A New Knowledge Reasoning Scheme for Soil Erosion Based on Knowledge Graph

Fan Lei[1], Yabo Liu[2,*], Zhichao Zhang[2], Yifan Hao[2], and Hao He[2]

[1] Key Laboratory of Natural Resource Monitoring and Supervision in the Southern Hilly Region, Ministry of Natural Resources, China.
[2] School of Electronic and Information Engineering, Hebei University of Technology, Tianjin, China.

* Corresponding author. Tel.: 13588827927; email: lyb196368888@foxmail.com.
Manuscript submitted July 15, 2025; revised August 22, 2025; accepted September 3, 2025; published October 23, 2025
doi: 10.17706/jsw.20.2.66-83

Abstract: Different types of natural resource data are stored in different databases, which results in them having a low degree of correlation. Combining the data of different natural resources to build a knowledge graph for knowledge reasoning would likely increase the correlation of these data, enabling information mining and assisting in the management of natural resources. We construct a natural resource knowledge graph that interrelates different databases and develop a new knowledge reasoning scheme using remote sensing, vegetation distribution, and soil erosion spatial distribution data from Hunan Province, China. The proposed scheme includes an input layer, unified generation layer, feature extraction layer, knowledge reasoning layer and output layer for analyzing the influence of altitude, soil type and vegetation type on the spatial distribution of soil erosion. The experimental results show that the proposed knowledge reasoning scheme performs well according to the MRR and Hits@N evaluation metrics. Our research provides a scientific basis for predicting the spatial distribution of soil erosion and preventing soil erosion in practical scenarios.

Keywords: soil erosion prediction, knowledge graph, knowledge reasoning, natural resources

1. Introduction

Natural resource data are rich and are often stored in different databases in various formats. The data correlation between different resource attribute databases in a specific natural resource region is relatively low, and it is impossible to fully utilize the various databases for natural resource management. A knowledge graph adopts an internationally standardized coding method, which is scientific, systematic, and normative, to formally express the correlation between different types of knowledge [1]. The use of knowledge graphs to manage natural resource data has become a hot topic in the field of natural resource information science [2]. A knowledge graph of natural resources can be constructed to design a structured knowledge base of natural resource data. Describing the concepts, attributes, data, and interrelationships of relevant natural resources, entities are interconnected, and various types of natural resource data are fused to form a network knowledge structure [3]. Based on the constructed natural resource knowledge graph, knowledge inference can be used to supplement existing knowledge graphs and infer new knowledge to better utilize natural resource data and manage natural resources.

A knowledge graph is a networked knowledge base consisting of entities with attributes linked through

relationships [4, 5]. Two main methods are used for constructing knowledge graphs: (1) ontology construction of open domain knowledge graphs, in which a bottom-up approach is typically used to automatically extract concepts, hierarchies, and the relationships between concepts from the knowledge graph, and (2) ontology construction of domain knowledge graphs, in which a top-down approach is often adopted. Current representative knowledge graphs include Freebase [6], WordNet [7], YAGO [8–10], DBpedia [11], NEL [12, 13], KnowltAll [14], Probase [15], etc. In recent years, scholars have utilized geospatial data to construct geographic knowledge graphs. For example, OS Monte extracts attribute labels from Open Street Map (OSM) files and unifies data concept information [16, 17]. LinkedGeoData builds a triplet data model through OSM data sources and establishes links with data sources such as DBpedia at the entity level [18]. The OSM semantic network obtains information from OSM data and establishes a machine-recognizable natural resource spatial semantic network [19]. Yago2 and Clinga use Wikipedia and Baidu Baike to obtain natural resource data such as roads, points of interest, buildings, and stations [20, 21]. Research on natural resource knowledge graphs encompasses various domains, such as earthquake disaster scenario management, knowledge services in the field of oil reservoir structure, construction of a mineral knowledge base, and intelligent Q&A for geological knowledge. However, many problems still need to be solved for the construction of large-scale natural resource knowledge graphs and big data analysis applications [22]. With respect to data quality issues, a knowledge graph may contain errors, incomplete data, or data inconsistencies, which affects reasoning accuracy and reliability. Challenging knowledge representation problems include the effective representation of entities, relationships, and attributes in a knowledge graph and how to combine them with inference algorithms. Regarding complexity and scalability, as the scale of a knowledge graph increases, the complexity and computation of inference problems also increase, making it challenging to design efficient inference algorithms. If a knowledge graph inference model lacks generalizability to different fields or tasks, transfer learning and other technologies to improve the generalizability of the inference model are important considerations.

Knowledge reasoning based on knowledge graphs is an important research direction in which the entity, relationship and attribute information in knowledge graphs is used to conduct inference to improve the constructed knowledge graph. Current research in this area encompasses the following methods. (1) Logical reasoning: This method typically uses rules of logical reasoning, such as predicate logic or descriptive logic, to reason in combination with entities and relations in the knowledge graph. For example, logical rules can be used to infer new facts or relationships. (2) Graph neural networks: This method uses a graph neural network for learning representations of knowledge graphs to achieve inference capabilities. By representing entities and relationships as vectors and using graph neural networks for information propagation and aggregation, inferences can be realized. (3) Graph matching: This method enables inference based on finding similar subgraphs or patterns in a knowledge graph. By finding new patterns that are similar to known patterns, new information or relationships can be inferred. (4) Rule-based and semantic similarity methods: Rule reasoning and semantic similarity calculations are combined, and inferences are realized by comparing the semantic similarity of entities and relations.

Reference [23] designed an ontology framework in a top-down manner and established the interactive relationship between chemical processes and risk analysis. In addition, a rule-based knowledge reasoning method was proposed based on expert chemistry knowledge to fill in missing relationships in the knowledge graph database. Reference [24] adds missing information to a knowledge graph by generating an inference path graph during the inference process. The inference model based on description logic proposed in references [25, 26] realizes knowledge inference of the cold chain of agricultural products.

In this paper, we construct a knowledge graph of natural resources in Hunan Province and present a knowledge reasoning scheme to infer vegetation types and soil erosion types. First, Tu ethnicity, altitude, and

soil erosion entities are extracted from different databases. Then, diverse entity information is associated with the same coordinate point, and a knowledge graph is constructed based on association relationships. Subsequently, we develop a knowledge reasoning scheme based on the HAKE model and RotatE model by dividing an entity into its modulus and phase components as well as considering the correlation relationship. The evaluation indicators for our scheme include MRR and Hits@N. The MR index represents the average ranking, so a useful result should be as small as possible. The MR using our scheme is 7.868131. The MRR index represents the reciprocal ranking of the average, so a useful result should be as large as possible. The MRR using our scheme is 0.759090. At N of 1, 3 and 10, the Hits@N are 0.706436, 0.781790 and 0.863422, respectively. Various evaluation indicators show that our knowledge reasoning scheme has good results and can be used to make accurate predictions and that it even outperforms the HAKE model and RotatE model.

