept of *soil moisture* then variables associated with it directly (e.g., precipitation) or through semantic expansion (e.g., evapotranspiration) will be retrieved.

Queries in MAGELLAN are declarative in the sense that users need not imperatively specify every aspect of the query or the associated bounds. The queries support wildcards and relax fuzzy bounds alongside the semantic expansion and contextual grounding that they provide. In particular, users define concepts and time, and spatial parameters are provided by mapping frameworks.

Ontology Based Search & Exploration: MAGELLAN supports richer, concept-based queries, that leverage the direct mappings generated using NLP techniques and semantic expansion through relationships encoded in the ontology graph. This allows for a solid declarative query framework foundation, where concepts are automatically translated to retrieve

data from our metadata tree. We take this one step further by enriching the ontology with node centrality and importance. Providing grounds for data exploration by starting with what matters. The graph can also be traversed by hierarchical relationships and common graph traversal algorithms. This can provide semantic insight of where the concept belongs and also concepts for which we are missing data. The integration of this conceptual layer ensures that the semantic context of data is preserved and accessible throughout the exploration process.

Query Structure: The underlying query system for the tree structure utilizes a flexible wildcard query system, which forms the foundation for other query types. In which queries can be executed across multiple dimensions: dataset (D), variable (V), time (T), and spatial keys (S). This system employs a notation, $Q(D, V, T, S)$, to define query structures, where each parameter can be specified explicitly or set as a wildcard ($*$) to include all possibilities within that dimension. For instance, a query such as $Q(*, V, *, S)$ targets all datasets and years for specified variables and quadkeys, demonstrating the system's adaptability in accommodating various analytical needs. Quadkeys are also acted on consecutively, thus providing shorter quadkeys will allow for aggregations in the z spatial dimension.

Aggregation Queries: The primary goal of node aggregation is to compute summary statistics across various dimensions (time, space, or both) efficiently by combining the metadata from relevant subtrees. The aggregations are computed on the results provided by the wildcard query system $Q(D, V, T, S)$, acting as a wrapper $Q_{agg}(Q(D, V, T, S))$.

Utilizing the abilities to query utilizing wildcards, we can generate the following rich queries:

- Temporal Aggregations: Aggregates statistics over time (e.g., yearly averages).
- Spatial Aggregations: Aggregates statistics across spatial nodes (quadkeys).
- Spatiotemporal Aggregations: Combines both spatial and temporal dimensions in comprehensive summaries.

To perform aggregations between summary statistics stored in different leaf nodes, we leverage the Parallel algorithm as proposed by Chan et al. [28]. The algorithm is a generalized form of Welford's method and can update with pre-aggregated values instead of single values. This maintains correctness when combining mean and variance across distributions.

Aggregating summary statistics across nodes in a hierarchical tree structure allows for efficient data analysis across various dimensions, such as time, space, or both. By consolidating numerical and categorical statistics from relevant nodes, the aggregation query simplifies aggregate data analysis.

Correlation Queries: The Pearson correlation coefficient measures the linear correlation between two variables. Knowledge of correlation between variables can help inform identification of related variables. Correlation has many applications in data analysis and interpretation.

MAGELLAN provides a query $Q_{corr}(D, V_1, V_2, T, S)$ that can compute a correlation value efficiently between any two variables within a dataset. The calculation utilizes the tracked covariance of the two variables, the total count of records, and each variable's tracked variance. All of these values are tracked at the most granular level, which then implies that the final correlation value is as well. This allows a user to determine the correlation value in partitions of the data space and to compare those values to see where in time and space two variables may affect each other the most.

Density Queries: Kernel density estimation (KDE) provides a continuous estimate of the data distribution. Which makes it ideal for applications like visualizing the probability of certain events occurring. This can also be useful for modeling the behaviour of real data distributions.

In MAGELLAN, we provide a function to query KDE across various dimensions. Another key querying capability, provided by the underlying query system $Q(D, V, T, S)$, noted as $Q_{kde}(x, Q(D, V, T, S)$. Queries will be evaluated over the metadata tree structure.

The KDE is computed in an online fashion by incrementally updating bins as new data arrives. For each new data point, the relevant bin is indexed and the count is updated. Additionally, the density of value x is the sum of the contribution from each bin using a Gaussian PDF center at the bin's index. The result is then normalized by total count and bin width. One use case would include allowing users to understand the probability of a value occurring across space and time. In detail, a user could evaluate the likelihood of encountering the value 0.739, of *soil organic carbon percentage*, at different spatial, temporal, or spatiotemporal combinations.

