Distributed processing of spatiotemporal ocean data: a survey

Xiaoyong Li[1] · Jingyun Gu[2] · Guolong Tan[2] · Wenjing Jiang[2] · Ao Cui[2] · Leiming Shu[2] · Kaijun Ren[1,3] · Haoyang Zhu[4] · Jedi S. Shang[5] · Zichen Xu[2]

Abstract

Ocean data exhibits interesting yet human critical features affecting all creatures around the world. Studies on Hydrology and Oceanology become the root of many disciplines, including global resource management, macro economy, environment protection, climate predictions, etc, which motivates our further exploration on the underlying feature behind the ocean data. However, with high dimensionality, large quantities, heterogeneous sources, and especially, the spatiotemporal manner, the diversity between the specific knowledge required and massive data chunk puts forward unique challenges in data representation and knowledge mining, effectively. This paper tends to provide a summary of studies on these issues, including the data representation, data processing, knowledge discovery, and algorithms on finding unique patterns on ocean environment changes, such as temperature, tide height, waves, salinity, etc. In detail, we comprehensively discuss about ocean spatiotemporal data processing techniques. We further summarize related representation works on ocean spatiotemporal data, the construction of a ocean knowledge graph, and the management of ocean spatiotemporal data. At last, we combine and compare the collection of the evolution and multiple state-of-the-arts on ocean spatiotemporal data processing.

Keywords Ocean data · Spatiotemporal · Knowledge graph · Representation · Distributed processing

1 Introduction

Ocean environment is highly related to the ecological system and human lives, by furnishing humankind with diverse resources and services. Ocean serves as oxygen supply, climate regulation, carbon sequestration, food and medicine supply, etc., which is of great significance to the survival and development of human society.

✉ Haoyang Zhu
zhuhaoyang@nudt.edu.cn

✉ Zichen Xu
xuz@ncu.edu.cn

Extended author information available on the last page of the article

With the rapid development of information technologies, data acquired from ocean observation platforms grows exponentially every day. Ocean data is mainly obtained through various observation devices from land, sea surface, underwater, aerospace, etc. It is an accumulation of a large amount of data from different time periods, scales and regions. Compared with normal data, ocean spatiotemporal data emphasizes more on the dynamic process. The spatiotemporal process of ocean data is mainly reflected in ocean phenomena. The spatiotemporal process of ocean phenomena not only exists in a certain spatial scope, but also holds a certain continuity in time. The characteristics of different temporal states are different, and the characteristics in different moments are different. Some characteristics change constantly.

The spatiotemporal process of ocean environmental data acts as a primary role in ocean environment research. There are a variety types of ocean data, and the format and record type of different sources and types of data are different. In practice, it is often necessary to use multiple formats of data, and the different formats of data bring great inconvenience. After obtaining ocean data, researchers tend to process the data differently and extract valuable information from it. In order to obtain valuable information that meets the needs, it is necessary to reorganize and represent data in a readable and operable way, so that the data can be further exploited and utilized.

Data representation is the transfer of our experience of the actual world into the computational domain, and it is the way how data is stored, processed, and transmitted [1]. However, the inherent complexity of ocean data brings a great challenge to representation. The traditional representation methods are not able to present the spatial and temporal features of ocean in an effective way. Therefore, multiple updated and enhanced representation are proposed to describe the dynamic and flexible process of ocean data. Among these methods, semantic web [2] gradually become the main trend for representing spatiotemporal data currently.

Semantic web can be originated back to 1956, when Richens first carried out semantic net or semantic network [3]. It is first used for knowledge-based system reasoning and problem solving. After that, MYCIN [4] was designed as an early expert diagnosis system base on rules. Then, RDF [5] and OWL [6] was introduced, as the core schema of semantic web. A series of open domain semantic web or ontology were brought in, including Cyc [7], Freebase [8], DBpedia [9], YAGO [10], PROSPERA [11].

In 2012, knowledge graph [12] was first introduced by Google. Since then, knowledge graph gained great popularity and further exploration. A knowledge fusion framework KnowledgeVault [13] is developed on the basis of knowledge graph for large-scale knowledge. A knowledge graph is essentially a large semantic web for describing concepts, entities and their relationships in the objective world [14]. Knowledge graph provides a more human-like method to represent information and knowledge in the computer world. Knowledge graph is labelled as large-scale, semantic-rich, high quality, structure friendly, etc.

The graph-based structure of knowledge graph can effectively represent and store the spatiotemporal characteristics of ocean data by entities and relationships. Geographic knowledge graph usually reflects spatiotemporal features in data. Typical geographic knowledge graph includes LinkedGeoData [15], LinedSpatiotemporalData [16], etc. In the construction of a ocean knowledge graph, to embed the spatiotemporal element into the structure, advanced theories techniques or methods are applied, including spatiotemporal entity recognition, spatiotemporal disambiguation, semantic extension and so on. In this survey, Section 2 introduces the representation methods on ocean spatiotemporal data based on the characteristics, Section 3 elaborates design concept of a knowledge graph, Section 4 introduces the construction steps of a ocean spatiotemporal knowledge graph. In Section 5, ocean data processing methods of single-node and multi-node is

comprehensively explained. In Section 6, the performance evaluation of ocean spatiotemporal knowledge graph is presented, and Section 7 makes a summary to this survey.

2 Representation on ocean data

Data representation is a reflection of real-world data in a computer-readable and operable version, providing an approach to analyze raw data. Simple data representation includes binary digits, numeric data, character data, etc. As data formats, data structure and data volume growing more complex, the representation methods are also required to be "upgraded" to different forms, including tables, graphs, vectors, functions, distribution, data models, etc. In terms of ocean data representation, the heterogeneity and spatiotemporal characteristics should be fully depicted by the representation.

2.1 Ocean data characteristics

Ocean data is vast and diverse, including meteorology data, hydrology data, hydroacoustics data, seafloor topography and geomorphology, ocean chemistry data, etc. Apart from the characteristics of big data, that is, high volume, high variety, high velocity, high value, high veracity and high validity, ocean data is also characterized by multiple ocean properties. Major ocean data characteristics that impact ocean data representation includes:

- *High volume.* Various ocean observation programs cover almost all the oceans worldwide and carry out huge amount of periodic and real-time data collection. The volume of ocean data is increasingly growing, and the overall volume has reached EB level.
- *Heterogeneous.* The sources of ocean data acquisition are from a wide range, including ocean surveys, observation platforms, remote sensing and so-on. The formats and quality of these data also vary from their observation methods, extraction models, structure, application and analysis. These characteristics have made ocean data heterogeneous and high-dimensional.
- *Dynamic.* Ocean is an obvious dynamic system with rapid changing data flow. With the advancement of observation methods and devices, and the improvement of data processing, ocean data are collected by seconds, which results in information in ocean database changing constantly and the data updates getting more frequently.
- *Spatiotemporal.* Ocean data carries both spatial and temporal attributes inherently. In spatial scale, ocean data involves nearshore, offshore, polar regions, sea surface, deep and distant ocean data, etc. In temporal scale, ocean data includes variability ranges from seconds, minutes, hours, days to seasons, years even multiple centuries. Ocean data embodies different characteristics at different spatiotemporal levels.

Therefore, ocean sptaiotemporal data representation are required to include all these characteristics to describe ocean data more correctly and make a better prerequisite for further ocean data analysis.