Research on natural resources in the current era not only encompasses traditional ecological services such as soil and water conservation, wind prevention and sand fixation, and biodiversity maintenance but also shoulders the new mission and task of ensuring the security and comprehensive management of natural resources in light of international trade patterns, resource price fluctuations, the development trend of the COVID-19 epidemic, development foci and development goals. At present, mountain, water, forest, field, lake, grass and other natural resource data are scattered across different databases, causing problems such as low degrees of correlation and information islands. By integrating all types of natural resource data and conducting knowledge reasoning, we can increase the correlation between existing natural resource data, mine more potential information, and provide a comprehensive and effective basis for natural resource management.

2. Construction of the Knowledge Graph

2.1. Datasets

The datasets used in our scheme consist of three different datasets, namely, land use remote sensing detection data from Hunan Province [27], vegetation type distribution data [28] and soil erosion spatial distribution data in China [29]. The data originate from a geographic remote sensing ecological network platform. The first dataset comprises soil erosion type and intensity data across China and contains 9,490,138 samples. The second dataset concerns the distribution of vegetation types across the country and contains 9,501,810 samples. The third dataset concerns the distribution of soil types across Hunan Province and contains 211,795 samples. Due to the close proximity of these sample locations to each other, their characteristics, such as soil type, soil erosion, and vegetation type, are the same. After eliminating duplicates, the training set and test sets contain 1909 and 637 samples, respectively.

2.2. Architecture Definition

Many datasets with different structures are used in the construction of knowledge graphs, so it is very important to use a suitable structure for the purpose of consistency. The elevation, soil type, vegetation, and soil erosion data extracted from the above datasets are treated as entities. The relationships among them are defined based on geographic and other information in the Baidu Encyclopedia, as shown in Fig. 1. The height of elevation affects the type of vegetation growth, and the distribution of vegetation is also affected by altitude. The growth of vegetation is affected by different types of soil. Vegetation may suffer from soil erosion, and soil erosion may also determine the type of vegetation that grows. For the rigor of the overall structure, we also define two negative relationships: altitude cannot affect soil erosion, and there is no direct correlation between altitude and soil nationality.

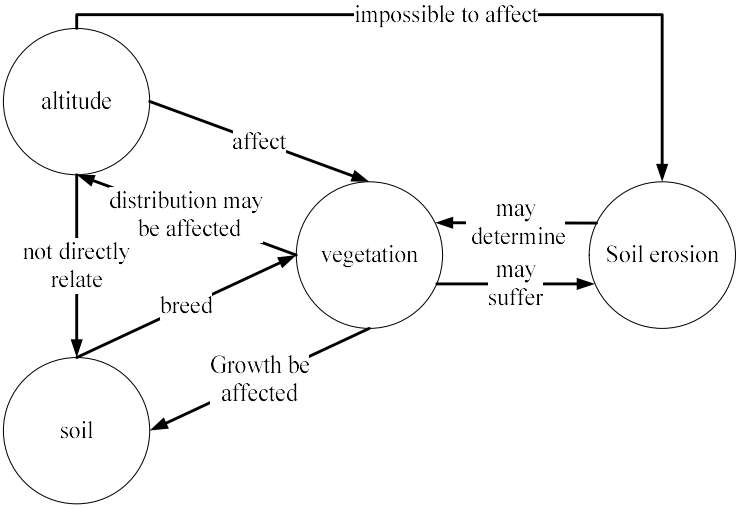

Fig. 1. Conceptual architecture.

2.3. Acquisition of Entities and Relationships

Our data originate from different datasets, so we needed to associate different datasets. The key for this step is coordinates; although the datasets are different, the geographical location is objectively the same when measuring the data. Therefore, we must obtain the coordinate information of different datasets. However, all the datasets are saved in .adf files, and there is no way to obtain coordinate information directly from them. Thus, we rasterize the layers in each file, mark the grid turn point and calculate the latitude and longitude coordinates of each point.

After obtaining the coordinate information of the soil erosion, vegetation type and land use remote sensing detection data from Hunan Province, we use it to correlate different datasets. In the proposed scheme, the Euclidean distance between each dataset is calculated according to latitude and longitude. At the smallest distance, it is assumed that two points are located at the same position. Then, we can associate the data by distance and merge different types of data. Since the extracted coordinates are in the form of degree minutes and seconds, converting degree minutes and seconds to latitude and longitude is necessary. The conversion equations are as follows:

$$L = D_L + \frac{M_L}{60} + \frac{S_L}{3600} \tag{1}$$

$$B = D_B + \frac{M_B}{60} + \frac{S_B}{3600} \tag{2}$$

In Eqs. (1) and (2), the variables D_L, M_L and S_L denote the degrees, minutes, and seconds of longitude, respectively, and the variables D_B, M_B and S_B denote the degrees, minutes, and seconds of latitude, respectively. When calculating the distance between two coordinate points, the longitude and latitude distance calculation formula is used. The equation for calculating distance is as follows:

$$d_{i,j} = 2R \arcsin\left(\sqrt{\sin^2\left(\frac{W_i - W_j}{2}\right) + \cos(W_i)\cos(W_j)\sin^2\left(\frac{J_i - J_j}{2}\right)}\right) \times \pi / 180 \tag{3}$$

In Eq. (3), R refers to the radius of the Earth, W_i is the latitude of point i, W_j is the latitude of point j, J_j is the

longitude of point i, and J_j is the longitude of point j. Importantly, the latitude and longitude are converted into radians. The information at two coordinate points in different datasets is then correlated by a distance calculation formula.

2.4. The Constructed Knowledge Graph

After processing the above data, we use Eq. (3) to associate different datasets and obtain entities and relationships. The obtained information contains many entities, attributes, and relationships. To merge these data, our scheme employs the Vue.js framework to construct a user interface and uses the AntVG6 library to display the constructed knowledge graph.

