Communication

Construction and Application of a Knowledge Graph

Xuejie Hao [1], **Zheng Ji** [2], **Xiuhong Li** [1,*], **Lizeyan Yin** [3], **Lu Liu** [1], **Meiying Sun** [1], **Qiang Liu** [1] **and Rongjin Yang** [4]

1. College of Global Change and Earth System Science, Beijing Normal University, Beijing 100875, China; 202031490007@mail.bnu.edu.cn (X.H.); 201821490033@mail.bnu.edu.cn (L.L.); mysun@mail.bnu.edu.cn (M.S.); toliuqiang@bnu.edu.cn (Q.L.)
2. School of Remote Sensing and Information Engineering, Wuhan University, Wuhan 430079, China; jz07@whu.edu.cn
3. Institute of Computing, Modeling and Their Applications, Clermont-Auvergne University, 63000 Clermont-Ferrand, France; Lizeyan.YIN@etu.uca.fr
4. Chinese Research Academy of Environmental Sciences, No. 8, Da Yang Fang, An Wai, Chao Yang District, Beijing 100012, China; yangrj@craes.org.cn
* Correspondence: lixh@bnu.edu.cn; Tel.: +86-136-2116-6693

Abstract: With the development and improvement of modern surveying and remote-sensing technology, data in the fields of surveying and remote sensing have grown rapidly. Due to the characteristics of large-scale, heterogeneous and diverse surveys and the loose organization of surveying and remote-sensing data, effectively obtaining information and knowledge from data can be difficult. Therefore, this paper proposes a method of using ontology for heterogeneous data integration. Based on the heterogeneous, decentralized, and dynamic updates of large surveying and remote-sensing data, this paper constructs a knowledge graph for surveying and remote-sensing applications. First, data are extracted. Second, using the ontology editing tool Protégé, a knowledge graph mode level is constructed. Then, using a relational database, data are stored, and a D2RQ tool maps the data from the mode level's ontology to the data layer. Then, using the D2RQ tool, a SPARQL protocol and resource description framework query language (SPARQL) endpoint service is used to describe functions such as query and reasoning of the knowledge graph. The graph database is then used to display the knowledge graph. Finally, the knowledge graph is used to describe the correlation between the fields of surveying and remote sensing.

1. Introduction

In 2012, Google officially proposed the concept of the knowledge graph, which aims to assist intelligent search engines [1]. After the knowledge graph was formally proposed, it was quickly popularized in academia and industry and was widely used in intelligent search, personalized recommendation, intelligence analysis, anti-fraud and other fields. Essentially, a knowledge graph is a semantic network and knowledge base with a directed graph structure that describes entities (concepts) and their relationships in the physical world in symbolic form. The knowledge graph is represented in the form of triples (Entity1-Relation-Entity2), where the nodes of the graph represent entities or concepts, and the edges represent the relationships between entities or concepts [2].

Knowledge graphs are a new method of knowledge representation. In essence, the semantic web is an early form of the knowledge graph, which is an abstract concept that describes entities and relationships between entities in the objective world and is also a networked knowledge base composed of entities, properties, and relationships. A knowledge graph is a collection of concepts, entities, and their relationships in the abstract physical world [3]. The knowledge graph has changed the traditional method of information retrieval. On the one hand, knowledge graphs describe the semantic and attribute

relationship between concepts to reason about concepts through fuzzy string matching. Conversely, knowledge graphs display the structured knowledge of classification and arrangement to users through the grid graphic information display interface. Concurrently, knowledge graphs solve the problem of manual filtering of useless information, which has practical significance for an intelligent society [4].

Knowledge graphs can be divided into general knowledge graphs and domain knowledge graphs. Knowledge graphs used in surveying and remote sensing are domain knowledge graphs. To date, few studies have investigated knowledge graphs for surveying and remote sensing. Wang and others proposed a framework for remote-sensing interpretation of knowledge graphs [5]. Xie and others designed a framework for the construction of a large knowledge graph in the field of remote-sensing satellites [6]. Geoscience knowledge graphs are also used and have been studied in detail. Xu and others proposed the concept, framework, theory, and characteristics of geoscience graphs based on geoscience graphs and geoscience information graphs. Jiang proposed the process of constructing knowledge graphs and explored the key technology of geographic knowledge graphs [7]. Lu and others systematically reviewed the research progress on topics related to geographic knowledge graphs and analysed the key issues of current geoscience knowledge graph construction [8]. However, these studies all discuss the construction of knowledge graphs in theory and do not provide any real examples of constructing knowledge graphs. Based on previous research experience, this paper describes an example of constructing knowledge graphs for surveying and remote sensing.

The construction of this knowledge graph can provide services for users of surveying and remote sensing, surveying and remote-sensing experts, and developers of surveying and remote-sensing software. Users can visualize knowledge through the knowledge graph and discover the relationship between knowledge more easily. Searches conducted through the knowledge graph improve the user's search efficiency. Remote-sensing experts gain insights and discover new rules in the field of surveying and remote sensing through the inference function of the knowledge graph. Software developers can integrate the knowledge graph into the remote-sensing product e-commerce platform, which can not only improve search efficiency, but also accurately recommend products for users. The professional field of surveying and remote sensing is the ladder of social progress. With the development of social intelligence, it is significant to study professional development technology for intelligent surveying and remote sensing.

2. The Theoretical Basis for the Construction of Knowledge Graphs

The system framework of a knowledge graph in the fields of surveying and remote sensing primarily refers to its construction mode structure, which describes the process of constructing a knowledge graph in the fields of surveying and remote sensing (Figure 1). The process of knowledge graphing in the field of surveying remote sensing can be divided into two parts: mode level construction and data layer construction.

The mode level is built on the data layer, and the factual expression in the data layer is standardized through the ontology library. The ontology is the conceptual template of the structured knowledge base. The knowledge base constructed through the ontology library has the advantages of a strong hierarchical structure and low redundancy. The data layer is composed of a series of knowledge entities or concepts. Knowledge is stored in units of facts, and the data layer expresses knowledge in the form of triples (Entity 1-Relation-Entity 2) or (Entity-Attribute-Attribute Value). The logical structure of the knowledge graph in the field of surveying and remote sensing is shown in Figure 2.
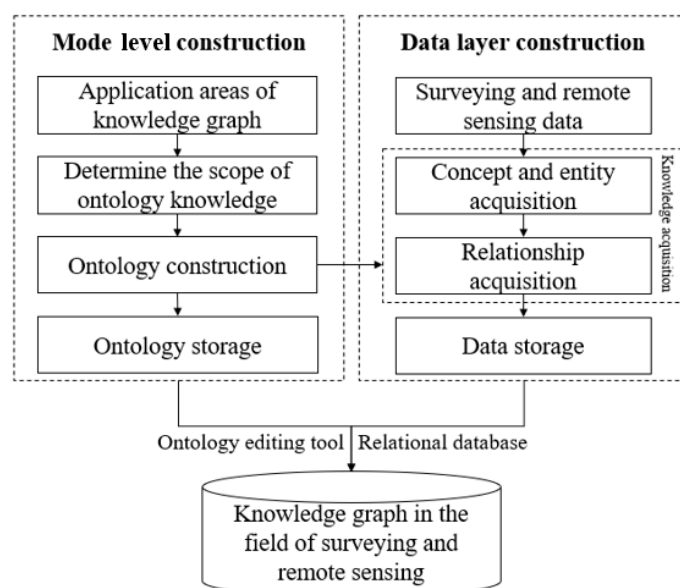
Figure 1. The framework of the knowledge graph in the fields of surveying and remote sensing.
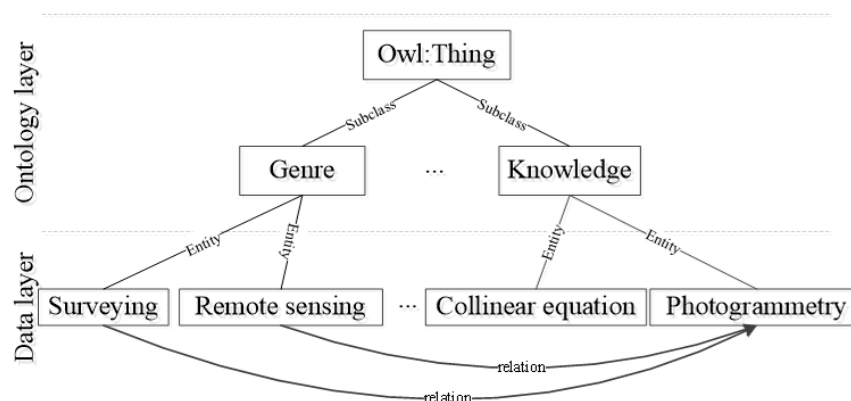

Figure 2. The logical structure of the knowledge graph in the fields of surveying and remote sensing.