2.2 Data representation methods

Researchers have performed different representation methods to depict spatiotemporal ocean data and continuously made improvements on these methods. For example, map is

a typical representation for spatially distributed data, but map generally only reflects the surface information whereas no information on deeper layers of earth. To solve this limitation, Chung et al. [1] analyzed three classical presentations-probability measures, Dempster-Shafer evidential belief functions and fuzzy logic functions, by applying favorability functions to represent information of *m* layers of the earth, in 1993. In [17], a spatial data representation with dynamic graphics was proposed, with a classification method where maps can incorporate dynamism [18]. In 1996, Tuohy et al. [19] proposed a geophysical data representation method with interval B-spline function, which facilitates data archiving and reduces data storage. Spline is a piecewise polynomial curve, which functions well in multi-dimension data interpolation. In 2010, Bibby et al. [20] proposed a hybrid representation method for ocean environment, where stationary objects are represented by point features and trajectories of dynamic objects are represented by cubic splines.

However, these methods either cannot present the spatiotemporal process completely or cannot present the dynamic process in a perfect shape. Therefore, graphed-based semantic web has been wildly applied to representation on ocean spatiotemporal data, for graph-based semantic is a more proper method in representing dynamic and heterogeneous data.

The use of semantic web can be originated back to 2000s [21–23]. There have been several different methods to involve spatiotemporal and other features into ocean data representation. Raskin [21] developed a semantic web (SWEET) for geo-terminology by building a collection of spatiotemporal ontologies. MacGregor [22] designed a semantic primitive (SEW) especially for contextualized data, to conduct abstractions on related resources. A mapping between semantic web and geospatial data processing standards were established for Spatial Data Infrastructure (SDI) [16].

In terms of information deluge in recent years, [24] presented an agile data architecture (CRISIS) for real-time data representation of multi-source heterogeneous ocean data streams with semantic web technologies in 2018. Later, [25] presented an reorganized and enhanced version of [24], including an isolation of functionalities to utilize multi-source querying and the discovery of alarms. Wang et al. [26] designed a formalized geographic knowledge representation (GeoKG) that describes the evolution of spatiotemporal data. Ren et al. [27] propounded an unified semantic model (OEDO) to represent heterogeneous ocean data by metadata.

3 Design of ocean spatiotemporal knowledge graph

Knowledge graphs are structured semantic knowledge bases for effectively and comprehensively describing concepts and the complex relationships between them in the physical world in a structured way, by aggregating a large amount of knowledge and creating connections between information, thus realizing quick response and knowledge reasoning. In terms of domain, knowledge graphs are usually divided into general knowledge graphs and domain knowledge graphs. General knowledge graph can be regarded as a structured encyclopedic knowledge base that contains a large amount of common sense knowledge in the real world with high convergence. Domain knowledge graph, also referred as industry knowledge graph or vertical knowledge graph, is usually oriented to a specific domain base on industry data, which has been widely used in the industrial field. We focus only on domain knowledge graph in connection with ocean spatiotemporal data in the survey. The logical structure of a knowledge graph consists of data layer and schema layer.

3.1 Data layer

Data layer stores real-world data. Data forms includes structured data , demi-structured (XML, json), and unstructured data (images, recordings or videos).In data layer, data or facts are stored in RDF (Resource Description Framework). RDF provides a unified standard for describing entities and resources, which is also a method of data representation. RDF is formally represented as an SPO(Subject, Predict, Object) triple, which stands for a piece of knowledge in a knowledge graph. RDF consists of nodes and edges. Nodes represent entities/resources or attributes, and edges stand for the relations between entity and entity or entity and attribute. Triples can be presented as "entity-relation-entity" or "entity-attribute-attribute value".

Entities are the basic elements of the knowledge graph, which refer to specific names of people, organizations, places, dates, times, etc. Relation is a semantic relationship between two entities, which is an instance of the relationship defined by the schema layer. An attribute is a description of an entity and is a mapping relation between an entity and an attribute value. However, RDF is limited in representation on how to distinguish classes and objects and on how to define and describe the relations of classes or attributes. Based on RDF, researchers have developed RDFS (Resource Description Schema) [28] and OWL (Web Ontology Language) [6]. RDFS is a set consisted of predefined vocabulary that can describe RDF, while OWL is more like an extension version of RDFS that provides fast and agile data modeling with effective reasoning.

3.2 Schema layer

Schema layer is on top of data layer, which is the core structure of knowledge graph. Schema layer is managed by ontology. Schema layer acts as the conceptual model and logical foundation of the knowledge graph, and provides the specification constraint for the data layer. Mostly, ontology is adopted as the schema layer of knowledge graph, and the data layer of knowledge graph is constrained with the rules and axioms defined by ontology. The knowledge graph can also be regarded as an instantiated ontology, and the data layer of the knowledge graph is an instance of the ontology. In the schema layer of the knowledge graph, nodes represent ontology concepts and edges represent relations between concepts.

3.2.1 Ontology

Ontology is originated from a branch of philosophy. In computer science and information technology, ontology refers to a specification vocabulary for a shared domain of discourse — definitions of classes, relations, functions, and other objects [29]. An ontology provides a shared vocabulary, which can be used to model the the type of objects or concepts and their properties and relations that exist within a given domain [30]. The purpose of ontology is to capture the knowledge in related domains, identify the commonly accepted vocabulary, describe the semantics of concepts through the relations between concepts, and provide a consensus understanding of the knowledge.

Knowledge in ontologies is represented formally through classes, relations, functions, axioms, and instances. Perez et al. [31] organized ontologies using a taxonomy that summarizes five basic modeling meta-speak.

1. Class or concept: Class or concept refers to any transaction, such as job descriptions, functions, behaviors, strategies, and reasoning processes. Semantically, it represents a collection of objects whose definition includes the name of the concept, a collection of relations with other concepts, and a description of the concept in natural language.
2. Relation: Relation is the interaction between concepts in the domain, formally defined as a subset of the n-dimensional Cartesian product. For example, subClassOf relations.
3. Function: Function is a special type of relations. The first $(n-1)$ elements of the relation can uniquely determine the n_{th} element, formally defined as $F : C_1 \times C_2 \times ... \times C_{n-1} \times C_n$. For example, memberOf is a function, memberOf(x,y) means y is the member of x.
4. Axiom: Axiom represents the eternal truth assertion, such as concept B belongs to the scope of concept A.
5. Instance: Instance represents elements, or in semantic, instance represents object.

There are many existing ontologies, and the process of constructing ontologies varies according to the consideration of their target domains and specific projects. Since there not exists an official standard for ontology construction, researchers have proposed a series of principles for constructing ontologies in practice. Some of the ontology construction principles that have proved to be pragmatic. The five principles proposed by Gruber in 1995 [32] are the most influential. These construction principles provide the basic idea and framework for constructing ontologies. However, the obvious shortcoming is that they only deliver a rather vague standard. It is now generally accepted that the process of constructing a domain-specific ontology requires the involvement of domain experts. Principles for ontology construction include:

1. Clarity and Objectivity : Ontologies should offer clear and objective semantic definitions on defined terms by objective definitions and natural language documents.
2. Completeness : Definition of the term should be complete and fully expresses the meaning of the described term.
3. Coherence: The inferences drawn from the terms are compatible with the meaning of the terms themselves, i.e., they support reasoning consistent with their definitions without contradiction. The axioms defined and the documents illustrated in natural language should also be consistent.
4. Maximum Monotonic Extendibility: Adding general or specialized terms to an ontology does not require modifying its existing conceptual definitions and content, and supports defining new terms based on existing concepts.
5. Minimal Ontological Commitments: Ontological commitments should be minimal and should hold as few constraints as possible to the modeled objects. The commitment in ontology refers to the consensus on how to use the shared vocabulary in a consistent and compatible way. In general, ontology commitments are sufficient to satisfy specific knowledge sharing needs, which can be ensured by defining the least constrained axioms and defining the vocabulary needed for communication only.