IV. PERFORMANCE BENCHMARKS & DISCUSSION

We assess several aspects of our methodology. In particular, this includes: (1) profiling the MAGELLAN knowledge graph's space efficiency, construction overheads, and time for incremental updates. (2) computing the latencies and throughputs associated with the supported queries. (3) profiling the impact of our support for ambient findability.

A. *Datasets and Experimental Setup*

We have validated our methodology with several real-world, high-dimensional scientific datasets. Here we provide a brief overview of the datasets used in our benchmarks, along with the number of attributes and records.

- GRACEnet soil biology network: This dataset encompasses field experiments led by USDA-ARS scientists across 19 states in the USA. The focus is on soil carbon and greenhouse gas (GHG) emissions under various agricultural management systems. The dataset includes 1,459 unique attributes across 509,888 records. This dataset is also represented as 25 different tables.
- Global soil carbon fractions in the context of regenerative and conventional croplands: This dataset contains results from studies examining the response of soil organic carbon (SOC) pools to soil management practices. It includes 323 unique attributes across 4,236 records.
- Global soil carbon fractions in the context of managed and unmanaged Ecosystems: This dataset combines SOC

fractions includes data from agricultural systems, control plots, and NEON's 47 terrestrial research sites. The dataset comprises 23 unique attributes across 519 records.
- Global time-series soil carbon for DAYCENT and MEMS: this dataset supports the development and parameterization of soil organic carbon models. The dataset comprises 64 unique attributes across 9,776 records.
- Global N_2O database: Serving as a comprehensive repository for N_2O emission data. This dataset includes 351 unique attributes across 406,490 records.
- Global SOC in the context of grassland management: This dataset synthesizes research on the impact of grassland management conversion on soil carbon. The dataset comprises 23 unique attributes across 519 records.

The data described above all share the underlying domain of agriculture and climate. Each requiring a certain level of domain expertise to be able to navigate effectively. In total there are 2,243 unique variables, 931,428 records, 32 tables, all totaling 814 MB. In MAGELLAN, we utilize knowledge described by domain experts as the foundation and driving force for data retrievals.

B. *Profiling the* MAGELLAN *knowledge graph* [RQ-1, RQ-2]

Underlying auxiliary data structures facilitate search *for* and *in* data with MAGELLAN's knowledge graph. Space efficiency of the knowledge graph is critical to ensuring memory residency and efficiency of travels during query evaluations. The following are the memory footprints for the auxiliary structures comprising MAGELLAN: (1) AFO consumes 27.69 MB, (2) the internal mapping RDF graph consumes 1.4 MB, (3) 1.98 MB is consumed by the page rank map, (4) and 19.84 MB for the betweenness centrality map. Lastly, the memory footprint of the metadata tree structure is determined by the level of detail at which it is constructed. For our benchmarks, we constructed the tree with 8 quadkey characters, which makes the tree occupy 686.31 MB. The total memory footprint of the MAGELLAN framework is 737.22 MB.

C. *Tree Performance* [RQ-1, RQ-2]

The tree was constructed incrementally with real-world data consisting of 2,243 unique variables, 931,428 records, and 32 tables, totaling 814 MB. The construction of the tree with agricultural data, results in a large memory footprint in comparison to the raw data. This overhead is expected to diminish as the tree structure represents increasingly more data. Furthermore, once the structure of the tree stabilizes, the tree reaches a memory-bound ceiling, as shown in Fig. 7(a). In particular, beyond this point the tree primarily updates metadata using preexisting paths, diminishing the cost of construction over time, illustrated by Fig. 7(b) and 7(c).

D. *Preprocessing costs for queries* [RQ-2, RQ-3]

We enhance our knowledge graph with graph measures that facilitate estimation of the importance of nodes in the knowledge graph. We accomplish this by computing (1) betweenness centrality scores for the ontology, and (2) page rank scores for vertices. These operations performed can be thought of as one-time costs since ontologies are updated at a significantly slower rate than the data; this allows these costs to be amortized over multiple queries.

TABLE II: Basic queries are evaluated interactively and at high throughput. Especially, when all parameters are specified. Aggregation queries add a small additional computation overhead over basic queries.