2.1. Design of Disciplinary Knowledge Graph Mode in the Field of Surveying and Remote Sensing

The mode level construction process begins from the application domain of the knowledge graph, determines the knowledge scope of the ontology, and then constructs the ontology. The construction of the ontology is the key link in the construction of the pattern layer. The ontology is an explicit description of the shared conceptual mode and is used to add semantics to semantic web pages and describe the relationship between concepts [9]. The ontology is the pattern layer, conceptual mode, and logical basis of the knowledge graph. The mode level of the knowledge graph in the field of surveying and remote sensing uses ontology to realize the storage and management of knowledge. When constructing an ontology, we mainly construct a collection of concepts from the subject words of remote-sensing teaching materials. The relationship is mainly a hierarchical relationship between the upper and lower positions. Entity filling is mainly obtained from structured data sources, and the ontology database is filled in a bottom-up manner.

A domain ontology is a specialized ontology that describes concepts in a specific domain (such as remote sensing, meteorology, environment, etc.) and the relationship between concepts. The domain ontology of surveying and remote sensing, as a kind of unique ontology, can clearly describe the relationship between concepts belonging to and concepts in this domain. The basic principles of ontology construction are clarity, objectivity, consistency, minimum coding deviation, and minimum ontology constraints.

Ontology construction uses a seven-step method. The steps of this method are as follows: (1) determine the professional field and category of the ontology; (2) examine the possibility of reusing existing ontology; (3) list the important terms in the ontology; (4) define the hierarchical relationship between classes; (5) define the properties of the class; (6) define the constraints between the properties; and (7) create an instance [10].

2.2. The Data Layer of the Knowledge Graph in the Field of Surveying and Remote Sensing
2.2.1. Data Layer Relational Structure Design

In this article, the data extraction method is used to construct the data layer. When extracting related entities, the relationship between the entities in the field of surveying and remote sensing is first defined. The relationship between entities is mainly the upper–lower relationship and the non-upper–lower relationship. Research on non-subordinate relationships in the knowledge graphs focuses on two aspects: entity attributes and entity relationships. The non-subordinate relationship of entity attributes is mainly used for triples: entity-attribute-attribute value, where the attribute depends on the corresponding entity. Each attribute will have its corresponding attribute value. In the definition of entity relationships, there is always a direct or indirect relationship between entities. Through the relationship analysis between entities, the general relationship between the entities used is defined. The relationship between entities mainly includes the relationship of belonging, containing, etc. These relationships are common relationships among entities in the field of surveying and remote sensing.

2.2.2. Data Layer Construction

The data layer construction process is based on unstructured data and uses manual, automatic, or semiautomatic techniques to extract knowledge from the data and store it in the database. Data acquisition is a key step in the construction of the data layer. The core of knowledge acquisition is how to automatically obtain knowledge elements of structured information, such as entities, relationships, and properties, from unstructured and semi-structured data sources. Typically, automatic or semi-automatic machine learning technology is used to extract entities, relationships, attributes, and other information about the knowledge graph from open multisource data [11]. Knowledge acquisition includes the extraction of entities, relationships, properties, etc. Entity acquisition is the process of automatically identifying called entities (knowledge points, type names, etc.) from text data sets. Relation extraction is the process of discovering semantic relationships between entities from data sources using methods such as machine learning. Attribute extraction is the process of extracting attribute information about entities from data sources. The difference between attribute extraction and relation extraction is not only to identify the attribute name of the entity but also to identify the attribute value of the entity. Therefore, most studies are based on rules for extraction. [12].

3. Construction of Knowledge Graphs

Construction of knowledge graphs is the core content of this article. It includes five parts: data acquisition and storage, ontology construction and storage, ontology and database mapping, query and reasoning of knowledge graphs, and visualizing the knowledge graph on Neo4j.

3.1. Data Acquisition and Storage
3.1.1. Data Acquisition

The data used in this article are from surveying and remote sensing. The data source is a textbook related to the field of surveying [13] and remote sensing [14]. DeepDive (http://deepdive.stanford.edu/, accessed on 6 May 2021) was used to extract semi-structured table data from unstructured text data [15]. DeepDive can extract structured data from unstructured data and perform a series of data processing steps to build a knowledge base and extract relationships. It is very good at handling data sources in different formats.

DeepDive has good database support, supports PubMed and other database data sources, and is a data source that has been processed in natural language. DeepDive has established a framework to standardize the construction process of the knowledge base system and allows users to design their own extractors and markers according to the knowledge base that needs to be constructed. The relationship extraction process of DeepDive is shown in Figure 3. The DeepDive based domain text knowledge extraction method includes the following steps [16]:

(1) Data processing. First, the original corpus will be loaded. The natural language processing (NLP) tag is added. A set of candidate relationships and the sparse feature representation of each candidate relationship are extracted.

(2) Remote supervision of data and rules, and then various strategies will be used to supervise the data set so that we can use machine learning to learn the weight of the mode.

(3) Learning and inference: mode specification. Then, the advanced configuration of the mode will be specified.

(4) Error analysis and debugging. Finally, we will show how to use DeepDive's tags, error analysis and debugging tools.
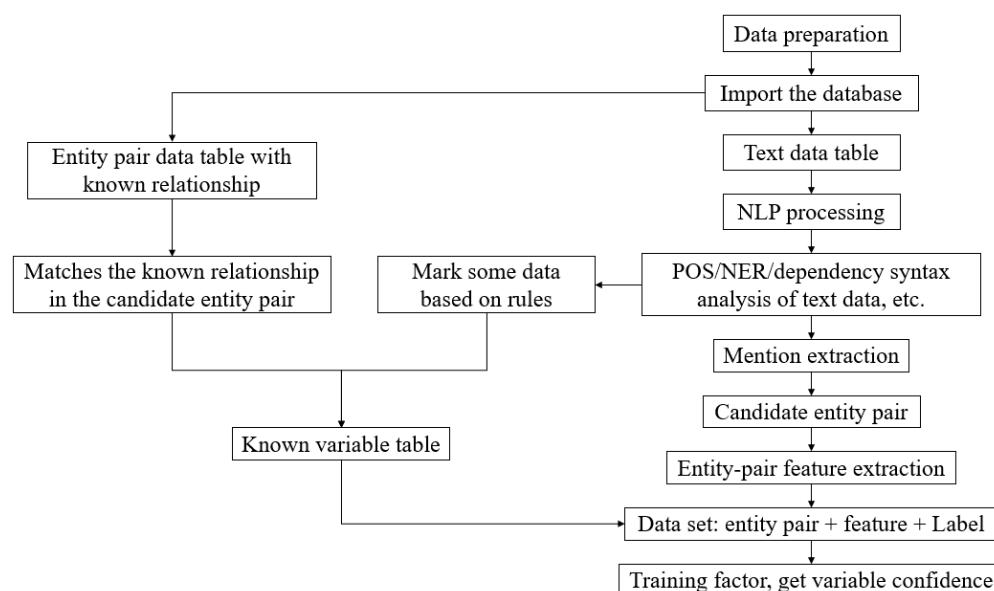

Figure 3. The relationship extraction process of DeepDive [15].

The extraction process of domain knowledge based on DeepDive:

(1) Experiment preparation (preparation before knowledge extraction). First, DeepDive stores the involved input, intermediate and output data in a relational database. DeepDive supports many databases, such as Postgres, Greenplum and MySQL. The database used in this experiment is Postgres.

(2) Data processing. This part is divided into four steps: ① Loading the original input data. First, we convert the target data into an electronic text format. Only the two fields of document id and document content are reserved. We store the cleaned data in a comma-separated values (CSV) format file. Then import the original text data into the associated database. We set the data format of the document storage in the app. ddlog file. There are two text fields in the articles table, articles (id text, content text). Then, we put the compressed file of the original data into the specified input folder and run the "DeepDive compile" command. Then, we execute the "DeepDive do articles" command, which imports the original data into the articles table of the associated database. At this time, we can query the imported raw data by executing the query command. ② Adding NLP markups. We use the CoreNLP

natural language processing system to add annotations to the original data. The steps of NLP are: first, the input original article is divided into sentences. The sentence is divided into words, and the part-of-speech tags, standard forms, dependency analysis and entity recognition tags of the words in the sentence are obtained. After NLP, some commonly used entities (person names, place names, etc.) can be marked. It is also necessary to ensure further entity identification of the data processed by NLP. The input are the data in the sentences table. The output are the marked data. Finally, we import the final marked data into the sentences_new table in the database. ③ Extracting candidate relation mentions. DeepDive proposes corresponding input and output interfaces, allowing users to design their entity or relationship extractors. Generally speaking, the SQL(Structured Query Language) statement is used as the input interface to extract data from the database; The output is the corresponding table in the database. We perform entity extraction on the data after entity recognition, and then establish the corresponding database table structure. ④ Extracting features for each candidate. First, we extract the feature description and store the feature in the func_feature table. The purpose is to use certain attributes or characteristics to represent each candidate pair. There is a library DDlib that can automatically generate features in DeepDive, which defines features that are not dependent on the domain. There are also many dictionaries in the DDlib library. These dictionaries contain words related to the correct classification of descriptions and relationships and are usually combined with domains and specific applications. We declare the extract_func_features function in app.ddlog. The input of this function includes the information of the two entities in the entity_mention table and the NLP results in the sentence where the two entities are located. The output is the two entities and their characteristics.