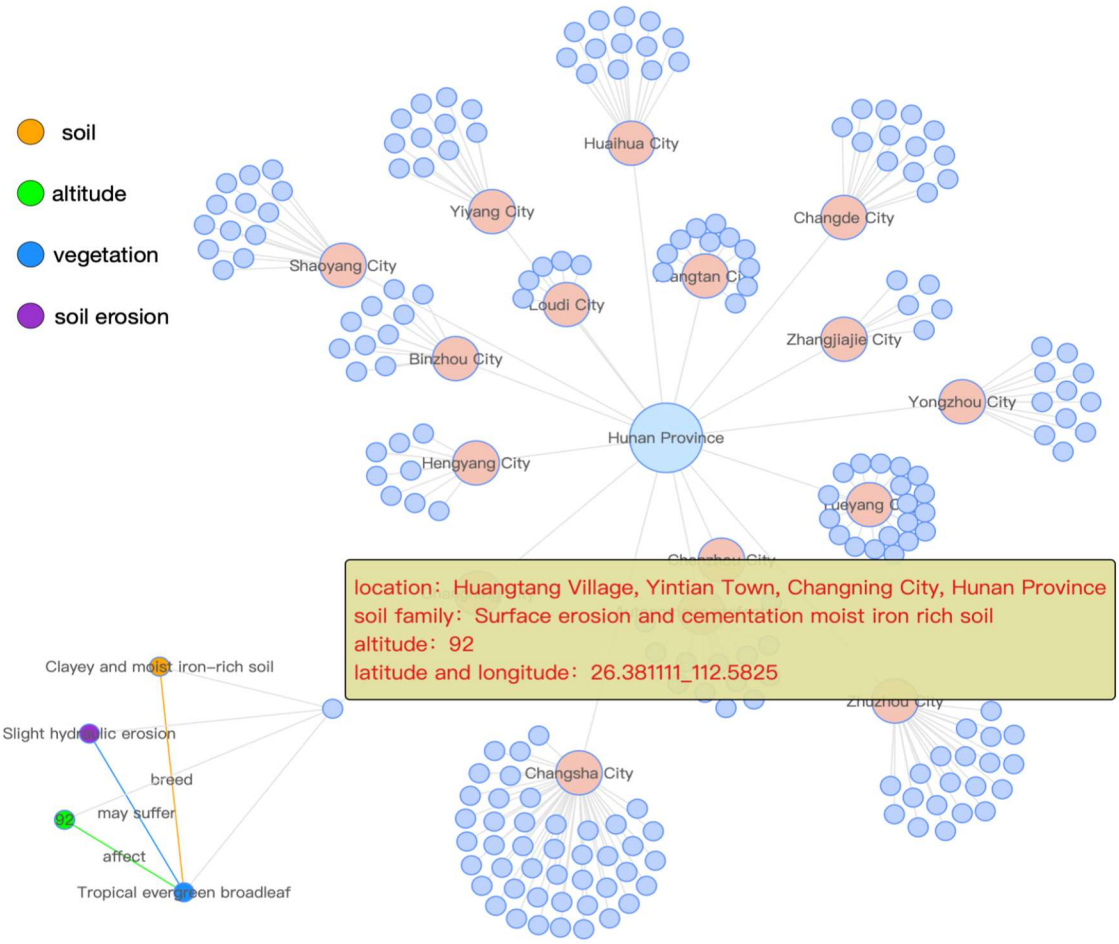

Fig. 2. Visualization of the knowledge graph.

Fig. 2 illustrates the constructed dataset presented in a knowledge graph format. The soil type, altitude, and detailed location information for each prefecture-level city in Hunan Province at different longitude and latitude coordinates is shown.

Taking the village of Huangtang in Changning as an example, clicking on a node in the graph displays the village's soil type, vegetation type, and potential for soil erosion. Fig. 2 shows that its altitude of 92m affects the growth of local tropical evergreen broadleaf trees.

In subsequent research, we can query our knowledge graph using the longitude and latitude coordinates of a specific location to determine its soil type, vegetation distribution, and altitude and use the knowledge inference model to predict soil erosion.

3. Knowledge Reasoning

3.1. Knowledge Reasoning Framework

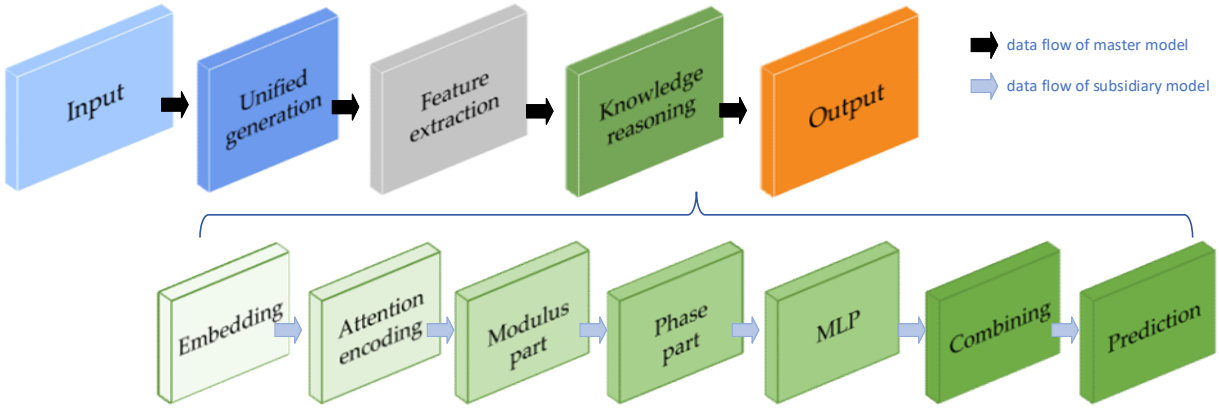

Fig. 3. Knowledge reasoning framework.

After obtaining the knowledge graph, we construct a knowledge reasoning model based on the graph and use the information collected by the graph for prediction. Fig. 3 shows the overall knowledge reasoning framework, which is divided into five layers: input, unified generation, feature extraction, knowledge reasoning and output.

Input Layer: The input data contain the latitude and longitude coordinates of the training samples. Because the required data come from different datasets, we use latitude and longitude coordinates to correlate the different datasets. The data consist of small grid elements, so grid inflection points are obtained first, and then the latitude and longitude coordinates of each inflection point are calculated.

Unified generation layer: The unified generation layer unifies the format of the training samples. The Euclidean distance between each dataset is calculated by latitude and longitude, allowing disparate data to be combined. Since the extracted coordinates are in the form of degree minutes and seconds, the degree, minutes and seconds are converted to latitude and longitude using Eqs. (1) and (2). The Euclidean distance is calculated using Eq. (3).

Feature extraction layer: The feature extraction layer performs characteristic extraction from training samples. Specifically, the spatial attributes of prefecture-level cities are determined by coordinate point positioning, and the quantitative analyses of soil types and elevation data employ minimal Euclidean distance computation for feature discrimination.

Knowledge reasoning layer: The knowledge reasoning layer processes training samples through an integrated reasoning architecture. Vegetation classifications and soil erosion taxonomies are predicted by leveraging the terrain data and soil characteristics in the training samples through our constructed knowledge graph framework. The knowledge reasoning layer comprises seven core functional modules:

(1) Embedding of the knowledge reasoning layer: For a knowledge graph embedding model to capture semantic hierarchies, it must distinguish between entities in two specific categories. Entities belonging to distinct hierarchies, such as vegetation types and soil erosion types, as well as entities at the same hierarchical level, such as entities of the same vegetation type, must be discerned by the model. The embedding of the knowledge reasoning layer facilitates the modeling of the subsequent modulus and phase components. Then we embed the entities and relationships and transform them into low-dimensional vectors.

(2) Attention encoding of the knowledge reasoning layer: After embedding relationships and entities, to better capture the complex interactions between them, our scheme contains an attention decoding block, which takes the embedded entities and relationships as inputs, as shown in the following Fig. 4. First, a normalization layer ensures the stability of the model during training. Then, multi-attention calculations are

carried out on the head entities, tail entities and relationships. To prevent overfitting, our scheme introduces a dropout layer here and then re-normalizes it.