3.2.2 Spatiotemporal ontology

Spatiotemporal ontology are required to present spatial attributes and temporal attributes of related entities. Spatiotemporal ontology is more than just an "enhanced" ontology, it also needs to combine business scenarios and domain knowledge, as well as semantically and spatially extended knowledge concepts, entities, and relationships based on the characteristics of

spatiotemporal knowledge. In addition to defining semantic linkages, spatiotemporal knowledge mapping must also address the description of spatial and temporal interactions, and the important challenge in the design of spatiotemporal knowledge mapping is how to map spatiotemporal and semantic relations. Galton [33] summarized that a fully spatiotemporal ontology must extend the field-based and object-based ontologies in spatiotemporal domains, especially with the natural phenomena that inhabits the data. However, spatiotemporal information processing today faces two major problems: challenges in information integration led by incompatible terminology, and a deficiency in interoperability among the different systems [34].

At present, there are two ways to design ontologies related to spatiotemporal data. The first one is to add or optimize spatiotemporal related entities and relations to them to extend the original semantics, based on the existing ontologies . Bittner et al. [35] proposed an ontology theory that can describe dynamic spatiotemporal processes and constant enduring entities. To enhance the exchange and integration of semantic heterogeneous of spatiotemporal data. Bittener [34] specified the meanings of terms that describes the basic types of entities and relations almost used in every domain and developed a formal logical-based ontology using an axiomatic theory.

Some research are developed on the basis of existing open-domain ontology. YAGO2 [36] is a spatially and temporally enhanced version built from Wikipedia, WordNet and GeoNames, by adding YAGO (YAGO unifies Wikipedia and WordNet [37] with high coverage that significantly improving the efficiency of information extraction, which combines extensive lexicons in Wikipedia and taxonomy from WordNet.) a temporal dimension and a spatial dimension for both entities and facts. In [38], researchers provided a timely YAGO that also extracts temporal facts from Wikipedia. In terms of information integration, an ontology with spatiotemporal entities integrated is developed to fit dynamic phenomena [39]. Kurte et al. [40] offered an ontological framework that integrates spatiotemporal dimensions for describing dynamic patterns. Hornsby et al. [41] proposed a method of tracking spatiotemporal changing based on the object identity. The semantic of this research applied systematic derivation to semantics associated with changes, and were able to extract more types of dynamic spatiotemporal changes compared to its former research. In [42], a structured, spatiotemporal data querying over some Open Data sets were proposed, by adding geo-entities, temporal entities and links between them.

Another method is to reconstruct a unified spatiotemporal ontology. Grenon [43] proposed a realist formal spatiotemporal ontology, where he presented ontology as a theory that the framework can be applied to various spatiotemporal domains. Carstensen [44] presented a new proposal for the design of spatiotemporal ontologies which has its origin in cognitively motivated spatial semantics. He also leveraged selective attention to ontologies that leads to defining an ontological upper structure covers spatiotemporal domains. In [45], researchers developed ontologies that solve semantic ambiguation of spatiotemporal entities in particular. Grenon et al. [46] presented another modular ontology to describe the changing and dynamic features as well as snapshots of time. Arpinar et al. [47] provided a geospatial ontology (SWETO) that integrates analytics, including spatial, temporal and thematic dimensions of information.

4 Construction of ocean spatiotemporal knowledge graph

There are two methods of constructing a knowledge graph: top-down and bottom-up. Top-down is to define the ontology and data schema for the knowledge graph first, and then extract the ontology and schema information from high-quality data to add to the

knowledge base with the help of structured data sources such as encyclopedic websites. Bottom-up is to propose resource schemas from public data through certain technical means, select the schema with higher confidence, add them to the knowledge base after manual review, and then construct the top-level ontology schema afterwards.Bottom-up organizes entities inductively to form bottom-level concepts, and then gradually abstract upward to form top-level concepts. This method can be converted into a data schema based on existing standards, or generated based on mapping of high-quality domain data sources. At present, knowledge graph construction generally adopts bottom-up method, thus we will only talk about bottom-up in this survey.

The basic processes of spatiotemporal knowledge graph is shown in Figure 1. There are six steps of the construction: knowledge modeling, knowledge storage, knowledge extraction, knowledge fusion, knowledge computation and application. It starts with raw data processing, where the data may be structured, unstructured and semi-structured. Then knowledge elements, that is, entities and relations, are extracted by a series of automated or semi-automated techniques and are stored in the schema layer and data layer of the knowledge base.

4.1 Knowledge modeling and knowledge storage

Knowledge modeling is abstraction based on knowledge characteristics and actual demand of the industry, under the mode of knowledge graph. Knowledge modeling is more like the same process as representation, which has been discussed in Section 2.

Knowledge storage will directly influence efficiency of data querying and application. At present, there are generally two methods for knowledge storage. The first one is to store through a standardized storage format such as RDF, which has been discussed in Section 3. Another approach is to use graph databases for storage, and we will discuss this in Section 5 in detail.

4.2 Knowledge extraction

The entities, attributes and relations among entities are extracted from various types of data sources, based on which the ontological knowledge representation is formed. Knowledge
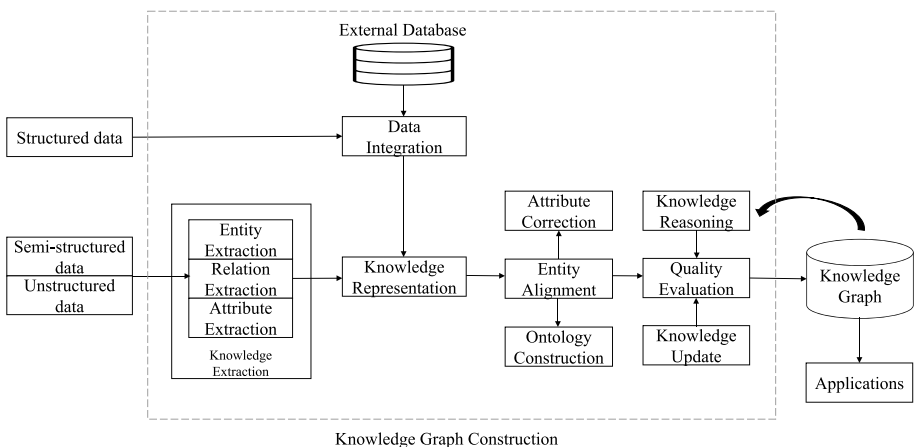

Figure 1 Knowledge graph construction steps

extraction is a technique to automatically extract structured information such as entities, relationships and entity attributes from semi-structured and unstructured data. For different types of data sources, the techniques used for knowledge extraction are different. For structured data (e.g. maps, gazetteers), spatial entities, attributes and their relations are automatically extracted from the database by establishing mapping relations between concepts in the database and ontologies in knowledge graph and rule-based reasoning. For semi-structured data (e.g. tables from webpages and list data), corresponding template extractors can be established to realize knowledge extraction. For unstructured data (e.g. webpage text or other text information), the existing knowledge graph can be used to build a training set by remote supervision, and the extract by using deep learning algorithms. Knowledge extraction includes entity extraction, relation extraction and attribute extraction.

4.2.1 Entity extraction

Entity extraction, also called named entity recognition (NER), can identify named entities from text database automatically. Main tasks of entity extraction is to identity the named entities and classify them.