(3) Distant supervision with data and rules. We will use remote supervision to provide noisy label sets for candidate relationships to train machine learning models. Generally speaking, we divide the description method into two basic categories: mapping from secondary data for distant supervision and using heuristic rules for distant supervision [17]. However, we will use a simple majority voting method to solve the problem of multiple labels in each example. This method can be implemented in ddlog. In this method, first, we sum the labels (all −1, 0, or 1). Then, we simply threshold and add these labels to the decision variable table has_spouse. In addition, we also need to make sure that all the spouse candidates who are not marked with rules are not included in this table. Once again, we execute all the above.

(4) Learning and inference: model specification. We need to specify the actual model that DeepDive will perform learning and inference. DeepDive will learn the parameters of the model (the weights of features and the potential connections between variables). Then we perform statistical inferences on the learned model to determine that the probability of each variable of interest is true. ① Specifying prediction variables. In our experiment, we have a variable to predict the mention of each spouse candidate. In other words, we want DeepDive to predict the value of a Boolean variable for each mentioned spouse candidate to indicate whether the value is correct. DeepDive cannot only predict the value of these variables but also predict the marginal probability, that is, DeepDive's confidence in each prediction. ② Specifying features. We need to define the following: each has_spouse variable will be connected to the elements of the corresponding spouse_candidate row; We hope that DeepDive understands the weights of these elements from the data we remotely monitor; those weights of the element should be the same for the specific function of all instances. ③ Specifying connections between variables. We can use learning weights or given weights to specify the dependencies between predictors. In the experiment, we specify two such rules, which have fixed (given) weights. First, we define the asymmetric connection, that is, if the model considers that a person mentions p1 and another person mentions p2 as a spouse relationship in the sentence, then it should also consider the opposite.

The model should be strongly biased towards everyone mentioning a sign of marriage. Instead, we use negative weights for this operation. ④ Finally, we want to perform learning and inference using the specified model. This will build a model based on the data in the database, learn the weights, infer the expected or marginal probabilities of the variables in the model and then load it back into the database. In this way, we can see the probability of the has_spouse variable inferred by DeepDive.

(5) Error analysis and debugging. To accurately analyze the experimental results, we first declare a score or a user-defined query sentence and define the part of the labeled data used for training. DeepDive uses this score to estimate the accuracy of the experiment. We declare these definitions in deepdive.conf and define deepdive.calibration.holdout_fraction as 0.25. The test set is 75% of the labeled data, and the test set is used to verify the correctness of the experimental results. Approximately 1400 labeled data and approximately 1000 data were used for testing. The graph on the left in Figure 4 is the correct rate graph. Under ideal conditions, the red curve should be close to the blue calibration line. However, this is not the case. It may be caused by the sparseness and noise of the training data of the test data. The middle graph in Figure 4 is the predicted number graph of the test set. The forecasted quantity map usually presents a "U" shape. The graph on the right in Figure 4 is the predicted probability quantity graph of the entire data set. Among them, the prediction data falling in the 0.5–0.6 probability interval indicates that there are still some hidden types of instances, and the features of DeepDive are insufficient for these instances. The predicted data whose probability does not fall at (0, 0.1) or (0.9, 1.0) are the data to be extracted. An important indicator to improve the quality of the system is to re-speculate the above data and attribute it to the probability interval (0, 0.1) or (0.9, 1.0).
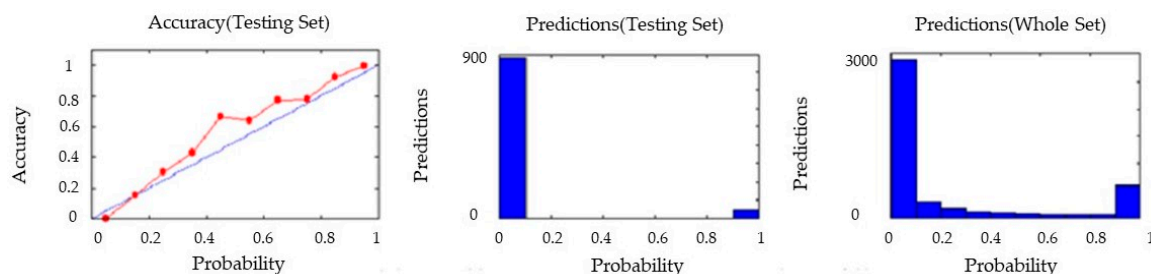

Figure 4. The test set correct rate calibration plot (**Left**), test set prediction plot (**Middle**), prediction plot for all data sets (**Right**).

3.1.2. Data Storage

The data of the knowledge graph is usually expressed in the form of triples, representing the relationship between entities or between entities and attribute values. In this article, the relational database MySQL and graph database Neo4j are used. The purpose of using MySQL is to realize the mapping between data and ontology and then to use existing tools to realize data query and reasoning. The purpose of using Neo4j is to update and search data. Neo4j can directly display the query results in the form of graphs to realize the function of knowledge visualization. Neo4j has local storage and data processing functions that are different from general databases, which can ensure high readability and integrity of data.

The extracted data are processed and stored in three tables, called "knowledge", "knowledge_to_genre", and "genre", which are imported into a MySQL database. The E-R(Entity Relationship) diagram is shown below (Figure 5).
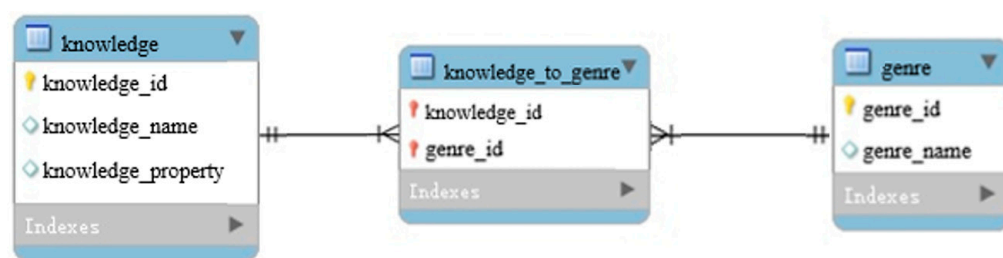
Figure 5. E-R diagram of data stored in the database.

In the entity table called "knowledge", each knowledge point is numbered, the attribute "knowledge_id" is created, and the attribute is used as the primary key. In the entity table called "genre", each category is numbered, the attribute "genre_id" is created, and the attribute is used as the primary key. In the relation table called "knowledge_to_genre", the properties "genre_id" and "knowledge_id" were set as the foreign keys of the entity tables "genre" and "knowledge" to create the relationship between genre and knowledge. For example, the knowledge "collinear equation" belongs to the field of "surveying"; "genre" is a "class". In the "genre" class, there are two entities: "surveying" and "remote sensing"; "genre" means the genre to which the knowledge node belongs. A knowledge node belongs to either the "surveying" genre or the "remote-sensing" genre.

3.2. Ontology Construction and Storage

3.2.1. Ontology Construction

There are two ways to construct ontologies: top-down and bottom-up. The ontology construction of the open domain knowledge graph typically uses a bottom-up method to automatically extract concepts, types of concept, and relationships between concepts from the knowledge graph. The open world is too complex to be considered in a top-down manner. As the world changes, the corresponding concepts are still growing. Most domain knowledge graph ontology construction uses a top-down approach. On the one hand, the concept and scope of the domain knowledge graph are fixed or controllable compared to the open domain knowledge graph; on the other hand, the domain knowledge graph must yield high-accuracy results. Currently, domain knowledge graphs are widely used in voice assistants [18]. These domain knowledge graphs can meet most user needs while ensuring accuracy.