Query	Latency (ms)		Throughput (q/s)	
	Mean	Std Dev	Mean	Std Dev
$Q(D, V, *, *)$	0.86	0.95	1,167	43
$Q(D, V, *, S)$	0.13	0.19	7,513	68
$Q(D, V, T, *)$	0.04	0.11	23,346	302
$Q(D, V, T, S)$	0.01	0.04	65,963	199
$Q_{agg}(D, V, *, *)$	1.15	1.51	869	18
$Q_{agg}(D, V, *, S)$	0.18	0.42	5,568	32
$Q_{agg}(D, V, T, *)$	0.08	0.17	13,233	193
$Q_{agg}(D, V, T, S)$	0.02	0.06	46,617	61
$Q_{kde}(x, Q(D, V, T, S))$	74.07	443.78	14	1

Computing betweenness centrality scores for the MAGELLAN graph takes 9.9 hours, and the page rank computation where the weight of each edge is 1 takes 2.48 seconds with 100 iterations. A cold-start construction of the metadata tree with real data takes 27 minutes.

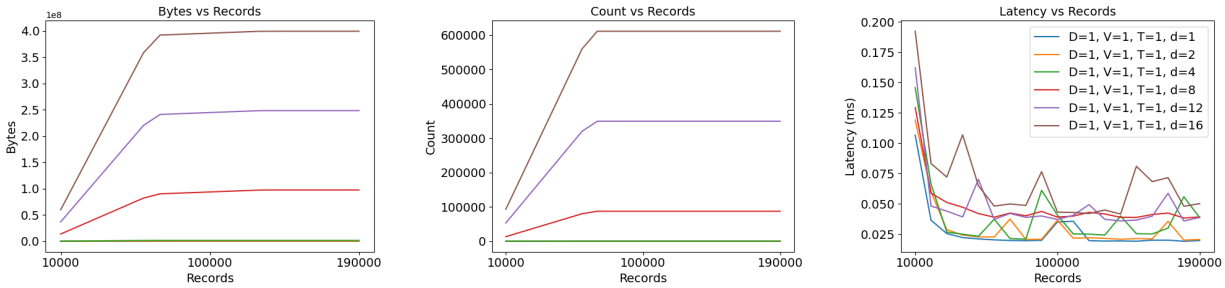
E. *Profiling Query Performance* [RQ-3]

MAGELLAN supports a large querying ecosystem consisting of simple single queries, aggregate queries, and specialized queries such as kernel density estimation (KDE) and correlations. Benchmarks are shown in Table II. All queries can be evaluated with different spatiotemporal combinations with support for wild cards. The user provides a value for the variable in which they want to evaluate the query upon.

After benchmarking, the query $Q_{kde}(x, Q(D, V, T, S))$ had a standard deviation of 443.78 ms. The standard deviation can be attributed to differences in record counts across datasets. For instance, GRACEnet soil biology network has 509,888 records, where as the global SOC grassland management dataset only has 519 records. Additionally, while MAGELLAN supports correlation queries within datasets, there are marginal performance differences compared to that of basic queries.

F. *Ambient Findability* [RQ-3]

MAGELLAN leverages a domain-specific ontology to build a rich contextual environment around the dataset variables. For instance, if a user queries the concept *potassium* MAGELLAN will surface variables; *K mgK kg*, *K Concentration g kg*, *K STD mgK kg*, and *Total K Amount kgK ha*. The variables return alongside information from "edges" encapsulating why those variables were chosen. Additionally, context will be included of the concept importance (page rank) and the centrality of the relationship (betweenness centrality). By associating each dataset variable with ontology concepts, the system allows users to start their search from a high-level concept rather than a specific variable, benchmarks show in Table III. This approach enhances findability by guiding the user from broader concepts to relevant features.



(a) Memory footprint as a function of the number of records for the tree, which grows in size as more records are added, but asymptotically approaches a limit of 399 MB at the highest level of detail (d=16).

(b) The number of nodes in the tree as a function of the number of records. The tree constructed with 190,000 records asymptotically approaches a limit near 611,672 nodes, at the highest level of detail (d=16).

(c) Query latency as a function of the number of records in the tree. Despite the growing complexity of the tree, latency declines due to the reuse of pre-existing paths for indexing.

Fig. 7: **Profiling tree construction.** D**: # of datasets,** V**: # of variables,** T**: # of time blocks, &** S^d**: detail level.**

TABLE III: Utilizing direct mappings generated using NLP, MAGELLAN is able to effectively return around 13 recommended variables, on average, when concepts are queried. Additionally, results are returned at interactive speeds.

Query	Latency (ms)		Throughput (q/s)		# Variables	
	Mean	Std Dev	Mean	Std Dev	Mean	Std Dev
Concept Query	10.27	3.90	97	3	13	25

TABLE IV: MAGELLAN semantically expands the query to recommend additional concepts. BFS-based semantic expansion yields a moderate number of related concepts with a higher latency. Narrower concept matches provide quicker responses with fewer results.