DeepDive [48] is a knowledge extraction tool developed by Stanford University. It extracts structured knowledge from less structured data and reason statistically without machine learning algorithms. In [49], researchers developed knowledge extraction that links ontological classes to the influenza-related spatiotemporal text data on Twitter. In [50], an knowledge extraction approach was presented that combines temporal information retrieval and spatial information retrieval in text documents.

4.2.2 Relation extraction

The text corpus obtained after entity extraction is a series of discrete named entities (nodes). To collect semantic information, it is necessary to extract the association relations (edges) between entities from the related text to link multiple entities or concepts to form a web-based knowledge structure. According to the dependence on annotated data, entity relationship extraction methods can be classified into supervised learning methods, semi-supervised learning methods, unsupervised learning methods and open extraction methods.

Supervised learning is a fundamental entity relation extraction method. The main idea is to train machine learning models on labeled training data and then to classify the relation of the test data. Supervised learning methods include rule based methods, feature based methods and kernel based methods. The rule based method needs to summarize the corresponding rules manually or through machine learning methods according to the different domains involved in the text to be processed, and then use the template matching method for entity relationship extraction. Spatiotemporal rule based relation extraction have to extract spatiotemporal relations in text corpus based on syntactical rules [51, 52]. Feature based method is simple and effective. The main idea is to extract useful information (including lexical and syntactic information) from the context of relations instances as features, construct feature vectors, and train entity relationship extraction models by computing the similarity of the feature vectors. Kernel based relation extraction includes word sequence kernel function methods, dependency tree kernel function methods, shortest path dependency tree kernel function methods, convolutional tree kernel function methods and the combined kernel function methods. Kernel based methods are more widely used for

spatiotemporal relation extraction, for its effectiveness in analyzing heterogeneous data and dealing with massive number of documents [53].

Semi-supervised relation extraction summarizes entity relationship sequence patterns from the context containing the relations, and then uses the relationship sequence patterns to discover more relationship seed instances to form a new set of relations. Semi-supervised method assist researchers in labeling professional spatiotemporal data without expert knowledge [54].

Unsupervised relation extraction method does not need to rely on entity relation annotation corpus. It consists of two steps: relationship instance clustering and relation type selection. Lu et al. [55] proposed an unsupervised learning methods based on variational autoencoder to extract information from spatiotemporal data.

Open relation extraction can avoid manual corpus construction for specific relationship types, and can discover the relation type and extract relation automatically . By mapping high-quality entity relation instances to large-scale text, training data can be obtained from external domain-independent entity knowledge bases (such as DBPedia, YAGO, OpenCyc, FreeBase) according to text alignment. Open relation extraction is effective for intrinsic difficulty in training individual extractors for every single relation [56].

4.2.3 Attribute extraction

Attribute extraction is to extract the attribute information of a specific entity from different information sources. Data mining method can be used to mine the relations between entity attributes and attribute values directly from the text.

4.3 Knowledge fusion

After knowledge extraction, spatiotemporal knowledge from different data sources have certain complementarities and differences, such as non-uniform classification systems, ambiguities in geospatial entities, different details of feature descriptions, conflicted entity relations, and other information redundancy and inconsistency issues. Knowledge fusion is an effective way to solve the problem of knowledge graph heterogeneity by associating the semantic understanding of different identified entities in different data to the same entity.Techniques of knowledge fusion includes entity disambiguation, and entity linking.

Knowledge fusion is an effective method to improve the quality of knowledge, disambiguate knowledge and get the true value of knowledge, especially for heterogeneous data [57]. Spatiotemporal knowledge fusion includes more step on time series cleaning, spatiotemporal cleaning of stale data [58] and stream data cleaning [57].

4.4 Knowledge computation

After information extraction and knowledge fusion, a series of basic fact representation has been acquired from raw chaotic data. The next step is to obtain a structured, networked knowledge system and update mechanism through knowledge computation. Main steps of knowledge computation involves ontology construction and knowledge reasoning. We have discussed ontology in Section 3. Knowledge reasoning is mainly used for completing the knowledge graph and verifying the quality.

In addition to the ontology, reasoning based on general rules and common sense is widly used in knowledge graphs. Spatiotemporal knowledge graphs are capable of temporal reasoning and spatial reasoning. Temporal reasoning can supplement the target query with temporal constraints so the result meets the temporal demand. It can be regarded as a constraint satisfaction problem, where the variables represent temporal objects and the constraints between variables correspond to the temporal relations between objects. Similar to temporal reasoning, the spatial reasoning process yields the understanding of multiple spatial objects and object-embedded spatial properties. Spatial reasoning contains the reasoning of multiple spatial relations, such as topology, direction, distance, etc. Other logic-based geo-knowledge language was added in aid of declaring spatiotemporal reasoning [59]. Mantle et al. [60] implemented ParQR, a parallel, distributed Qualitative Spatial Temporal Reasoning (QSTR) with Apache Spark to reasoning through massive spatiotemporal data. A incorporation spatiotemporal reasoning is presented in [61], which infers spatiotemporal representations over underlying ontology.

4.5 Ocean knowledge graph application

The construction of ocean spatiotemporal knowledge graph present the ocean related information in a structured way, which helps us learn more of variation and prediction of ocean environment. Based on the structured ocean knowledge, more supportive and executable decisions can made.

5 Management of spatiotemporal ocean data

Storing and analyzing ocean and marine environmental data is an important way of understanding our planet and preparing us in advance for potentially adverse ocean conditions in the future. In addition, the marine spatiotemporal data collected from various sources (such as meteorological satellites, road-based weather stations, meteorological hot air balloons, buoys, various ships, underwater sensors, etc.) has reached the petabyte level, and traditional centralized data processing has gradually been unable to adapt the need for ocean spatiotemporal data management. How to store and utilize these ocean spatiotemporal big data is an urgent problem to be solved at present. Ocean spatiotemporal data management can be divided into two categories in terms of the number of nodes, namely the single-node storage and processing model and the distributed multi-node storage and processing model. The two types of models are introduced below.

5.1 Single-node

The traditional relational database management system RDBMS is a typical single-node processing model. Therefore, many researchers have developed some spatiotemporal RDBMS that support spatiotemporal data storage based on traditional RDBMS, and have been widely used in the industry, such as PostGIS of PostgreSQL [62], Oracle [63], IBM DB2 Spatial Extemder [64], SQLite [65], MySQL Spatial [66], SpatiaLite of Microsoft SQL Server [67], etc. Besides, new hardware GPUs also assistant in accelerating graph computation [68, 69]. These spatiotemporal RDBMSs are stable, mature, and efficient, including efficient SQL query engines. Among them, only PostgreSQL's PostGIS and Oracle Spatial support the storage and processing of spatial raster data. PostgreSQL's PostGIS,

Oracle Spatial, and SQL Server, among others, provide OGC [70] and support the full set of spatial relational and analytical functions defined in the ISOSQL/MM (part-3) [71] standard. Therefore, queries such as spatial joins, query spatial extents, etc., can be performed in these databases.

However, these spatiotemporal database systems developed and expanded based on traditional RDBMS lack distributed data storage and processing capabilities like traditional RDBMS. These single-node data services are limited by I/O bottlenecks, lack parallel computing capabilities, and are difficult to scale horizontally. As the amount of marine spatiotemporal data increases, their corresponding latency and performance continue to decline, making it difficult to process PB-level marine spatiotemporal data. Moreover, the marine spatiotemporal data has the characteristics of complex sources, diverse structures and different qualities, making it difficult to model them in spatiotemporal RDBMS. Although traditional RDBMSs can scale horizontally through data sharding, it is still difficult to store data in tabular format to support distributed storage and processing of ocean data.