This article uses a top-down approach to construct ontologies, and the creation tool uses Protégé (https://protege.stanford.edu, accessed on 6 May 2021), an ontology editing tool [19]. The specific creation process is described as follows. First, we create the ontology class and two classes of knowledge and genre. Note that all classes are subclasses of "Thing", and all classes must be mutually exclusive; an instance can only be one of the two classes. Second, the relationship between the classes (i.e., the object properties) is created. This article created the object attribute "belong_to" (We classify the knowledge extracted from the "Surveying" textbook into the "Surveying" class. We define the relationship between this knowledge and the class "surveying" as "belong_to". Similarly, we classify the knowledge extracted from the "Remote Sensing" textbook as "Remote Sensing" class. We also define the relationship between this knowledge and the class "Remote Sensing" as "belong_to") to indicate that a certain knowledge point is in a certain field (type). Therefore, its attribute "domain" is defined as the class "knowledge", and its attribute "range" is the class "genre". "Domain" indicates which class the attribute belongs to. "Range" represents the value range of the attribute, which defines the inverse of this attribute as "belong_to". Thus, ontology describes reasoning rules for knowledge reasoning. The class, object properties, and data properties are shown below (Figure 6). For example, in resource description framework (RDF) data, the knowledge point "electromagnetic waves" belongs to the class of "remote sensing". When inquiring, the knowledge point "electromagnetic wave" can also be found in the class "remote sensing". Finally, class properties (data

properties) are created and are similar to object properties. Concurrently, Protégé also has a visual display function to show the structure of the ontology. The ontology structure is shown below (Figure 7).
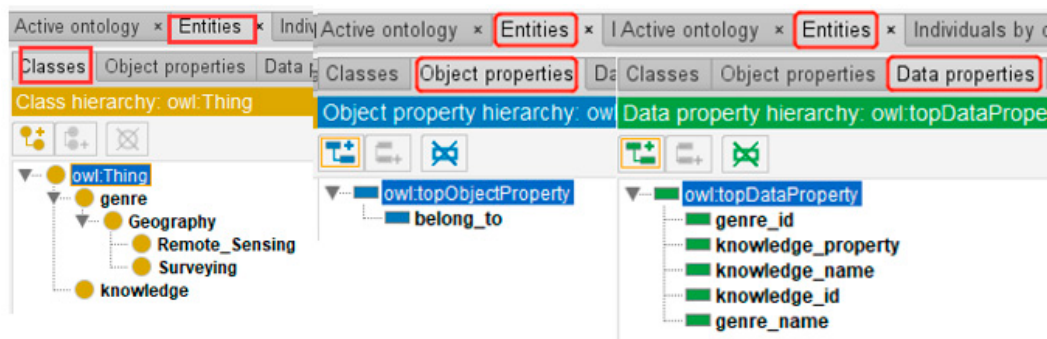

Figure 6. Construction of ontology class (**left**), object properties (**middle**), and data properties (**right**).
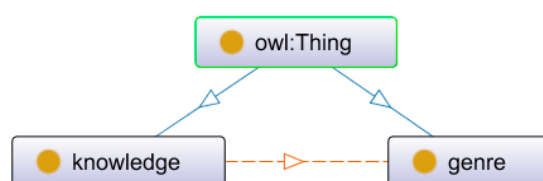

Figure 7. Visual display of ontology structure.

3.2.2. Ontology Storage

The ontology storage mode in this article is the Relational Database Management System (RDBMS). The principle of RDBMS is to map the ontology to one or more tables, and then divide it into several modes according to the mapping mode, such as horizontal storage, decomposition storage, vertical storage and hybrid storage. As the storage mechanism and data management capabilities of relational databases are relatively mature, relational database storage is widely used in many storage methods.

This article uses the RDF and web ontology language (OWL) to describe ontologies [20]. RDF is used to represent any resource information and describes resources through the mode of attribute-attribute value [21]. OWL is used to express the relationship between classes, the constraints of the set cardinality, the equality relationship, the attribute type, and the characteristics of the attribute [22]. Compared with other ontology description languages, OWL has better description ability and more description vocabulary. The description vocabulary expands the reasoning and query capabilities of the ontology.

The ontology is constructed and stored using the ontology editing tool Protégé. It cannot only construct and operate the ontology but also visually display the generated ontology, including the display of the hierarchical relationship between ontology concepts, as well as the visual display of ontology conceptual entities, entity relationships, and entity attributes. When the ontology is created using the ontology description language OWL and Protégé, semiautomatic construction can be achieved. Protégé can also reason about the ontology based on the hierarchical relationship of the ontology, with the help of Jena's query mode, and can also realize the editing operation of the ontology in the RDF and OWL languages [23].

3.3. Ontology and Database Mapping

Two standards to convert the structured data of the relational database into RDF format data have been developed by the RDB2RDF studio of W3C. The process is applied by the D2RQ tool [24].

The first standard is direct mapping, which is defined by the rule that tables in the database are classes in the associated ontology. For example, there are 3 tables for the data stored in MySQL. After mapping, the ontology has 2 classes instead of the 3 classes previously defined. The columns of tables are attributes, and the rows are instances. The content in each cell of tables is text. If a certain column in the cell is a foreign key, then it will not be able to map the data in the database to the defined ontology. In this case, the shortcomings of direct mapping are marked. In response to this defect, the RDB2RDF studio proposed R2RML and allows users to flexibly edit and set mapping rules. This mapping also provides the ability to view existing relational data in the RDF data mode, which is represented by mapping the structure selected by the customer and the target vocabulary. R2RML mapping is an RDF graph and is recorded in Turtle syntax. R2RML supports different types of mapping implementations. The processor can provide virtual SPARQL protocol and resource description framework query language (SPARQL) endpoints, generate RDF dumps, or provide link data interfaces on the mapped relational data [25]. The specific mapping diagram is shown below (Figure 8).
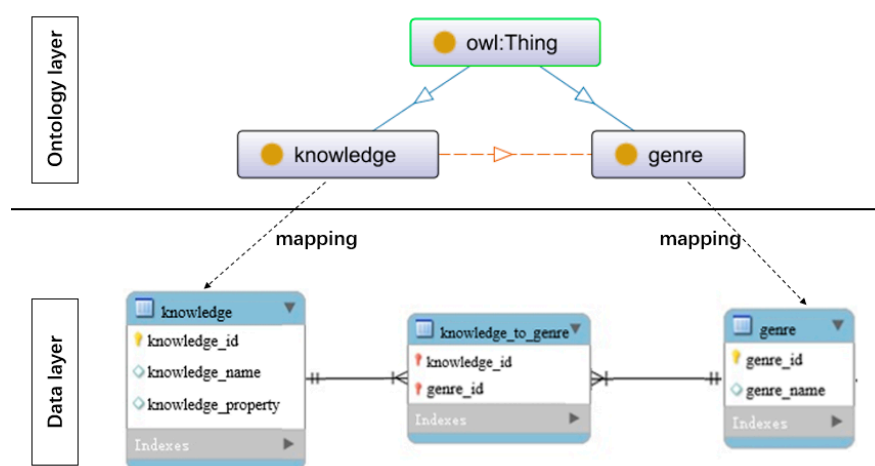

Figure 8. Schematic diagram of ontology to database mapping.

3.4. Query and Reasoning

This article describes how to use D2RQ to setup a SPARQL service endpoint and use query operations in the browser. A SPARQL endpoint provides a service that is compliant with the SPARQL protocol. Through the default or defined mapping file, RDF data can be queried in the rules of RDF. In other words, to complete the final query, D2RQ converts the SPARQL query into an SQL statement based on the mapping file and then returns the result to the user [26].

The following steps are executed: (1) start the D2R server; (2) enter the browser to start the SPARQL endpoint; and (3) enter the SPARQL statement to execute the query. Concurrently, operations can also be queried by writing Python scripts. The third-party library SPARQL Wrapper of Python can easily interact with endpoints.

Using D2RQ to open the endpoint service has two disadvantages: it does not support publishing RDF data directly to the network through the endpoint and does not support the inference. To solve these problems, certain components of Fuseki, Jena, and the tuple database (TDB) in Apache Jena are investigated experimentally. Fuseki is a SPARQL server provided by Apache Jena, which is primarily run as a web application or as an embedded server. Jena provides resource description framework schema (RDFS). In the case of a single machine, storage layer technology can provide high RDF storage performance [27]. Next, we must define reasoning rules and conduct knowledge reasoning.

3.5. Visualizing the Knowledge Graph on Neo4j

Based on the literature [2], the knowledge graph has basic query and reasoning functions; however, the visualization function of the knowledge graph has not been reflected. Therefore, the graph database Neo4j is used to visualize the knowledge graph [28]. Neo4j provides good visualizations of knowledge graphs. The specific implementation process and visualization are shown below. The process of generating the knowledge graph is as follows.

First, table data are converted into the format of CSV. These files are then moved to the import directory of Neo4j because Neo4j defaults to open files in the import directory. After starting the database, the Cypher language [29] is used to import data, and the construction mode of the node is created (k: knowledge {name: "absolute orientation"}). This statement creates a node with a knowledge label, and this node has a name attribute, an attribute value of "absolute orientation", and a variable name k. The Cypher sentence for importing structured data entity data into the database is Code 1. The result of Code 1 is shown in Figure 9.