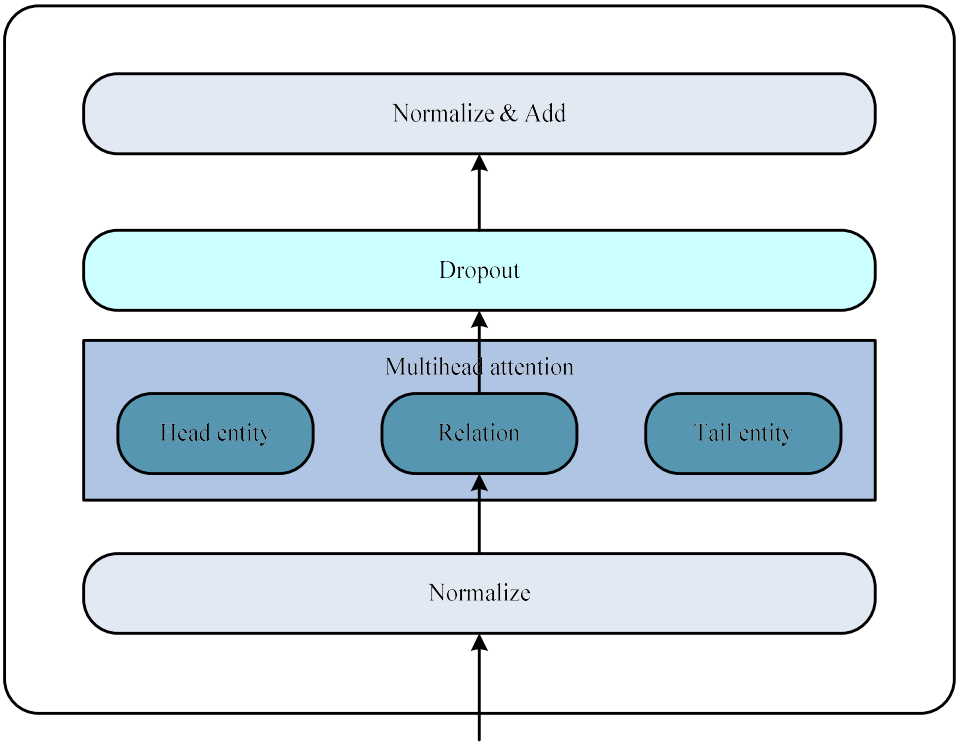

Fig. 4. Attention decoding block framework.

The multi-head attention mechanism in the block takes the embedded entities and relationships as inputs and multiplies them with three learnable weight matrices (W^q, W^k, and W^v) to obtain Q, K, and V of the head entity, tail entity and relationship. Next, the correlation between Q and K is calculated by the dot product, and then the output, Z, is obtained by multiplying V after softmax normalization. d is the embedded dimension, and Z_H, Z_T, and Z_R are the multi-head attention calculation outputs of the head entity, tail entity, and relationship, respectively. This process allows the model to pay attention to the correlations between entities and relationships.

$$Z_H(Q,K,V) = Layernormal(X(Q_H,K_H,V_H) + soft\max(\frac{Q_H K_H^T}{\sqrt{\mathrm{d}}})V_H) \tag{4}$$

$$Z_R(Q,K,V) = Layernormal(X(Q_R,K_R,V_R) + soft\max(\frac{Q_R K_R^T}{\sqrt{\mathrm{d}}})V_R) \tag{5}$$

$$Z_T(Q,K,V) = Layernormal(X(Q_T,K_T,V_T) + soft\max(\frac{Q_T K_T^T}{\sqrt{\mathrm{d}}})V_T) \tag{6}$$

Our scheme uses three heads to calculate the outputs of the header entity, tail entity and relationship. By adding multiple attention heads to allocate attention to different dimensions of the input information in parallel, we obtain three independent weight matrices that better capture the correlations between entities and relationships. The output Z from each header is merged and then multiplied by a weight matrix W^0 to obtain the final output. Fig. 5 shows a concrete example to explain the flow of the entire multi-head attention mechanism.

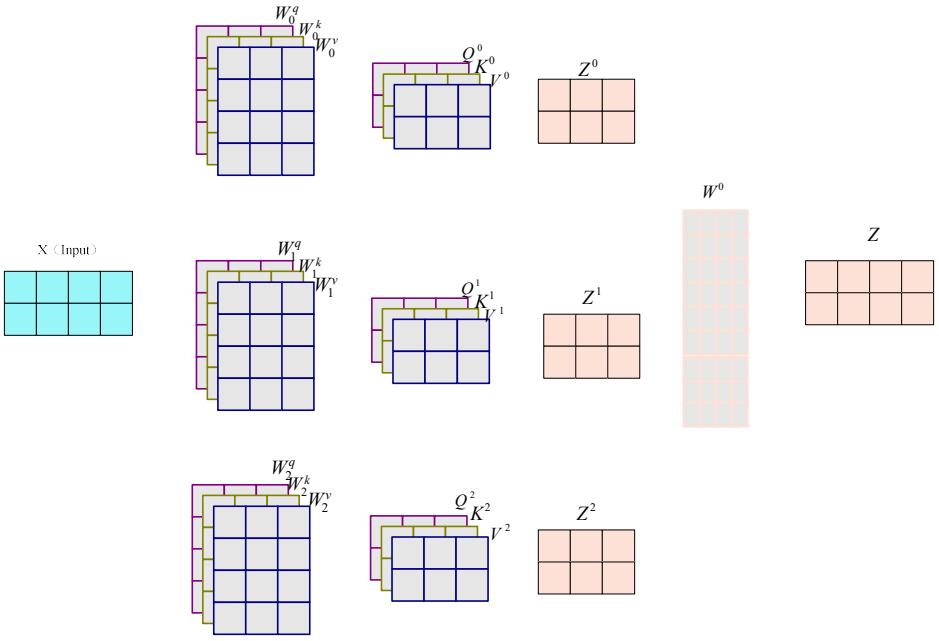

Fig. 5. Example of the flow of the multi-head attention mechanism.

(3) **Modulus component of the knowledge reasoning layer:** To model the entities at different levels of the hierarchy, our scheme models all the entities into different hierarchies. Thus, the soil type, vegetation type and soil erosion entities are placed into different hierarchical structures. Then, the moduli of their entity embeddings are respectively calculated. Next, relation embeddings are used for scale conversion to calculate the distance between the entities of the modulus components. Inspired by HAKE [30] and RotatE [31], our scheme divides the modulus component into two parts. The distance functions are as follows:

$$d_{r,m}(h_m,t_m,h_p,r_p)=d_{r,m(1)}(h_m,t_m,h_p,r_p)+d_{r,m(2)}(h_m,t_m,h_p,r_p) \tag{7}$$

$$d_{r,m(1)}(h_m,t_m,h_p,r_p)=\left\|h_m\circ r_m-t_m\right\|_2 \tag{8}$$

$$\mathrm{d}_{\mathrm{r,m(2)}}(h_m,t_m,h_p,r_p)=\left\|h_m\circ r_m-h_p\circ r_p-t_m\right\|_2 \tag{9}$$

In Eqs. (7)–(9), h_m, and t_m refer to the embedding vectors of the moduli of the head and tail entity, respectively, and r_m is the embedding of the relation. h_p refers to the embedding vector of the phase of the head entity, and r_p is the embedding of the phase of the relation. The ° symbol represents finding the Hadamard product, and $\|\cdot\|_2$ represents finding the norm.