5.2 Multi-node

Multi-node data processing refers to the use of distributed computing technology to process data, and distributed computing is a concept relative to centralized computing. A distributed network consists of several computers that can communicate with each other, each with its own processor and storage device. The huge computing tasks that were originally concentrated on a single node are distributed to the computers in the distributed network for parallel processing in a load-balanced manner.As shown in Figure 2, each cluster of a distributed storage system generally has a master control node, and the load balance of data on each node is realized through the master control node scheduling. Worker nodes send information about node load to the master node through heartbeat. The master node calculates the workload of the worker nodes and the data to be migrated, generates migration tasks and puts them in the migration queue for execution.

In order to ensure the high reliability and high availability of the distributed storage system, multiple copies of the data of each node need to be replicated and backed up, as shown in Figure 3. Generally, there is only one primary replica, which can provide read/write services, and there can be multiple backups replica, which provide read-only services. In a distributed data processing system, data can be synchronized to multiple storage nodes through a replication protocol, and data consistency between multiple copies can be ensured.

Use multi-node management methods for ocean spatiotemporal data, including spatiotemporal databases based in part on traditional RDBMS, and new data processing methods. This new approach to data processing was proposed by Carlo Strozzi [72] in 1998 and called it NoSQL. It is a brand-new database revolution, which advocates the use of non-relational data storage, which is very suitable for the semi-structure of ocean space-time data, unstructured data format and large amount of data. Although NoSQL can not completely replace traditional RDBMS, it has a very wide range of applications in the field of ocean spatiotemporal data storage and processing. For example, database Redis [73], Oracle NoSQL [74] based on key-value pairs. Column Family (Wide-Column) database Cassandra [75], HBase, etc. document database MongoDB [76], Couchbase [77], etc. graph database Nebula [78], Neo4j [79], etc.

For traditional RDBMS, researchers continue to provide new extensions to meet the processing needs of marine spatiotemporal data. In this article, we mainly introduce the
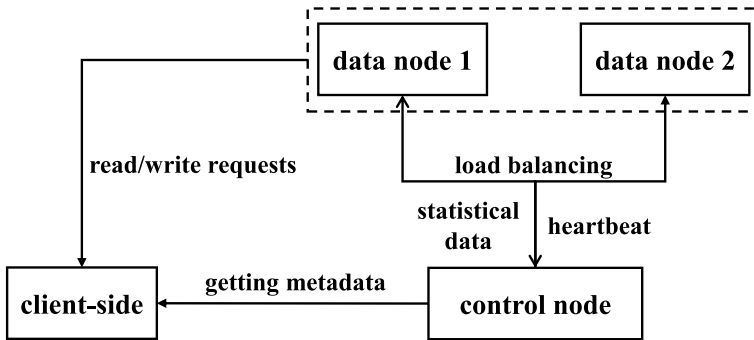
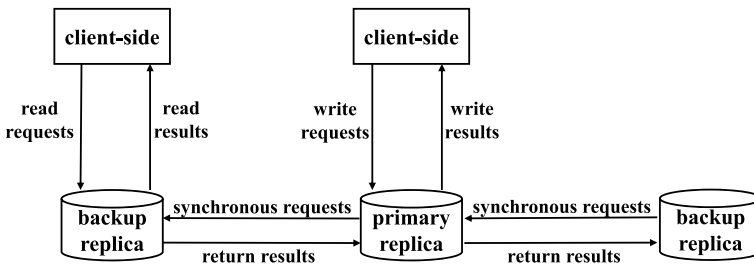**Figure 2** A cluster of distributed storage systems

Figure 3 Distributed storage system data replication

PostGIS extension of PostgreSQL, which supports OGC-compliant spatiotemporal SQL queries. Horizontal sharding of ocean spatiotemporal data to enable horizontal scaling when ocean spatiotemporal data exceeds the capacity of a single node, and read scalability can also be achieved by leveraging pgpool (Pgpool-II [80]) and streaming replication. However, ocean spatiotemporal data can be distributed among multiple nodes through data sharding, which can effectively reduce the I/O bottleneck [81]. Balancing I/O bottleneck between single machines in microservices can help in reaching Quality-of-Service (QoS) [82] goals.

In addition, there are several ways to achieve horizontal scaling and parallel acceleration of queries through data sharding, for example PostGIS can integrate with Citus and PostgresXL [83], or use PL/Proxy [84], etc. PostgreSQL has added a built-in sharding function after version 9.6, called foreign data wrapper (FDW), which enables PostgreSQL to access data from external source data. Therefore, ocean spatiotemporal data can be stored on different nodes of the cluster in a distributed manner, where each data partition can be accessed directly from disk or main memory through FDW. In addition, in the latest PostgreSQL 12, the use of PostGIS 3.0 can support functions such as parallel sequence scan, parallel join, and parallel aggregation for parallel spatial query processing.

Traditional RDBMSs store data in tables, and it is difficult to support today's marine spatiotemporal data in multiple formats from many different sources. However, researchers added the JSON and JSONB data types to PostgreSQL support in 2012 and 2014, respectively. Also, SQL/JSON compliant with the SQL-2016 standard was introduced in

the latest PostgreSQL 12. So now we can query and index ocean spatiotemporal data using JSON and JSONB in PostgreSQL [85].

Finally, spatiotemporal databases based on traditional RDBMS can consider using distributed file systems such as HDFS to support distributed processing capabilities for themselves, or use in-memory computing frameworks such as Spark [86] and Flink [87] to accelerate computing.

NoSQL database system has added distributed support at the beginning of its design, which has many advantages such as fault tolerance, scalability, high availability, and high flexibility. Currently, NoSQL that supports spatiotemporal data storage and processing includes Redis, Oracle NoSQL, MongoDB, Couchbase, Neo4j, Nebula, TigerGraph, Cassandra, etc. Among them, Redis is a key-value storage system, which operates based on the Geo Set data structure constructed by Sorted Set, and implements a geohash spatial index that can speed up query processing. Oracle NoSQL supports a SQL-like query language that supports all common geometry objects, geohash indexes, and a set of operators for working with spatiotemporal data.

Couchbase and MongoDB are a distributed document-oriented high-performance NoSQL database management system that natively supports processing spatiotemporal data. Both Couchbase's GeoCouch [88] extension and MongoDB support common Geo-JSON objects such as point, linestring, polygon, and collections. Couchbase's GeoCouch extension is developed based on T-Trees, allowing BBox to perform spatiotemporal queries and supporting SQL-like The query language N1QL. MongoDB does not have a SQL-like query language, but provides a set of spatiotemporal operators such as nearSphere, geoIntersect and geoNear to perform spatial queries.

Nebula Graph [78] is an open-source, distributed, and easily scalable native graph database that can carry ultra-large datasets with hundreds of billions of points and trillions of edges, and provides millisecond-level queries. It adopts a shared-nothing architecture. It supports scaling up and down without stopping the database service. It introduced full support for Geospatial Data in version 2.6, including storage, computation, and indexing of oceanic spatiotemporal data. Nebula Graph currently supports marine spatiotemporal data of the Geography type, which models geographic location information represented by pairs of latitude and longitude coordinates in the earth space coordinate system. It also supports the efficient SQL-like query language nGQL. It also supports spatiotemporal function query operations (contain, cover, intersect, and so on) on common geometric objects (point, linestring, polygon, and collections).

6 Performance evaluation on spatiotemporal ocean data

In terms of the massive volume of spatiotemporal data, the processing of spatiotemporal data becomes a key problem. Performance evaluation on saptiotemporal data mainly considers its interactive performance, which reflects in response time, and system scalability.