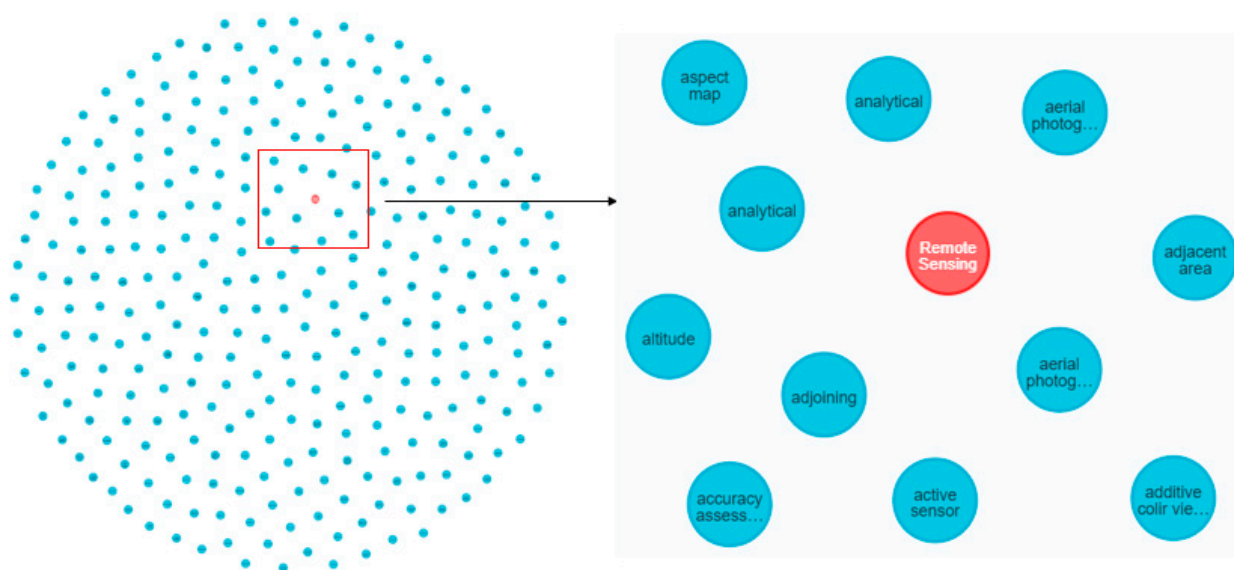| **Code 1 Entity Data Imported Based on Cypher Language** |
| --- |
| LOAD CSV WITH HEADERS FROM "file:///knowledge.csv" AS line MERGE (k:knowledge{id:line.knowledge_id, name:line.knowledge_name, genre:line.knowledge_genre}) |

Figure 9. Part of the visualization shows the result of importing entities into the database.

The name of the imported file is "knowledge.csv", and the label is "knowledge". Attributes and attribute values correspond to row and column data in the table. The imports of nodes and relationships are similar. The Cypher sentence for importing structured relational data into the database is Code 2. The result of Code 1 is shown in Figure 10.

Code 2 Relational Data Imported Based on Cypher Language
LOAD CSV WITH HEADERS FROM "file:///knowledge_to_genre.csv" AS line MATCH (from:knowledge{id:line.knowledge_id}),(to:genre{id:line.genre_id}) MERGE (from)-[r:belong_to{Relation:line.knowledge_to_genre}]->(to)

The name of the imported file is "knowledge_to_genre". The partial result of the imported visualization is shown below (Figure 11). In this picture, the two red nodes, "Surveying" and "Remote Sensing", represent the class of knowledge, and the blue node represents knowledge.
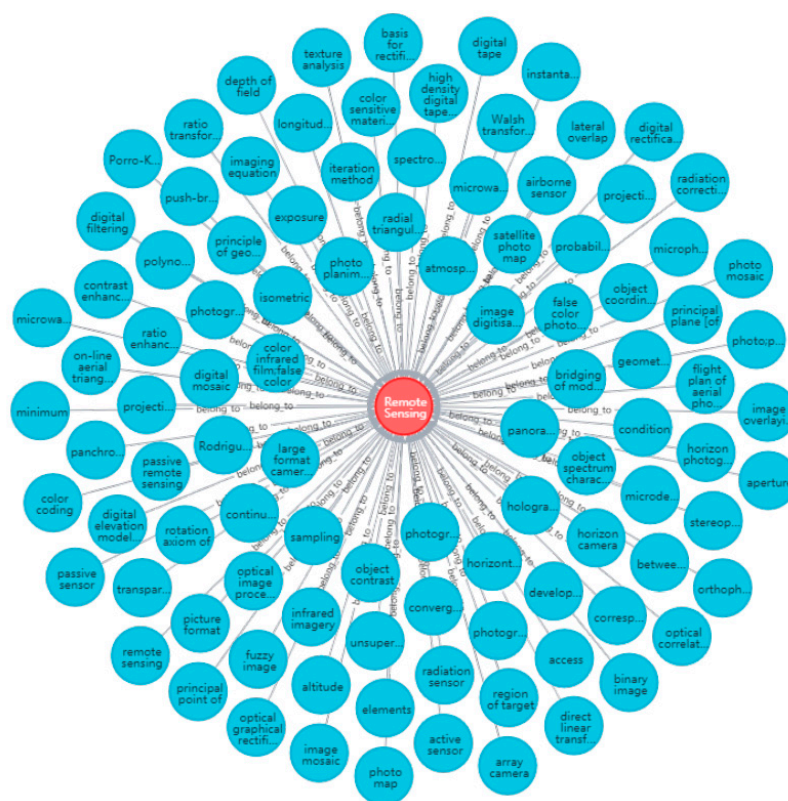
Figure 10. Part of the visualization shows the result of importing relationships into the database.
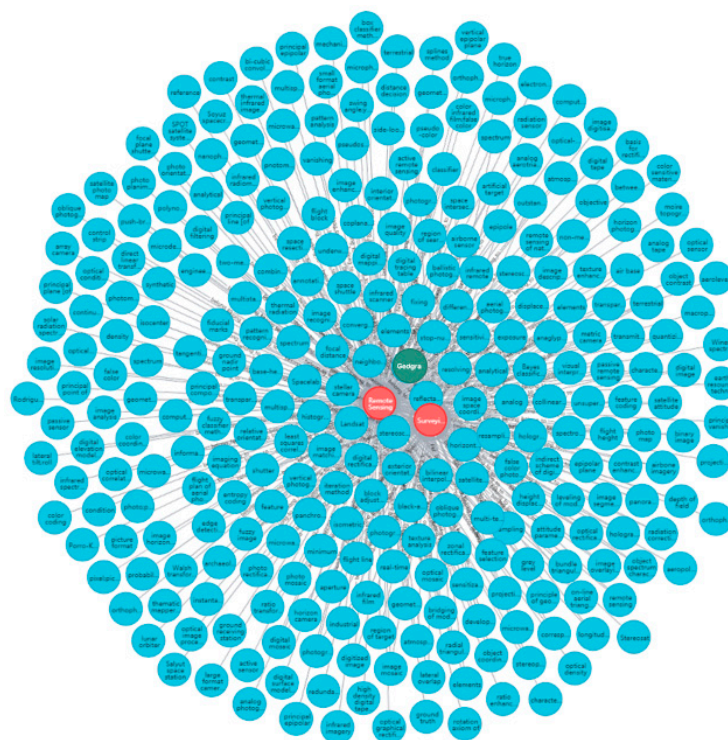

Figure 11. Part of the visualization of the knowledge graph in the field of surveying and remote sensing.

The database contains 1024 nodes, which belong to the two classes of nodes "knowledge" and "genre". The nodes in the knowledge graph are the primary components of the knowledge graph, primarily the knowledge of surveying and remote sensing. The

library also contains 1295 relationships, which represent the relationship between nodes in the class "knowledge" and nodes in the class "genre". For example, "belong_to" is the belonging relationship between knowledge and type. The library also contains attribute information, such as the "genre" and "name" of the "knowledge" node. Certain basic information for creating the database is shown below (Figure 12).
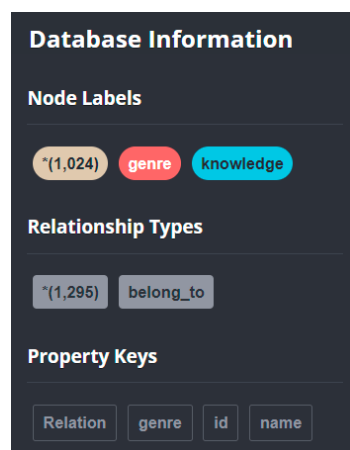

Figure 12. Part of the information of the knowledge graph knowledge base in the field of surveying and remote sensing.