Query	Latency (ms)		Throughput (q/s)		# Concepts	
	Mean	Std Dev	Mean	Std Dev	Mean	Std Dev
BFS	125.02	224.05	8	0	8	18
Exact Concept Match	8.58	0.79	117	2	0	1
Related Concept	8.71	0.90	115	2	1	2
Broad Concept	8.90	1.49	112	3	6	19
Narrow Concept	8.57	1.40	117	1	6	18

Ambient findability is supported through semantic expansion techniques, where the system automatically expands a user's query by identifying and including semantically related concepts. Table IV shows benchmarked examples of our semantic expansion techniques.

V. CONCLUSIONS & FUTURE WORK

Our methodology allows fast, rapid explorations of the data space by enabling searches *for* the data and *in* the data.

RQ-1: Ontologies provide a key entry point to leverage domain knowledge. Because the ontology encapsulates information between concepts, it informs semantic expansion in our queries. Fusing the metadata tree with the ontology allows richer, deeper connections between variables in our datasets. Crucially, it also provides avenues for connections across disparate datasets maintained at a site.

RQ-2: Our knowledge graphs accelerate rapid explorations by combining different query types. This includes (1) SPARQL queries that explore semantic connections in the data; (2) graph-specific queries that we support by computing betweenness centrality and page rank estimates for nodes in our knowledge graph, (3) Queries targeting the metadata that allow explorations of statistical properties across portions of the data space, and (4) traditional queries that explore record-level connections amenable to conjunction, negation and intersections.

RQ-3: We provide a simplified, declarative interface that we then transform into series of predicates based on each query type's (imperative specification) requirements. The generation of query predicates is in the format expected by the engine, which frees the user from having to master the specificity of each query type. Our ambient findability dynamically relaxes bounds and finds proximate concepts and features of interest. Together, this allows richer explorations with a simplified interface allowing users to minimize blind spots in the data.

Our future work will focus on enhancing the query evaluation framework with a recommendation engine and a distributed architecture. The main goal is to guide searches based on how others have explored the search space, using collaborative filtering and a feedback system that allows users to rate the quality of recommendations.

VI. ACKNOWLEDGMENTS

This research was supported by the National Science Foundation (1931363, 2312319), the National Institute of Food Agriculture (COL014021223), and an NSF/NIFA Artificial Intelligence Institutes AI-CLIMATE Award [2023-03616].

REFERENCES

[1] I. Horrocks, M. Giese, E. Kharlamov, and A. Waaler, "Using semantic technology to tame the data variety challenge," *IEEE Internet Computing*, vol. 20, no. 6, pp. 62–66, 2016.

[2] D. Calvanese, I. Horrocks, E. Jiménez-Ruiz, E. Kharlamov, M. Meier, M. Rodriguez-Muro, and D. Zheleznyakov, "On rewriting and answering queries in obda systems for big data," vol. 1080 of *CEUR Workshop Proceedings*, (Aachen), RWTH, 2013.

[3] C. Keet, R. Alberts, A. Gerber, and G. Chimamiwa, "Enhancing web portals with ontology-based data access: The case study of south africa's accessibility portal for people with disabilities.," vol. 432, 01 2008.

[4] I. Athanasiadis, A.-E. Rizzoli, S. Janssen, E. Andersen, and F. Villa, "Ontology for seamless integration of agricultural data and models," vol. 46, pp. 282–293, 10 2009.

[5] D. Calvanese, G. De Giacomo, D. Lembo, M. Lenzerini, A. Poggi, M. Rodriguez-Muro, R. Rosati, M. Ruzzi, and D. F. Savo, "The mastro system for ontology-based data access," *Semantic Web*, vol. 2, pp. 43–53, 01 2011.

[6] P. Haase, I. Horrocks, D. Hovland, T. Hubauer, E. Jiménez-Ruiz, E. Kharlamov, J. W. Klüwer, C. Pinkel, R. Rosati, V. Santarelli, A. Soylu, and D. Zheleznyakov, "Optique system: towards ontology and mapping management in obda solutions," in *WoDOOM*, 2013.

[7] E. Galeota and M. Pelizzola, "Ontology-based annotations and semantic relations in large-scale (epi)genomics data," *Briefings in Bioinformatics*, vol. 18, pp. 403–412, May 2017.

[8] C. Jonquet, N. H. Shah, C. H. Youn, M. A. Musen, C. Callendar, and M.-A. Storey, "NCBO Annotator: Semantic Annotation of Biomedical Data," in *ISWC 2009 - 8th International Semantic Web Conference, Poster and Demo Session*, Oct. 2009. Issue: 171.

[9] D. Chaves-Fraga, O. Corcho, F. Yedro, R. Moreno, J. Olías, and A. D. L. Azuela, "Systematic construction of knowledge graphs for research-performing organizations," *Information*, vol. 13, no. 12, p. 562, 2022.