In [89], an evaluation on distributed spatial database GeoMesa and ElasticSearch is conducted based on number of records returned and response time concerning number of records, area of query polygons and size of temporal window, respectively, where the results show that GeoMesa queries outperform ElasticSearch queries. Yu et al. [90] implemented a spatiotemporal computing frame work GeoSpark and proves it outperforms SpatialHadoop in spatial co-location, in terms of response time. Researchers also design benchmarks specially for evaluation of spatiotemporal databases such as SEQUOIA [91]

and Paradise Geo-Spatial [92]. Makris et al. [93] evaluated the spatiotemporal data performance of NoSQL database MongoDB and open source RDBMS-PostgreSQL, where results reveal a better performance of PostgreSQL in all queries compared with MongoDB. In [94], researchers conduct performance evluation on five Spark based spatial analytics systems (Magellan, SpatialSpark, Simba, LocationSpark, GeoSpark) with different spatial queries and datatypes. Among these, GeoSpark was proved to be the most complet spatial analytic system with all queries and data types supported.

7 Summary

The dramatically high-rate growth of ocean data have lead to the challenge of ocean data processing. The inherent heterogeneity, spatiotemporal involved and constant changing of ocean data lead to difficulty in representing ocean data as well. While traditional methods no longer functional satisfy the ocean data processing demand, graph-based structure of data processing are extensively adopted. In this survey, we systematically illustrate the processing methods on ocean spatiotemporal data. We discuss about data representation

Table 1 Summary of data processing methods on ocean spatiotemporal data

Name	Method & Type	Contributes in
SWEET [21]	Ontology extension	Representation, querying
SEW [22]	Semantic primitive invention	Situated data management
Janowicz et al. [16]	Semantic extension	SDI
CRISIS [24, 25]	Semantic web tech	Knowledge interoperability
GeoKG [26]	Ontology design	Representation
OEDO [26]	Ontology design	Representation, querying
Bittner [34]	Axiomatic formalisation	Geo-entity relations
YAGO2 [36]	Ontology extension	Representation, integration
Timely YAGO [38]	Knowledge extraction	Representation, querying
Vasseur et al. [39]	Ontology design	Integration
Hornsby et al. [41]	Semantic primitive	Representation, reasoning
Kurte et al. [40]	Ontology design	Disaster monitoring
Neumaier[42]	Entity extension	Representation, querying
Grenon [43]	Ontology design	Representation, reasoning
Carstensen [44]	Ontology design, selective attention	Representation
Kauppinen et al. [45]	Ontology design	Semantic ambiguation
SWETO [47]	Ontology design	Entity extraction
Jayawardhana et al. [49]	Semantic mapping	Entity extraction
Rule-based [51, 52]	Supervise learning	Relation extraction
Kernel-based [53]	Supervise learning	Relation extraction
Chen et al. [54]	Semi-supervised learning	Relation extraction
Lu et al. [55]	Unsupervised learning	Relation extraction
Zhou et al. [57]	Data cleaning	Knowledge fusion
Raffaeta et al. [59]	Ontology design	Knowledge computing
Batsakis [61]	Ontology design	Knowledge computing

methods, design and construction of ocean knowledge graphs. Main methods on spatiotemporal data representation and knowledge graph construction are summarized in the table below (Table 1). In addition, we compare different management techniques of ocean spatiotemporal knowledge as well as performance evaluation on ocean spatiotemporal data.

Author contributions Xiaoyong Li worked on the full manuscript, Jingyun Gu wrote the Section 2-4, Guolong Tan and Wenjing Jiang prepared Section 5-6 and Figure 1-3, Ao Cui and Leiming Shu worked on the Section 1. Kaijun Ren, Haoyang Zhu and Jedi S. Shang participated in proofreading. Zichen Xu contributed to the proofreading work as well as Abstract and Section 7.

Data availability Not applicable.

Declarations

Human and animal ethics Not applicable.

Ethics approval and consent to participate Not applicable.

Consent for publication Not applicable.

Competing interests We declare that authors have no known competing interests or personal relationships that might be perceived to influence the discussion reported in this paper.

References

1. Chung, C.-J.F., Fabbri, A.G.: The representation of geoscience information for data integration. Nonrenewab. Resour. **2**(2), 122–139 (1993)
2. Berners-Lee, T., Hendler, J., Lassila, O.: The semantic web. Sci. Amer. **284**(5), 34–43 (2001)
3. Sowa, J.F.: Semantic networks (1987)
4. Shortliffe, E.: Computer-based Medical Consultations: MYCIN, vol. 2. Elsevier (2012)
5. Group, R.W.: Resource Description Framework (RDF). https://www.w3.org/RDF/ Accessed 02 Feb 2014
6. Staab, S., Studer, R.: Handbook on Ontologies. Springer (2010)
7. Lenat, D.B.: Cyc: A large-scale investment in knowledge infrastructure. Commun. ACM **38**(11), 33–38 (1995)
8. Bollacker, K., Evans, C., Paritosh, P., Sturge, T., Taylor, J.: Freebase: A collaboratively created graph database for structuring human knowledge. In: Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data, pp 1247–1250 (2008)
9. Lehmann, J., Isele, R., Jakob, M., Jentzsch, A., Kontokostas, D., Mendes, P.N., Hellmann, S., Morsey, M., Van Kleef, P., Auer, S., et al: Dbpedia–a large-scale, multilingual knowledge base extracted from wikipedia. Sem. Web **6**(2), 167–195 (2015)
10. Suchanek, F.M., Kasneci, G., Weikum, G.: Yago: A core of semantic knowledge. In: Proceedings of the 16th International Conference on World Wide Web, pp 697–706 (2007)
11. Nakashole, N., Theobald, M., Weikum, G.: Scalable knowledge harvesting with high precision and high recall. In: Proceedings of the Fourth ACM International Conference on Web Search and Data Mining, pp 227–236 (2011)

12. Singhal, A.: Introducing the Knowledge Graph: Things, Not Strings. https://blog.google/products/search/introducing-knowledge-graph-things-not.html Accessed: 2012

13. Dong, X., Gabrilovich, E., Heitz, G., Horn, W., Lao, N., Murphy, K., Strohmann, T., Sun, S., Zhang, W.: Knowledge vault: A web-scale approach to probabilistic knowledge fusion. In: Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp 601–610 (2014)

14. James, P.: Knowledge graphs. In: Invan de Riet, R., Meersman, R. (eds.) . 1991 Workshop on Linguistic Instruments in Knowledge Engineering ; Conference date: 17-01-1991 Through 18-01-1991, pp 97–117. Elsevier (1992)

15. Stadler, C., Lehmann, J., Höffner, K., Auer, S.: Linkedgeodata: A core for a web of spatial open data. Sem. Web **3**(4), 333–354 (2012)

16. Janowicz, K., Schade, S., Bröring, A., Keßler, C., Maué, P., Stasch, C.: Semantic enablement for spatial data infrastructures. Trans. GIS **14** (2), 111–129 (2010)

17. Dykes, J.A.: Exploring spatial data representation with dynamic graphics. Comput. Geosci. **23**(4), 345–370 (1997)

18. Shepherd, I.: Putting time on the map: Dynamic displays in data visualization and GIS. In: Fisher, PF (ed.) Innovations in GIS, vol. 2. Taylor & Francis, London (1995)

19. Tuohy, S.T., Patrikalakis, N.M.: Non-linear data representation for ocean exploration and visualization. J. Vis. Comput. Animat. **7**(3), 125–139 (1996)

20. Bibby, C., Reid, I.: A hybrid slam representation for dynamic marine environments. In: 2010 IEEE International Conference on Robotics and Automation, pp 257–264. IEEE (2010)