4. Application Analysis of the Knowledge Graph

Application analysis of knowledge graph is an important means to verify the value of knowledge graph. This article verifies the application value of the knowledge graph from the two application scenarios of "domain relevance analysis" and "knowledge reasoning in the field of surveying and remote sensing" in the smart campus platform.

4.1. Domain Relevance Analysis

In many fields, there are various degrees of correlation. For example, in education, there is a common phenomenon in which pieces of knowledge are related between different disciplines of the same major. Zhou and others proposed a method for constructing a scientific knowledge graph based on the degree of interdisciplinary association, which aims to help students quickly understand the relationship between disciplines [30]. The relevance analysis based on the knowledge graph can assist students in choosing courses and improve teaching quality.

In recent years, with the development of science and technology, the "smart campus platform" has risen rapidly. Therefore, in the field of surveying and remote sensing, the knowledge graph can be leveraged in the context of the smart campus platform. The same knowledge is shown to exist in multiple domains, thereby highlighting the association between fields (Figure 13). The association between domains is related to the number of common knowledge between domains. The more the number of common knowledge, the higher the association between fields. In this research, if one knowledge belongs to both "surveying" and "remote sensing", then the knowledge is the common knowledge between the two fields. The more common knowledge between two fields, the higher degree of association between fields.
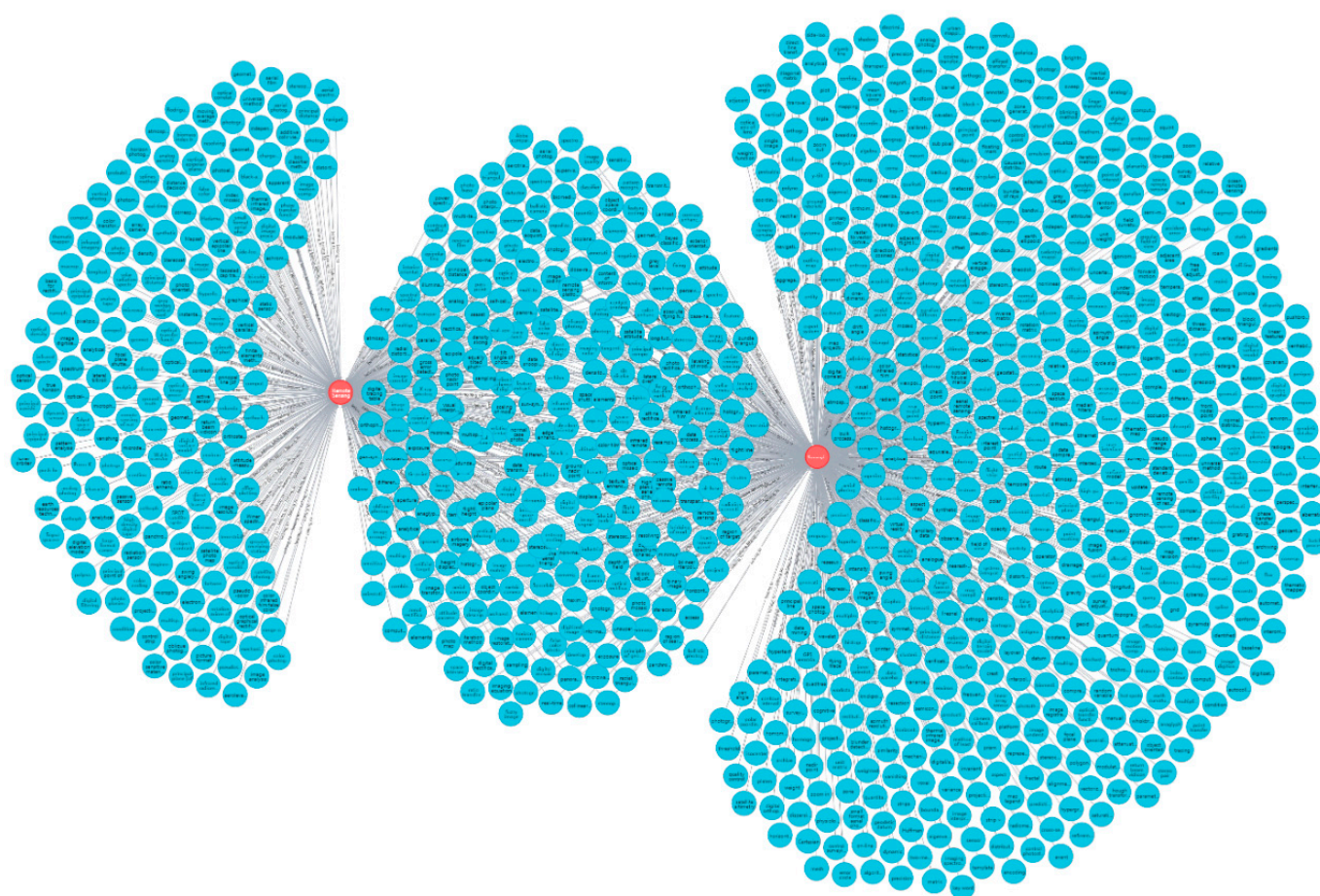
Figure 13. Display of relevance between courses.

The domain is the class in ontology. In this research, the domains of surveying and remote sensing respectively correspond to the classes of "Surveying" and "Remote Sensing" in the knowledge graph. Among all the knowledge, some knowledge belongs to the class of "Surveying", and some knowledge belongs to the class of "Remote Sensing". Some knowledge belongs to both the class of "Surveying" and the class of "Remote Sensing". These are the common knowledge of the two fields. Then this common knowledge determines the degree of association between the two fields (Figure 13).

The picture (Figure 13) contains nodes of "Surveying" and "Remote Sensing", as well as all the knowledge belonging to the field of remote sensing and surveying, which also contains the relationship "belong_to" between classes nodes and knowledge. The knowledge graph contains 1022 knowledge points. There are 273 knowledge points shared by the two disciplines. Its repetition rate is 26.7%. According to the degree of relevance, it can be used as a reference when selecting courses. If you were a student in a field unrelated to surveying and remote sensing and sought to have a general and comprehensive understanding of the field, then in order to save time and cost, you could just choose the two subjects of photogrammetry and remote sensing principles and applications. Accordingly, professional students in this field can quickly help such students understand the degree of coverage of knowledge content between subjects, help them quickly understand a nearby subject that has already studied a subject, and improve learning efficiency.

4.2. Knowledge Reasoning in the Field of Surveying and Remote Sensing

Knowledge reasoning is primarily based on known knowledge to infer new knowledge or distinguish incorrect knowledge. The reasoning function of knowledge is a prominent feature of the knowledge graph. Compared to traditional knowledge reasoning,

knowledge reasoning based on knowledge graphs is more flexible, and its methods are more diverse. There are many methods; however, the method used in this article is primarily based on the reasoning of description logic and thus primarily introduces knowledge reasoning based on this method.

The description logic system is primarily divided into four parts: (1) three basic elements (concepts, relationships and entities); (2) the axiom set, to which the concept belongs; (3) the assertion set of the entity; and (4) the reasoning mechanism. Reasoning tools based on supporting OWL DL (description logic) language include FaCT++, Racer, and pellet [31]. In addition to existing tools, knowledge can be reasoned by writing rules.

In this knowledge graph, the reasoning of knowledge is described by writing rules based on the reasoning method of description logic. This article primarily involves two types of reasoning: entity and relational. Entity reasoning infers unknown entities based on existing entities and relationships (e.g., known entity-relation-unknown entity). Relational reasoning infers the relationship between one or more entities by editing rules under the condition of existing entities (e.g., known entity-unknown relationship-known entity).

Generally, missing knowledge is inherent to knowledge graphs, that is, the incompleteness of entities or relationships. We can apply the knowledge graph in the field of surveying and remote sensing to complement the knowledge graph in the field of surveying and remote sensing. Knowledge graph completion is an important way to acquire knowledge. The goal of knowledge graph completion is to find these missing items of knowledge and add them to the knowledge graph so that the knowledge graph tends to be complete [32]. A knowledge graph is named G, and its basic components include the entity set $E = e_1, e_2, \ldots e_i$ (i is the number of entities), relation set $R = r_1, r_2 \ldots r_j$ (j is the number of relations) and corresponding triple set $T = \{(e_m, r_k, e_n)\}$ ($e_m, e_n \in E$, $r_k \in R$). Since the number of entities E and relationships R in the knowledge graph are limited, there may be some entities and relationships that are not in G. According to the content we want to complete, we can divide knowledge completion into three subtasks: ① Given a partial triplet $T_1 = (?, r_k, e_n)$, we predict the head entity. ② Given a partial triplet $T_2 = (e_m, r_k, ?)$, we predict the tail entity. ③ Given a partial triplet $T_3 = (e_m, ?, e_n)$, we predict the relationship between entities. According to whether the entities and relationships belong to the original knowledge graph, we can divide the knowledge graph completion into static knowledge graph completion and dynamic knowledge graph completion. The entities and relationships in the completion of the static knowledge graph are all in the original knowledge graph. The entities and relationships in the completion of the dynamic knowledge graph are not in the original knowledge graph. Through the completion of the knowledge graph, the collection of entities and relationships of the original knowledge graph can be expanded.