(4) **Phase component of the knowledge reasoning layer:** Because the purpose of the fourth module phase component is to characterize the entities existing at an equal level within the semantic hierarchy, our scheme continues to use the previous modulus component and models the phase component into two parts. In the first part, the different erosion types involved in soil erosion are in the same hierarchical structure. Next, the phases of the entity embeddings are calculated in turn. Its distance is $d_{r,p(1)}$, based on the periodicity of the phase on the circle. The distance of the other part is $d_{r,p(2)}$. Our scheme allows all entities to be in the same hierarchy. Then entities and relationships are mapped to a complex vector space, and each relationship is defined as the rotation from the head to tail entities. In this manner, the two parts can be combined, with the rotation between the entities being the phase. The distance between entities combining the above two

phase component parts is as follows:

$$d_{r,p}(h_p, t_p) = d_{r,p(1)}(h_p, t_p) + d_{r,p(2)}(h_p, t_p) \tag{10}$$

$$d_{r,p(1)}(h_p, t_p) = \left\| \sin\left((h_p + r_p - t_p)/2 \right) \right\|_1 \tag{11}$$

$$d_{r,p(2)}(h_p, t_p) = \left\| h_p \circ r_p - t_p \right\|_1 \tag{12}$$

In Eqs. (10–12), h_p, and t_p refer to the embedding vectors of the phases of the head and tail entity, respectively, and r_p is the embedding of the relation.

(5) **MLP of the knowledge reasoning layer:** We divide the modulus component and the phase component into two parts. The essence of the MLP of the knowledge reasoning layer is the linear transformation from one feature space to another. Any dimension of the target space is affected by each dimension of the source space. Regardless of rigor, the target vector can be said to be the weighted sum of the source vector. So, we use the fully connected layer to update the weight parameters through iterative calculation to connect them in a more scientific manner. Eq. (13) is the combination formula of the modulus component, and Eq. (14) is the combination formula of the phase component, as follows:

$$d_{r,p} = d_{r,p} \circ W_m + b_m \tag{13}$$

$$d_{r,p} = d_{r,p} \circ W_p + b_p \tag{14}$$

In Eqs. (13) and (14), W_m and W_p are the weight matrices of the modulus and phase components, respectively, and b_m and b_p are the biases of the modulus and phase components, respectively.

(6) **Combining the knowledge reasoning layer:** The modulus and phase components of the entity are combined and mapped to polar coordinates. Radial coordinates are used to model entities at different levels of the hierarchy, and angular coordinates are used to distinguish entities at the same level of the hierarchy. Fig. 6 shows the modeling process using a subset of experimental entities.

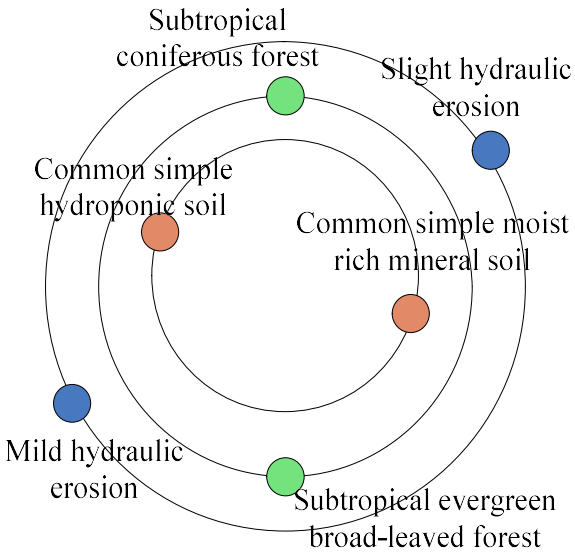

Fig. 6. Modeling display at different hierarchies.

(7) **Prediction of the knowledge reasoning layer:** By integrating the distance function of the modulus and phase components, the distance function of the whole model is obtained as:

$$d_r(h,r,t) = d_{r,m}(h_m,t_m,h_p,r_p) + \lambda d_{r,p}(h_p,t_p) \tag{15}$$

where λ is the parameter learned by the model, and the rest of the parameters are defined as given in Eqs. (5) and (6).

The score function plays an important role in the subsequent quality assessment in this study. The score function is as follows:

$$f_r(h,t) = -d_{r,m}(h,t,r) - \lambda d_{r,p}(h,t) \tag{16}$$

The distance-based score between the entities in each triplet is computed according to Eq. (16). Triplets with higher scores are ranked in ascending order, ultimately, the best-ranked triplet becomes the predicted result.

To train the model, we use a negative sampling loss function with self-adversarial training where γ serves as a margin parameter that defines the target score for the model's scoring function, σ is the sigmoid function, and (h_i', r, t_i') is the ith negative triplet. The loss function is as follows:

$$L = -\log\sigma(\gamma - d_r(h,t)) - \sum_{i=1}^{n} p(h_i',r,t_i')\log\sigma(d_r(h_i',t_i') - \gamma) \tag{17}$$

Output layer: The results of the output layer are the vegetation type and soil erosion type of the training samples. The output also shows the elevation, existing soil types, existing vegetation types, and existing soil erosion at this coordinate point.

4. Experimental Results and Analysis

4.1. Experimental Environment

Due to the large amount of data in the above datasets, it was prohibitive to use a local machine for data processing, training and testing. Therefore, a cloud server was used for training and testing, configured as follows: 14 vCPU Intel(R)Xeon(R) Gold 6330 CPU @2.00 GHz, 80 GB memory, NVIDIA GeForce RTX 3090 GPU, and 24 GB graphics card running the Linux operating system. We used the Python 3.8 programming language and the PyTorch 2.0 deep learning framework.

4.2. Evaluation Index and Hyperparameter Settings

The main task of the knowledge reasoning in our scheme is to predict soil erosion type by vegetation type, and the evaluation criterion is to judge the accuracy of the predicted soil erosion type. To accomplish this, our scheme adopts MRR and Hits@N as evaluation indicators. MRR is the average reciprocal ranking. The calculation formula of MRR is as follows:

$$MRR = \frac{1}{|S|}\sum_{i=1}^{|S|} = \frac{1}{|S|}\left(\frac{1}{\text{rank}_1} + \frac{1}{\text{rank}_2} + \cdots + \frac{1}{\text{rank}_i} + \cdots + \frac{1}{\text{rank}_{|S|}}\right) \tag{18}$$

For example, in {common simple hydroponic man-made soil, breed, subtropical coniferous forest}. each planting type in the vegetation area is replaced by "subtropical coniferous forest" to form a new triplet, and the score of each new triplet is obtained by Eq. (16) and sorted by the score. The ranking of “subtropical

coniferous forest" is $rank_i$, and the total number of triples is S. Therefore, when the ranking is small, the evaluation index MRR is greater, and the performance of our model is better.