21. Raskin, R., Pan, M.: Semantic web for earth and environmental terminology (sweet). In: Proc. of the Workshop on Semantic Web Technologies for Searching and Retrieving Scientific Data, vol. 25 (2003)

22. MacGregor, R.M., Ko, I.-Y.: Representing contextualized data using semantic web tools. In: PSSS (2003)

23. Frank, A.U.: Ontology for spatio-temporal databases. In: Spatio-temporal Databases, pp. 9–77. Springer (2003)

24. Dividino, R., Soares, A., Isenor, A., Webb, S., Brousseau, M.: Semantic integration of real-time heterogeneous data streams for ocean-related decision making. https://doi.org/10.14339/STO-MP-IST-160-S1-3-PDF (2018)

25. Soares, A., Dividino, R., Abreu, F., Brousseau, M., Isenor, A.W., Webb, S., Matwin, S.: Crisis: Integrating ais and ocean data streams using semantic web standards for event detection. In: 2019 International Conference on Military Communications and Information Systems (ICMCIS), pp. 1–7. IEEE (2019)

26. Wang, S., Zhang, X., Ye, P., Du, M., Lu, Y., Xue, H.: Geographic knowledge graph (geokg): A formalized geographic knowledge representation. ISPRS Int. J. Geo-Inform. **8**(4), 184 (2019)

27. Ren, X.-L., Ren, K.-J., Xu, Z.-C., Li, X.-Y., Zhou, A.-L., Song, J.-Q., Deng, K.-F.: Improving ocean data services with semantics and quick index. J. Comput. Sci. Technol. **36**(5), 963–984 (2021)

28. Russell, S., Norvig, P.: Artificial intelligence: A modern approach (2002)

29. Gruber, T.R.: A translation approach to portable ontology specifications. Knowl. Acquis. **5**(2), 199–220 (1993)

30. Arvidsson, F., Flycht-Eriksson, A.: Ontologies i. (PDF). http://www.ida.liu.se/janma/SemWeb/Slides/ontologies1.pdf. Retrieved 26 (2008)

31. Gómez-Pérez, A.: Knowledge sharing and reuse. In: The Handbook of Applied Expert Systems, pp. 10–1. CRC Press (2019)

32. Gruber, T.R.: Toward principles for the design of ontologies used for knowledge sharing? Int. J. Human-Comput. Stud. **43**(5-6), 907–928 (1995)

33. Galton, A.: Desiderata for a spatio-temporal geo-ontology. In: International Conference on Spatial Information Theory, pp. 1–12. Springer (2003)

34. Bittner, T., Donnelly, M., Smith, B.: A spatio-temporal ontology for geographic information integration. Int. J. Geogr. Inf. Sci. **23**(6), 765–798 (2009)

35. Bittner, T., Smith, B.: Granular spatio-temporal ontologies. In: Proceedings of the AAAI Spring Symposium on Foundations and Applications of Spatio-temporal Reasoning, pp 12–17 (2003)

36. Hoffart, J., Suchanek, F.M., Berberich, K., Lewis-Kelham, E., De Melo, G., Weikum, G.: Yago2: Exploring and querying world knowledge in time, space, context, and many languages. In: Proceedings of the 20th International Conference Companion on World Wide Web, pp 229–232 (2011)

37. Miller, G.A.: WordNet: An Electronic Lexical Database. MIT Press (1998)

38. Wang, Y., Zhu, M., Qu, L., Spaniol, M., Weikum, G.: Timely yago: Harvesting, querying, and visualizing temporal knowledge from wikipedia. In: Proceedings of the 13th International Conference on Extending Database Technology, pp 697–700 (2010)

39. Vasseur, B., Van de Vlag, D., Stein, A., Jeansoulin, R., Dilo, A.: Spatio-temporal ontology for defining the quality of an application. In: Proceedings of ISSDQ, Bruck an der Leitha, pp 15–17, Austria (2004)

40. Kurte, K.R., Durbha, S.S., King, R.L., Younan, N.H., Potnis, A.V.: A spatio-temporal ontological model for flood disaster monitoring. In: 2017 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), pp. 5213–5216. IEEE (2017)

41. Hornsby, K., Egenhofer, M.J.: Identity-based change: A foundation for spatio-temporal knowledge representation. Int. J Geograph. Inform. Sci. **14**(3), 207–224 (2000)

42. Neumaier, S., Polleres, A.: Enabling spatio-temporal search in open data. J. Web Sem. **55**, 21–36 (2019)

43. Grenon, P.: The Formal Ontology of Spatio-Temporal Reality and its Formalization. AAAI Press, Amsterdam (2003)

44. Carstensen, K.-U.: Spatio-temporal ontologies and attention. Spatial Cogn. Comput. **7**(1), 13–32 (2007)

45. Kauppinen, T., Henriksson, R., Sinkkilä, R., Lindroos, R., Väätäinen, J., Hyvönen, E.: Ontology-Based Disambiguation of Spatiotemporal Locations. In: IRSW. Citeseer (2008)

46. Grenon, P., Smith, B.: Snap and span: Towards dynamic spatial ontology. Spatial Cogn. Comput. **4**(1), 69–104 (2004)

47. Budak Arpinar, I., Sheth, A., Ramakrishnan, C., Lynn Usery, E., Azami, M., Kwan, M.-P.: Geospatial ontology development and semantic analytics. Trans. GIS **10**(4), 551–575 (2006)

48. Niu, F., Zhang, C., Ré, C., Shavlik, J.W.: Deepdive: Web-scale knowledge-base construction using statistical learning and inference. VLDS **12**, 25–28 (2012)

49. Jayawardhana, U.K., Gorsevski, P.V.: An ontology-based framework for extracting spatio-temporal influenza data using twitter. Int. J. Gigit. Earth **12**(1), 2–24 (2019)

50. Strötgen, J., Gertz, M., Popov, P.: Extraction and exploration of spatio-temporal information in documents. In: Proceedings of the 6th Workshop on Geographic Information Retrieval, pp. 1–8 (2010)

51. Zhang, C., Zhang, X., Jiang, W., Shen, Q., Zhang, S.: Rule-based extraction of spatial relations in natural language text. In: 2009 International Conference on Computational Intelligence and Software Engineering, pp. 1–4. IEEE (2009)

52. Mirza, P., Tonelli, S.: Catena: Causal and temporal relation extraction from natural language texts. In: The 26th International Conference on Computational Linguistics, pp. 64–75. ACL (2016)

53. Qiu, Q., Xie, Z., Wu, L., Tao, L.: Automatic spatiotemporal and semantic information extraction from unstructured geoscience reports using text mining techniques. Earth Sci. Inform. **13**(4), 1393–1410 (2020)

54. Chen, Y., Sun, Q.L., Zhong, K.: Semi-supervised spatio-temporal cnn for recognition of surgical workflow. EURASIP J. Image Video Process. **2018** (1), 1–9 (2018)

55. Lu, P.Y., Kim, S., Soljačić, M.: Extracting interpretable physical parameters from spatiotemporal systems using unsupervised learning. Phys. Rev. X **10**(3), 031056 (2020)

56. Mesquita, F., Schmidek, J., Barbosa, D.: Effectiveness and efficiency of open relation extraction. In: Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, pp 447–457 (2013)

57. Zhou, H., Li, M., Gu, Z., Tian, Z.: Spatiotemporal data cleaning and knowledge fusion. In: MDATA: A New Knowledge Representation Model, pp. 32–50. Springer (2021)