Knowledge graph completion is the most widely used field of knowledge reasoning. The original intention of a large number of knowledge graph reasoning algorithms is to be applied to knowledge graph completion, such as the Markov logic network (MLN), translating relation embeddings (TransR), capsule network-based embedding (CapsE), and relational graph neural network with hierarchical attention (RGHAT). All the methods mentioned above can determine whether there is a certain relationship between any entities by reasoning in the vector space, and then realize the completion of the knowledge graph.

Integrating the knowledge graph in the field of surveying and remote sensing into the smart campus platform can help students discover new entities or relationships between entities. For example, relational reasoning is based on the Jena tool, as shown in code 3 (based on the SPARQL language). This code defines a rule named "rule", which means that if there is an entity that belongs to "Remote Sensing", then this entity belongs to "Geography", which is (Remote Sensing-belong_to-Geography). For example: knowing (Aberration-belong_to-Remote Sensing), according to this code you can obtain the result: (Aberration-belong_to-Geography). The visual display of the whole reasoning process is shown in Figure 14.

Code 3 Inference Rules Based on SPARQL Language

@prefix: <http://www.kbdemo.com#>.
@prefix owl: <http://www.w3.org/2002/07/owl#>.
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>.
@prefix xsd: <XML Schema>.
@prefix rdfs: <http://www.w3.org/2000/02/rdf-schema#>.
[rule:(?k:belong_to ?g)(?g:hasname ?n)(?n:genre_name 'Remote Sensing')->(?k rdf:type:Geography)]
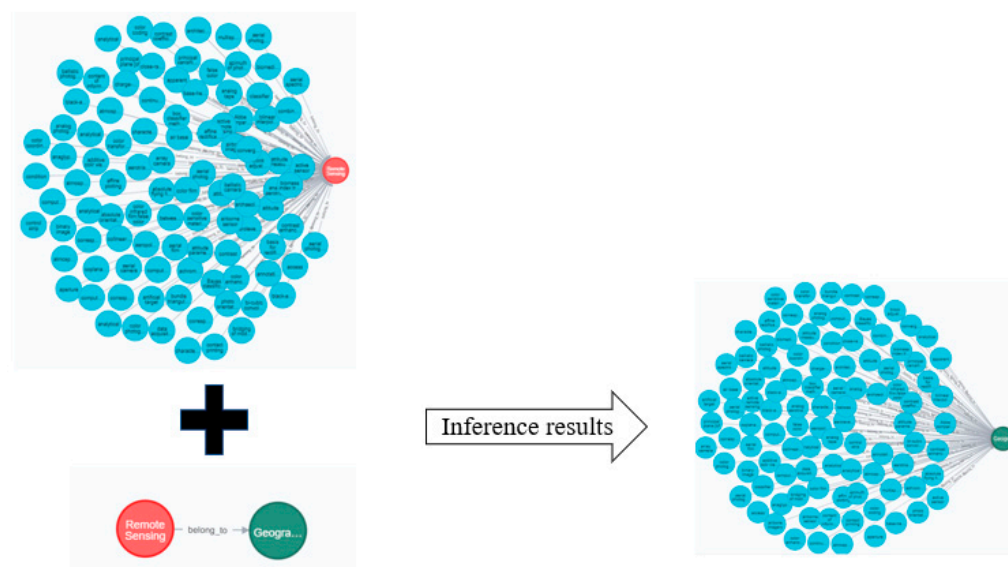[ruleInverse:(?k:belong_to ?g)->(?g:hasKnowledge ?k)]

Figure 14. Visual display of the inference process.

5. Summary and Prospect

In the field of surveying and remote sensing, rapid acquisition, efficient processing and effective application of remote-sensing data are the core tasks. However, in the face of the massive amount of data accumulated, as well as the heterogeneous, decentralized, and dynamic update characteristics of the massive amount of data, it is difficult to realize the semantic integration, interoperability and sharing platform construction of massive data application services. Since the knowledge graph is a kind of semantic network, hierarchical interconnection and semantic processing capabilities can be realized in the form of a graphical structure, and the system and relevance of knowledge can be displayed more intuitively. The knowledge graph provides a better organization and management method for isolated information and knowledge. It describes the real-world entities, concepts, and the relationships between entities and concepts in a structured and semantic form and organizes information into a form that is easier for people to understand. The purpose is to express the objective world as a well-structured knowledge expression. It is hailed as a booster for the next generation of artificial intelligence. The contributions of this article are as follows:

(1)   To quickly obtain effective data from massive amounts of heterogeneous, decentralized, and dynamically updated data. This article proposes a method for constructing a subject knowledge graph in the field of surveying and remote sensing.

(2)   To verify the application value of the knowledge graph in the field of surveying and remote sensing. This article verifies its functions from the query, visualization and reasoning of the knowledge graph.

The challenges and countermeasures encountered in the research are as follows:

(1)   The connection between the mode layer and the data layer. The knowledge graph in the field of surveying and remote sensing is mainly divided into two parts: mode

level construction and a data layer. The mode level is the foundation of the data layer. Through the factual expression of the ontology library standard data layer, ontology is the conceptual template of a structured knowledge base. The data layer is composed of a series of knowledge entities or concepts. Knowledge is stored in units of facts. The data layer expresses knowledge in the form of triples (entity 1-relation-entity 2) or (entity-attribute-attribute value). Realizing the association between ontology and data at two levels is a major challenge in constructing a knowledge graph. This article uses the D2RQ tool to realize mapping from ontology to the database. The D2RQ tool converts the structured data of the relational database into data in RDF format. This mapping also provides the ability to view existing relational data that exist in the RDF data model, which is represented by mapping the structure selected by the customer and the target vocabulary. The R2RML mapping itself is an RDF graph and is recorded in Turtle syntax. R2RML supports different types of mapping implementation. The processor can provide virtual SPARQL endpoints on the mapped relational data, or generate RDF dumps, provide a link data interface.

(2) Application and practice of the domain knowledge graph. This article gives an example of the application of integrating the knowledge graph in the field of surveying and remote sensing into the smart campus platform. The knowledge visualization application of the domain knowledge graph on the smart campus platform can assist teachers and students in selecting courses. The domain knowledge graph is applied to knowledge reasoning on the smart campus platform, which can help teachers and students discover and reason about new knowledge, as well as new relationships between knowledge.

Knowledge in the field of surveying and remote sensing is diverse and complex, and knowledge graphs can be studied in more detail in the following areas. Data in these fields of study are typically images, and the recognition and acquisition of knowledge in image data are understudied; in particular, the knowledge graph of image data lacks a time dimension. Thus, future research should investigate how to add the time dimension to the knowledge graph and how to extend the time dimension to the application of the knowledge graph. If the above problems can be broken through, then the knowledge graph in the field of surveying and remote sensing will have greater potential application value:

(1) The discovery of new rules of surveying and remote sensing: The continuous increase in surveying and remote-sensing data and the continuous improvement of digital management and utilization technology have provided great convenience for scientific researchers to carry out research work. The surveying and remote-sensing knowledge graph provides support for the insight and discovery of regular knowledge of surveying and remote-sensing resources by associating a large amount of surveying and remote sensing knowledge into a network structure. Researchers can discover various knowledge and rules hidden behind the development process through the analysis of surveying and remote-sensing data to provide relevant scientific research personnel and scientific research policy makers with scientific research directions and a policy-making basis.

(2) Application of machine learning methods in the analysis of knowledge graphs in the field of surveying and remote sensing: From the development process of the combination of machine learning and knowledge graphs (the knowledge graph as a complex network; traditional machine learning methods to conduct graph mining and analysis on the knowledge graph; further application of deep learning methods and graph neural network methods in knowledge graphs), the value and function of the knowledge graphs have been further embodied. The surveying and remote-sensing domain knowledge graph is a special domain knowledge graph, and the analysis method in the general knowledge graph is used to mine and analyse the graph, but it cannot make full use of the structure and characteristics of the surveying and remote-sensing domain knowledge graph. For the specific structural features and entity attributes in the knowledge graph of surveying and remote sensing, it

is necessary to design specific machine learning methods or deep neural network structures. At the same time, for different application scenarios, different objective functions are usually designed to learn the parameters of the algorithm. Therefore, the study of machine learning and deep learning mining methods for the knowledge graph in the field of surveying and remote sensing is helpful to the further analysis and application of the knowledge graph in the field of surveying and remote sensing.