The index Hits@N is the average proportion of triples ranked less than N. The ranking is the $rank_i$ mentioned above. The more numbers that are ranked less than N, the better the performance of the scheme. In our scheme, we consider N of 1, 3, and 10 for our evaluation.

The hyperparameter settings for our knowledge inference model are shown in Table 1. Consistent with the knowledge graph construction section in Section 2, the model inputs consist of 211 entities and 8 relationships. The entity and relationship numbers vary depending on the graph used.

The aim of the inference model is to maximize the score of positive samples (correct triplets h, r, t) and minimize the score of negative samples (incorrect triplets). The initial γ parameter is 12.0, providing a clear optimization target range for the model and ensuring a sufficient margin between the scores of the positive and negative samples to learn meaningful embeddings. Through iterative training, the positive sample scores are kept close to γ, whereas the negative sample scores are kept well below γ.

The modulus weight controls the relative importance of the modulus space score (r score) in the final total score, and the phase weight controls the relative importance of the phase space score (phase score). During iterative training, the model dynamically adjusts its reliance on these two types of information based on specific task requirements.

The MRR of the current model on the validation set is calculated every 1000 steps. If the MRR value increases, the current model parameters are saved. Otherwise, the next iterative training step is performed. In this way, the optimal model is saved during the training process.

Table 1. Hyperparameters of the knowledge reasoning scheme

Hyperparameters	Value
Epoch	50
Batch Size	32
Logging Step	100
Valid Step	1000
Learning rate	0.0001
Initial weight of modulus	1.0
Initial weight of phase	0.5
γ	12
Hidden dim	500
Number of entities	221
Number of relationships	8

4.3. Validation Results of the Knowledge Reasoning Scheme

The training set contains 1909 samples. The sample batch size used for model training is 32, and the step size for saving parameters is 100, with a total of 50,000 steps. The validation set contains 637 samples, the sample batch size for model evaluation is 4, our knowledge reasoning scheme is evaluated, and the parameters are updated every 1000 steps. Fig. 7 shows the loss curve of our scheme compared with those of HAKE and RotatE. The overall trend indicates that the loss value decreases with an increasing number of training steps. The initial loss of our scheme is significantly lower than that of the others. After completion of training, the loss of our scheme is significantly lower than that of HAKE and slightly lower than that of RotatE.

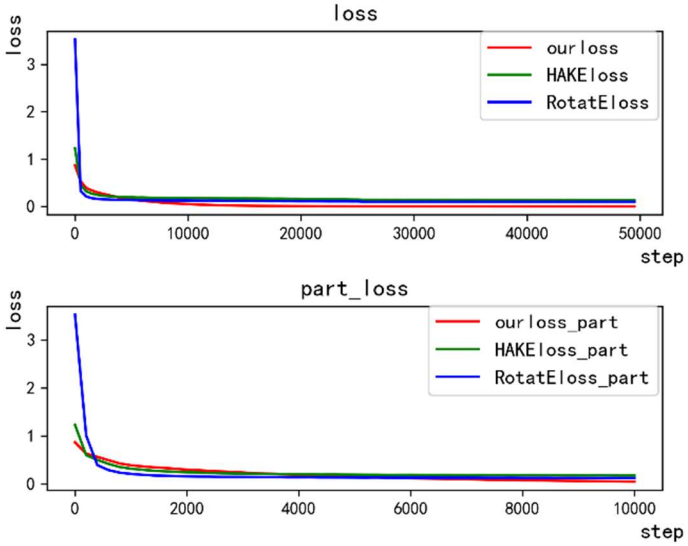

Fig. 7. Training loss curves.

Fig. 8 shows a comparison of our scheme with HAKE and RotatE in terms of MRR parameters. The overall trend indicates that the MRR value increases with an increasing number of training steps. For evaluating the model, we consider the comprehensive indicator MRR to be an important evaluation metric for saving model parameters. At 2000 steps, our scheme outperforms the other two schemes, and the effect is best over the entire process. Our scheme saves the testing parameters in the test set. At 2000 steps, our other evaluation indicators are also better than those of the other two schemes.

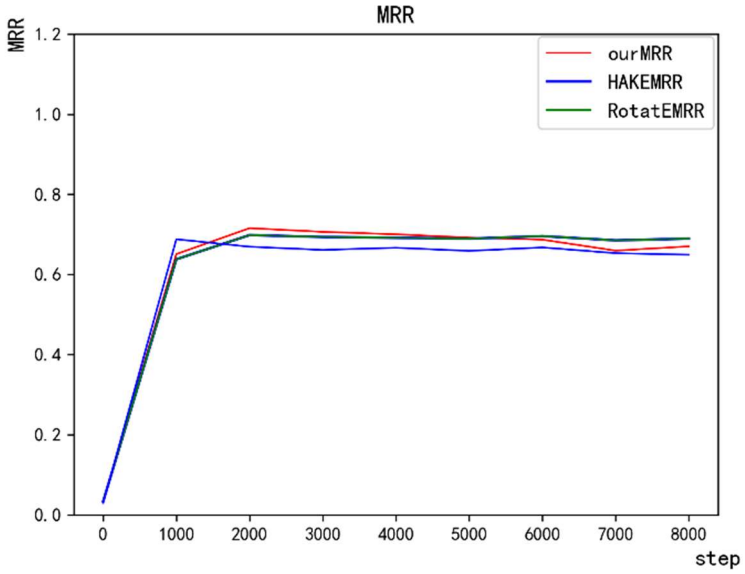

Fig. 8. MRR change curve.

Fig. 9 shows the Hits@1 values of the proposed scheme and of HAKE and RotatE. The overall trend is that the value of Hits@1 increases with an increasing number of training steps. Due to the small value of N and high variability of the validation set results, the three schemes' Hits@1 values fluctuate after 3000 training steps. Our scheme performs better than the others at 2000 steps.

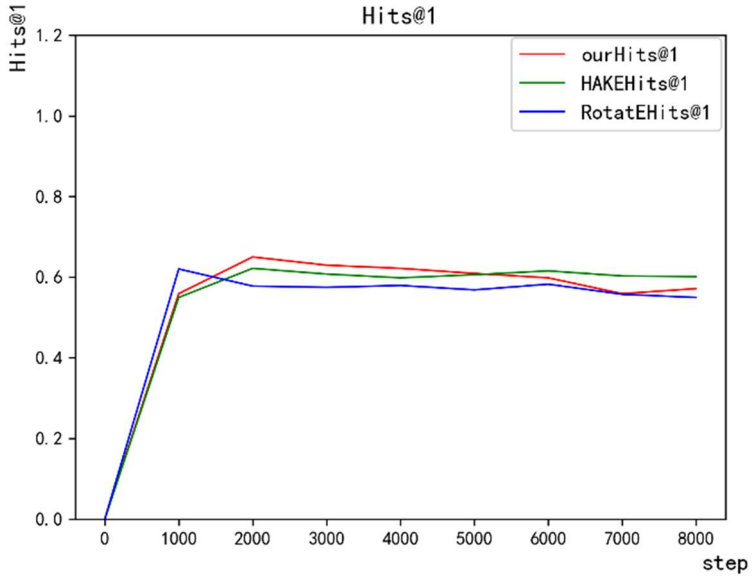

Fig. 9. Hits@1 change curve.