[10] Y. Lin, H. Xu, and Y. Bai, "Semantically enhanced catalogue search model for remotely sensed imagery," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 10, no. 4, pp. 1256–1264, 2017.

[11] B. Cui, B. C. Coi, J. Su, and K.-L. Tan, "Indexing high-dimensional data for efficient in-memory similarity search," *IEEE Transactions on Knowledge and Data Engineering*, vol. 17, no. 3, pp. 339–353, 2005.

[12] P. Bereuter and R. Weibel, "Real-time generalization of point data in mobile and web mapping using quadtrees," *Cartography and Geographic Information Science*, vol. 40, no. 4, pp. 271–281, 2013.

[13] J. M. Perea-Ortega and L. A. Ureña-López, "Geographic expansion of queries to improve the geographic information retrieval task," in *Natural Language Processing and Information Systems*, pp. 94–103, Springer, 2012.

[14] G. Fu, C. B. Jones, and A. I. Abdelmoty, "Ontology-based spatial query expansion in information retrieval," in *On the Move to Meaningful Internet Systems 2005: CoopIS, DOA, and ODBASE*, pp. 1466–1482, Springer, 2005.

[15] D. Ramagem, B. Margerin, and J. Kendall, "Annoterra: building an integrated earth science resource using semantic web technologies," *IEEE Intelligent Systems*, vol. 19, no. 3, pp. 48–57, 2004.

[16] F. Mata and C. Claramunt, "Geost: Geographic, thematic and temporal information retrieval from heterogeneous web data sources," in *Web and Wireless Geographical Information Systems*, pp. 5–20, Springer, 2011.

[17] M. Lutz and E. Klien, "Ontology-based retrieval of geographic information," *International Journal of Geographical Information Science*, vol. 20, no. 3, pp. 233–260, 2006.

[18] M. Malensek, S. Pallickara, and S. Pallickara, "Fast, ad hoc query evaluations over multidimensional geospatial datasets," *IEEE Transactions on Cloud Computing*, vol. 5, no. 1, pp. 28–42, 2015.

[19] S. Mitra, P. Khandelwal, S. Pallickara, and S. L. Pallickara, "Stash: Fast hierarchical aggregation queries for effective visual spatiotemporal explorations," in *2019 IEEE International Conference on Cluster Computing (CLUSTER)*, pp. 1–11, IEEE, 2019.

[20] G. Fox, S. Pallickara, M. Pierce, and H. Gadgil, "Building messaging substrates for web and grid applications," *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 363, no. 1833, pp. 1757–1773, 2005.

[21] S. Pallickara and G. C. Fox, "On the matching of events in distributed brokering systems.," in *ITCC (2)*, pp. 68–76, 2004.

[22] C. Jonquet, A. Toulet, E. Arnaud, S. Aubin, E. Dzalé Yeumo, V. Emonet, J. Graybeal, M.-A. Laporte, M. A. Musen, V. Pesce, and P. Larmande, "AgroPortal: A vocabulary and ontology repository for agronomy," *Computers and Electronics in Agriculture*, vol. 144, pp. 126–143, Jan. 2018.

[23] M. A. Musen, "The protégé project: a look back and a look forward," *AI Matters*, vol. 1, pp. 4–12, June 2015.

[24] M. Dai, N. H. Shah, W. Xuan, M. A. Musen, S. J. Watson, B. D. Athey, F. Meng, *et al.*, "An efficient solution for mapping free text to ontology terms," *AMIA summit on translational bioinformatics*, vol. 21, 2008.

[25] S. Loh, L. K. Wives, D. Lichtnow, and J. P. M. De Oliveira, "Concept-Based Text Mining," in *Handbook of Research on Text and Web Mining Technologies*, pp. 346–358, IGI Global, 2009.

[26] B. P. Welford, "Note on a Method for Calculating Corrected Sums of Squares and Products," *Technometrics*, vol. 4, pp. 419–420, Aug. 1962.

[27] E. Schubert and M. Gertz, "Numerically stable parallel computation of (co-)variance," in *Proceedings of the 30th International Conference on Scientific and Statistical Database Management*, SSDBM '18, (New York, NY, USA), pp. 1–12, ACM, July 2018.

[28] T. F. Chan, G. H. Golub, and R. J. LeVeque, "Updating Formulae and a Pairwise Algorithm for Computing Sample Variances," in *COMPSTAT 1982 5th Symposium held at Toulouse 1982*, pp. 30–41, Physica-Verlag HD, 1982.