58. Zhou, H., Li, M., Gu, Z.: Knowledge fusion and spatiotemporal data cleaning: A review. In: 2020 IEEE Fifth International Conference on Data Science in Cyberspace (DSC), pp. 295–301. IEEE (2020)

59. Raffaetà, A., Turini, F., Renso, C.: Enhancing giss for spatio-temporal reasoning. In: Proceedings of the 10th ACM International Symposium on Advances in Geographic Information Systems, pp 42–48 (2002)

60. Mantle, M., Batsakis, S., Antoniou, G.: Large scale distributed spatio-temporal reasoning using real-world knowledge graphs. Knowl.-Based Syst. **163**, 214–226 (2019)

61. Batsakis, S., Petrakis, E.G.: Sowl: Spatio-temporal representation, reasoning and querying over the semantic web. In: Proceedings of the 6th International Conference on Semantic Systems, pp 1–9 (2010)

62. Strobl, C.: Postgis 891–898 (2008)

63. OracleSpatialTeam: Oracle Spatial and Graph Features. https://www.oracle.com/database/techn ologies/spatialandgraph.html

64. Adler, D.W.: Db2 spatial extender-spatial data within the rdbms. In: VLDB, pp. 687–690. Roma (2001)

65. Furieri, A.: Spatialite. linha] Disponível em: https://www.gaiagis.it/fossil/libspatialite/index. [Acedido: 30-Nov-2015] (2014)

66. MySQLTeam: MySQL 8.0 Reference Manual. https://dev.mysql.com/doc/refman/8.0/en/
67. Fang, Y., Friedman, M., Nair, G., Rys, M., Schmid, A.-E.: Spatial indexing in microsoft sql server 2008. In: Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data, pp 1207–1216 (2008)
68. Wang, P., Wang, J., Li, C., Wang, J., Zhu, H., Guo, M.: Grus: Toward unified-memory-efficient high-performance graph processing on gpu. ACM Trans. Arch. Code Optim. (TACO) **18**(2), 1–25 (2021)
69. Zhang, W., Chen, Q., Zheng, N., Cui, W., Fu, K., Guo, M.: Towards qos-awareness and improved utilization of spatial multitasking gpus. IEEE Transactions on Computers (2021)
70. OGC: OGC Simple Feature Access - Part 2: SQL. https://www.ogc.org/standards/sfs
71. Stolze, K.: The standard to manage spatial data in relational database systems. In: Memorias de 10th Conference on Database Systems for Busines, Technology and Web (2003)
72. Han, J., Haihong, E., Le, G., Du, J.: Survey on nosql database. In: 2011 6th International Conference on Pervasive Computing and Applications, pp. 363–366. IEEE (2011)
73. Salvatore, S., Pieter, N., Matt, S.: Redis: An in-memory database that persists on disk (2011)
74. Narayanam, S., Wang, S.: Oracle nosql database (2016)
75. Cassandra, A.: Manage massive amounts of data, fast, without losing sleep. Cassandra. apache org (2015)
76. Mongo, D.: Mongodb. https://docs.mongodb.com/manual/geospatial-queries/(2015)
77. Lorezno, C.M., Mata, P.M.: Couchbase (2015)
78. Graph, N.: Open Source, Distributed, Scalable, Lightning Fast. https://nebula-graph.io/ (2022)
79. Webber, J.: A programmatic introduction to neo4j. In: Proceedings of the 3rd Annual Conference on Systems, Programming, and Applications: Software for Humanity, pp. 217–218 (2012)
80. PgpoolTeam: Pgpool-II: A middleware that works between PostgreSQL servers and a PostgreSQL database client. https://wiki.postgresql.org/wiki/Pgpool-{{II}} (2020)
81. Momjian, B.: The Future of Postgres Sharding. https://momjian.us/main/writings/pgsql/sharding.pdf (2021)
82. Fu, K., Zhang, W., Chen, Q., Zeng, D., Guo, M.: Adaptive resource efficient microservice deployment in cloud-edge continuum. IEEE Trans. Parallel Distrib. Syst. **33**(8), 1825–1840 (2022). https://doi.org/10.1109/TPDS.2021.3128037
83. Li-rong, A., Kai, L.: Study on optimization technology of data management based on postgres-xl. Computer Technology and Development (2018)
84. PL/ProxyTeam: PL/Proxy: Function-based sharding for PostgreSQL. https://plproxy.github.io/ (2022)
85. Korotkov., A.: The NoSQL Postgres. https://youtu.be/70dBszaO67Af (2019)
86. Spark, A.: Apache spark. Retrieved January **17**(2018), 1 (2018)
87. Carbone, P., Katsifodimos, A., Ewen, S., Markl, V., Haridi, S., Tzoumas, K.: Apache flink: Stream and batch processing in a single engine. Bulletin of the IEEE Computer Society Technical Committee on Data Engineering 36(4) (2015)
88. CouchDB, A.: Apache couchdb. https://couchdb.apache.org
89. Hulbert, A., Kunicki, T., Hughes, J.N., Fox, A.D., Eichelberger, C.N.: An experimental study of big spatial data systems. In: 2016 IEEE International Conference on Big Data (Big Data), pp. 2664–2671. IEEE (2016)
90. Yu, J., Wu, J., Sarwat, M.: Geospark: A cluster computing framework for processing large-scale spatial data. In: Proceedings of the 23rd SIGSPATIAL International Conference on Advances in Geographic Information Systems, pp 1–4 (2015)
91. Stonebraker, M., Frew, J., Gardels, K., Meredith, J.: The sequoia 2000 storage benchmark. ACM SIGMOD Record **22**(2), 2–11 (1993)
92. Patel, J., Yu, J., Kabra, N., Tufte, K., Nag, B., Burger, J., Hall, N., Ramasamy, K., Lueder, R., Ellmann, C., et al: Building a scaleable geo-spatial dbms: Technology, implementation, and evaluation. In: Proceedings of the 1997 ACM SIGMOD International Conference on Management of Data, pp 336–347 (1997)
93. Makris, A., Tserpes, K., Spiliopoulos, G., Anagnostopoulos, D.: Performance evaluation of mongodb and postgresql for spatio-temporal data. In: EDBT/ICDT Workshops (2019)
94. Pandey, V., Kipf, A., Neumann, T., Kemper, A.: How good are modern spatial analytics systems? Proc. VLDB Endow. **11**(11), 1661–1673 (2018)

Authors and Affiliations

Xiaoyong Li[1] · Jingyun Gu[2] · Guolong Tan[2] · Wenjing Jiang[2] · Ao Cui[2] · Leiming Shu[2] · Kaijun Ren[1,3] · Haoyang Zhu[4] · Jedi S. Shang[5] · Zichen Xu[2]

> Xiaoyong Li
> sayingxmu@nudt.edu.cn

> Jingyun Gu
> jgu@email.ncu.edu.cn

> Guolong Tan
> guolongtan@email.ncu.edu.cn

> Wenjing Jiang
> wenjingjiang@email.ncu.edu.cn

> Ao Cui
> aocui22@email.ncu.edu.cn

> Leiming Shu
> shuleiming@email.ncu.edu.cn

> Kaijun Ren
> renkaijun@nudt.edu.cn

> Jedi S. Shang
> shangshuo1@thinvent.com

[1] College of Meteorology and Oceanography, National University of Defensive Technology, Changsha 410073, China

[2] College of Mathematics and Computer Science, Nanchang University, Nanchang 330031, China

[3] College of Computer Science and Technology, National University of Defensive Technology, Changsha 410073, China

[4] Institute of Systems Engineering, Beijing 100000, China

[5] Thinvent Technology Co. LTD., Nanchang 330000, China