Fig. 10 shows the Hits@3 values of our scheme and of HAKE and RotatE. The overall trend is the value of Hits@3 increases with an increasing number of training steps. At 2000 steps, our scheme has slightly higher Hits@3 than the others.

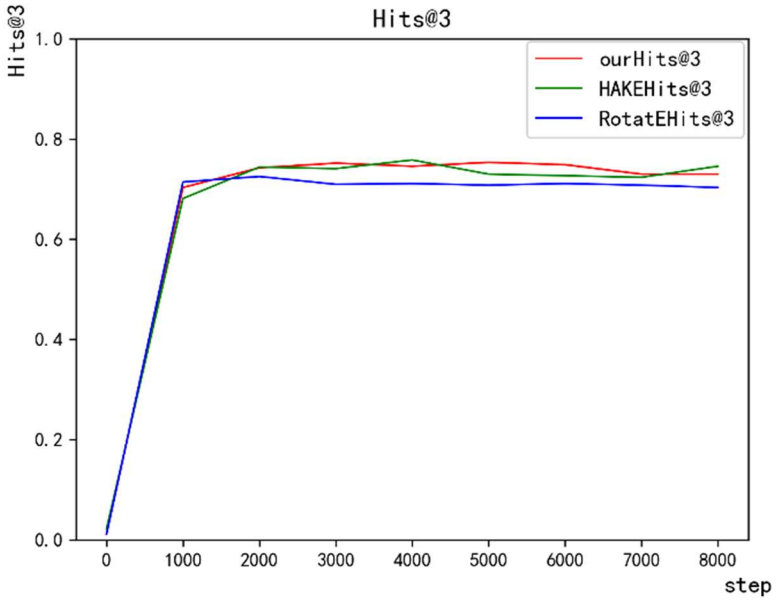

Fig. 10. Hits@3 change curve.

Fig. 11 shows the Hits@10 values of our scheme and of HAKE and RotatE. The overall trend is that the value of Hits@10 increases with an increasing number of training steps. During the entire training process, there is almost no difference between the three schemes. At 2000 steps, our scheme has nearly the same Hits@10 as the HAKE scheme and slightly higher Hits@10 than RotatE.

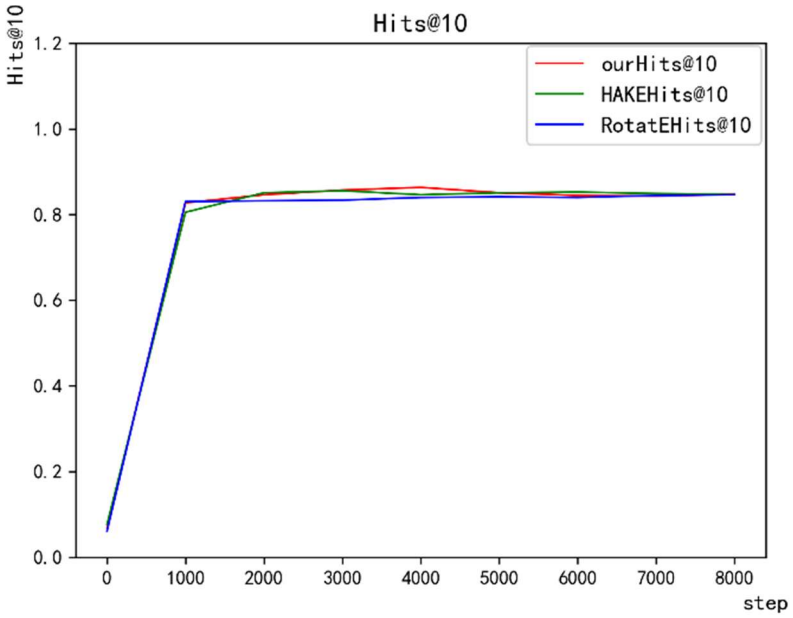

Fig. 11. Hits@10 change curve.

4.4. Test Results of the Knowledge Reasoning Scheme

We determine the parameters of the model through training. At this stage, we use the trained model to reason about the test set. The model is trained within an allowable error range. This no-attention scheme is essentially an ablation experiment with no added attention decoding block. The results of the knowledge reasoning model in our scheme are evaluated by the evaluation indicators described in Section 3.2, as shown in Table 2.

Table 2. Evaluation results.

Scheme	MRR	Hits@1	Hits@3	Hits@10
Our scheme	0.759090	0.706436	0.781790	0.863422
No-attention	0.755310	0.698587	0.781790	0.861852
HAKE	0.740938	0.673469	0.773940	0.854003
RotatE	0.719797	0.646782	0.758242	0.860283

In Table 2, the MRR index represents average reciprocal ranking, so the result should be as large as possible. The MRR of our scheme is 0.759090, which is greater than those of HAKE and RotatE. In addition, the maximum value of this parameter is 1, and our result is close to 1. The test results of this parameter indicate that our scheme is relatively effective. In addition, as seen in Table 1, when N is set to 1, 3 and 10, the Hits@N indicators are 0.706436, 0.781790 and 0.863422, respectively. When N is 1, our scheme performs significantly better than HAKE and RotatE, indicating that our scheme performs best in terms of prediction accuracy. When N is 3, the precision requirement is slightly lower, and our scheme still performs better than the other two schemes. When N is 10, the accuracy requirement is low, and most of the results of our scheme are in the accurate region. Here, our scheme achieves almost the same performance as the Rotate scheme and slightly higher than the HAKE scheme. Overall, our scheme is the most accurate scheme.

4.5. Analysis of Reasoning Results

As seen in Fig. 12, the knowledge inference input consists of latitude and longitude coordinates. Information on the soil type and elevation is obtained through the unified generation and feature extraction layer.

The knowledge reasoning model is invoked twice to obtain the predicted vegetation type and soil erosion type.