(3) Construction of the service platform of the knowledge graph in the field of surveying and remote sensing: In the construction of the knowledge graph in the field of surveying and remote sensing, the work efficiency is affected due to the problem of scattered tools. The next research plan is to build a knowledge graph service platform in the field of surveying and remote sensing to realize the integration of tools and services. At the data source level, it integrates all kinds of open and available data in the field of surveying and remote sensing, as well as data unique to each demand side. Through the provided functions of surveying and remote sensing data acquisition, data storage, ontology construction, graph construction and update, a knowledge graph of the field of surveying and remote sensing that can be updated in time can be constructed. In terms of services, through the provision of knowledge service algorithms and models such as knowledge queries, knowledge visualization, and knowledge reasoning, a service platform provides targeted knowledge services for different roles.

Author Contributions: Conceptualization, X.H. and Z.J.; methodology, X.H.; validation, X.L., Q.L. and R.Y.; formal analysis, Z.J.; investigation, L.L.; resources, X.H.; data curation, X.L.; writing—original draft preparation, X.H.; writing—review and editing, L.L., L.Y.; visualization, M.S.; supervision, X.L.; project administration, Z.J.; funding acquisition, X.L. All authors have read and agreed to the published version of the manuscript.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The storage URL of the structured raw data to construct the knowledge graph is: https://github.com/hao1661282457/Knowledge-graphs.git (accessed on 25 June 2021).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Li, W.; Xiao, Y.W.; Wang, W. People Entity Recognition Based on Chinese Knowledge Graph. *Comput. Eng.* **2017**, *43*, 225–231,240. [CrossRef]
2. Liu, Q.; Li, Y.; Duan, H.; Liu, Y.; Qin, Z. Knowledge Graph Construction Techniques. *J. Comput. Res. Dev.* **2016**, *53*, 582–600. [CrossRef]
3. Cao, Q.; Zhao, Y. The Technical Realization Process and Related Applications of Knowledge Graph. *Inf. Stud. Theory Appl.* **2015**, *12*, 127–132. [CrossRef]
4. Xu, Z.; Sheng, Y.; He, L.; Wang, Y. Review on Knowledge Graph Techniques. *J. Univ. Electron. Sci. Technol. China* **2016**, *45*, 589–606. [CrossRef]
5. Wang, Z.; Xiong, C.; Zhang, L.; Xia, G. Accurate Annotation of Remote Sensing Images via Active Spectral Clustering with Little Expert Knowledge. *Remote Sens.* **2015**, *7*, 15014–15045. [CrossRef]
6. Xie, R.; Luo, Z.; Wang, Y.; Chen, W. Key Techniques for Establishing Domain Specific Large Scale Knowledge Graph of Remote Sensing Satellite. *Radio Eng.* **2017**, *47*, 1–6. [CrossRef]
7. Jiang, B.; Wan, G.; Xu, J. Geographic Knowledge Graph Building Extracted from Multi-sourced Heterogeneous Data. *Acta Geod. et Cartogr. Sin.* **2018**, *47*, 1051–1061. [CrossRef]
8. Lu, F.; Yu, L.; Qiu, P. On Geographic Knowledge graph. *J. Geo Inf. Sci.* **2017**, *19*, 723–734. [CrossRef]

9.	Zhu, L.J. Research on Information Resource Management Model Based on Domain Knowledge in World Wide Web Environment. Ph.D. Thesis, China Agricultural University, Beijing, China, 1 June 2004.

10.	Wang, L.; Wang, J.; Xu, N.; Deng, Y. Knowledge Graph-based Metro Engineering Accidents Knowledge Modeling and Analysis. *J. Civil. Eng. Manag.* **2019**, *36*, 109–114,122. [CrossRef]

11.	Wei, T.; Wang, J. Construction of Knowledge Graph based on Non-classification Relation Extraction Technology. *Ind. Technol. Innov.* **2020**, *37*, 27–32. [CrossRef]

12.	He, L. Research on Key Techniques of Entity Attribute Extraction for Unstructured Text. Master's Thesis, Harbin University of Science and Technology, Harbin, China, 1 June 2020.

13.	Sun, J.B. *Principles and Applications of Remote Sensing*, 3rd ed.; Whuhan University Press: Whuhan, China, 2009; pp. 23–126.

14.	Kong, X.Y.; Guo, J.; Liu, Z. *Founding of Geodesy*, 4th ed.; Whuhan University Press: Wuhan, China, 2005; pp. 56–89.

15.	Vyas, A.; Kadakia, U.; Jat, P. Extraction of Professional Details from Web-URLs using DeepDive. *Procedia Comput. Sci.* **2018**, *132*, 1602–1610. [CrossRef]

16.	Ma, H.B. Research on Construction and Application of Knowledge Graph of Enterprise Related Information for Risk Control. Master's Thesis, Beijing University of Technology, Beijing, China, 30 May 2019; pp. 25–27.

17.	Abad, A.; Moschitti, A. Distant supervision for relation extraction using tree kernels. *Appl. Clay Sci.* **2015**, *115*, 108–114. [CrossRef]

18.	Mallory, E.; Zhang, C.; Christopher, R.; Altman, R. Large-scale extraction of gene interactions from full-text literature using DeepDive. *Bioinformatics* **2016**, *1*, 106–113. [CrossRef]

19.	John, H.; Gennari, J.; Musen, M.; Fergerson, R. The Evolution of Protégé: An Environment for Knowledge-Based Systems Development. *Int. J. Hum. Comput. Stud.* **2003**, *58*, 89–123. [CrossRef]

20.	Zhang, R. Research and Analysis Based on Semantics of Rice Domain Knowledge Expression. Master's Thesis, Hunan Agricultural University, Hunan, China, 8 May 2007.

21.	Arenas, M.; Ugarte, M. Designing a Query Language for RDF. *ACM Trans. Database Syst.* **2017**, *42*, 21.1–21.46. [CrossRef]

22.	Gan, J.; Xia, Y.; Xu, T.; Zhang, X. Extension of Web Ontology Language OWL in Knowledge Representation. *J. Yunnan Norm. Univ.* **2005**, *25*, 9–14. [CrossRef]

23.	Duan, X.; Wang, L.; Wang, S. A Preliminary Study on the Application of Knowledge Graphs in Professional Fields. *Electron. World* **2020**, 4. [CrossRef]

24.	Liu, J. Research on the Construction and Application of Knowledge Graph in Tourism Domain. Master's Thesis, Zhejiang University, Hangzhou, China, 1 June 2019.

25.	Ye, S. Research on the Construction and Query Method of Knowledge Graph in Coalmine Based on Neo4j. Master's Thesis, China University of Mining and Technology, Xuzhou, China, 30 May 2019.

26.	Zhou, W. The Construction and Application of Knowledge Graph Incorporating Causal Events. Master's Thesis, East China Normal University, Shanghai, China, 23 May 2019.

27.	Yang, X.; Yang, M.; Yang, D.; Huang, Y. Research on Knowledge Fusion Triplets Storage Structure Based on Jena System. *Value Eng.* **2018**, *8*, 134–137. [CrossRef]

28.	Zhao, K.; Wang, H.; Shi, N.; Sa, Z.; Xu, X. Study and Implementation on Knowledge Graph of Guizhi Decoction Associated Formulas Based on Neo4j. *World Chin. Med.* **2019**, *14*, 2636–2646. [CrossRef]

29.	Zhang, Z. Research on the Parsing of Graph Database Query Language Cypher. Master's Thesis, Huazhong University of Science and Technology, Wuhan, China, 8 May 2018.

30.	Zhou, Z.; Xue, D.; Xin, X.; Yang, L. A Construction Method of Scientific Knowledge Graph Based on the Degree of Inter-disciplinary Association. In Proceedings of the 2011 Fall Academic Conference of Chinese Physical Society, Hangzhou, China, 15 September 2011.

31.	Chen, B.; Li, G.; Zhang, J.; Li, J. Framework design of SWRL-based Reasoning Mechanism. *Comput. Eng. Des.* **2010**, *31*, 847–849,853. [CrossRef]

32.	Wang, W.G. Knowledge Graph Reasoning: Modern Methods and Applications. *Big Data Res.* **2021**, *1*, 1–24. Available online: https://kns.cnki.net/kcms/detail/10.1321.G2.20210331.1811.0O4.html (accessed on 9 May 2021).