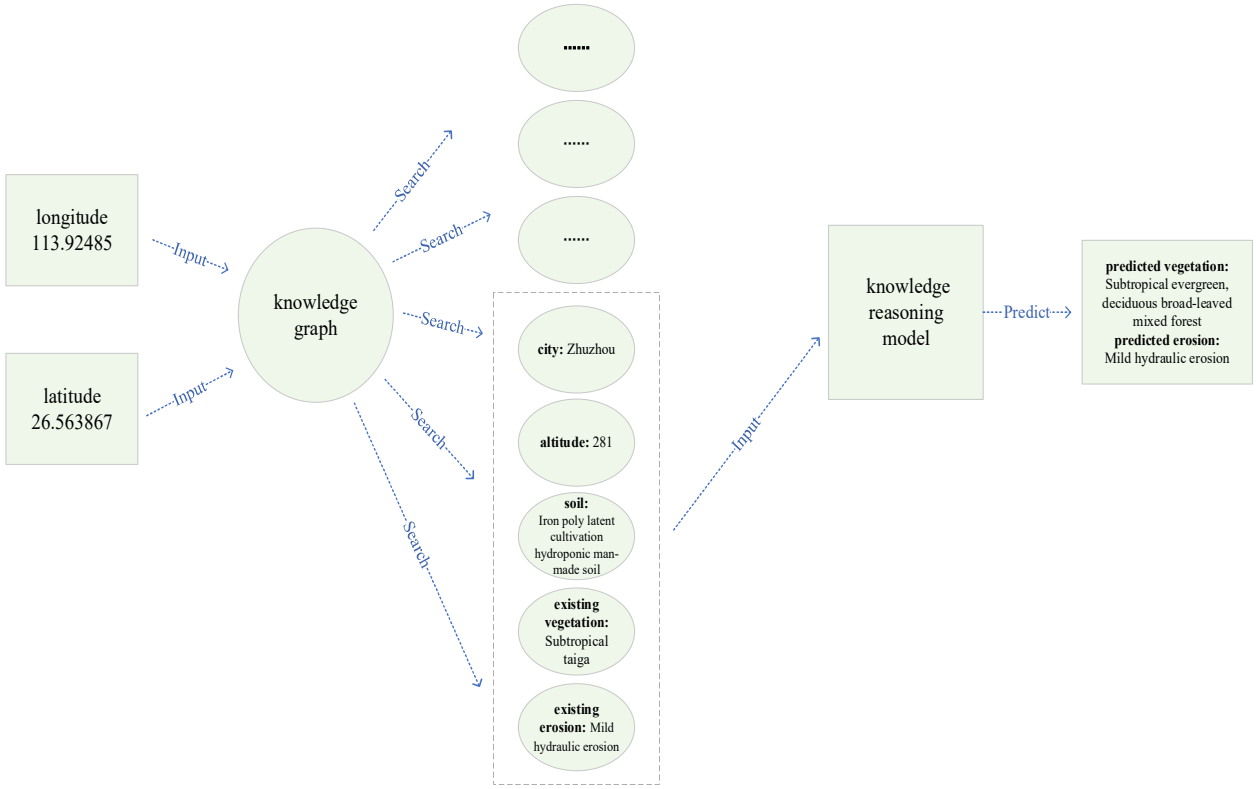

Fig. 12. Knowledge reasoning results.

Fig. 12 shows the inference results of the knowledge reasoning scheme based on specific input data. The output also shows the elevation, existing soil types, existing vegetation types, and existing soil erosion at this coordinate point. The results indicate that we have gained a clearer understanding of this information at these latitude and longitude coordinates.

5. Conclusions

In this paper, we construct a knowledge graph that associates the entity information of Tu ethnicity, altitude, and soil erosion to the same coordinate point based on the definition of minimum Euclidean. Furthermore, based on established relationships, a knowledge reasoning scheme is proposed by referencing the modulus and phase components of the entity for knowledge reasoning, which is applied to a case study of vegetation and soil erosion type prediction in Hunan Province. Our experiments show that the proposed scheme can effectively infer vegetation types and soil erosion types, based on prediction indicators such as MRR and Hits@N. The MR index represents the average ranking. The ranking should be as small as possible. Our scheme performs slightly better than HAKE and much better than RotatE based on MR. The MRR index represents the average reciprocal ranking, so the results should be as large as possible. The MRR obtained in our scheme is 0.759090, which is greater than those of HAKE and RotatE. When N is 1, 3, and 10, the results of the Hits@N indicators are 0.706436, 0.781790, and 0.863422, respectively. When N is 1 or 3, our results are much better than those of HAKE and RotatE. When N is 10, the accuracy requirement is low, and most of the results of our scheme are in the accurate region. This signifies accurate model prediction, and the results are relatively good. The constructed knowledge graph has good adaptability in the field of geographic knowledge specialization. A knowledge graph representation system can be used to transform natural resource data into content that can be recognized and processed by computers, promoting further research on natural resource

data knowledge in the field of artificial intelligence. In the future, we will integrate more natural resource databases to expand the knowledge graph. A large-scale and high-quality geographic knowledge graph provides an important foundation for people to quickly acquire, infer, and apply natural resource knowledge.

Conflict of Interest

The authors declare no conflict of interest.

Author Contributions

Hao He and Fan Lei were responsible for funding application and management; Zhichao Zhang and Yifan Hao were responsible for data collection and analysis; Yabo Liu, Zhichao Zhang, Yifan Hao, Fan Lei, and Hao He were responsible for experimental design and implementation; Yabo Liu, Zhichao Zhang, and Yifan Hao were responsible for result analysis; Yabo Liu, Zhichao Zhang, and Yifan Hao were responsible for paper writing. All authors had approved the final version.

Acknowledgment

This work was supported by the National Natural Science Foundation of China (No. 32200546), Open Topics in the Key Laboratory of Natural Resources Monitoring and Supervision in the Southern Hilly Region, Ministry of Natural Resources (NRMSSHR-2022-Y03), the Hebei Natural Science Foundation (No. C2024202003), and the Digital Grid Open Fund of China Southern Power Grid Co., Ltd. under Grant DPGCSG-2024-KF-35.

References

[1] Ma, X.; Ma, C.; & Wang, C. (2020). A new structure for representing and tracking version information in a deep time knowledge graph. *Computers & Geosciences*, *145*, 104620.

[2] Zhang, X. Y., Zhang, C. J., Wu, M. G., *et al.* (2020). Spatio-temporal features based geographical knowledge graph construction. *Sci Sin Inform*, *50*, 1019–1032. (in Chinese). doi: 10.1360/SSI-2019-0269

[3] Haiqi, Wang, Qiong, W., Like, L. I., *et al.* (2023). Construction and analysis of poverty alleviation geographic knowledge graph:A case study of Linyi city. *Geography and Geo-Information Science*, *39(4)*. doi:10.3969/j.issn.1672-0504.2023.04.001

[4] Ma, X. (2022). Knowledge graph construction and application in geosciences: A review. *Computers & Geosciences*, *161*, 105082.

[5] Mantovani, A., Piana, F., & Lombardo, V. (2020). Ontology-driven representation of knowledge for geological maps. *Computers & Geosciences*, *139*, 104446.

[6] Myburgh, P. H. (2022). Infodemiologists beware: Recent changes to the google health trends API result in incomparable data as of 1 January 2022. *Int. J. Environ. Res. Public Health*, *19(22)*, 15396. https://doi.org/10.3390/ijerph192215396

[7] Wang, X., Li, Y., Wang, H., & Lv, M. (2023). MKBQA: Question answering over knowledge graph based on semantic analysis and priority marking method. *Appl. Sci., 13(10)*, 6104. https://doi.org/10.3390/app13106104

[8] Wang, S., Zhang, X., Ye, P., *et al.* Geographic knowledge graph (GeoKG): A formalized geographic knowledge representation. *ISPRS Int. J. Geo-Inf.*, *8(4)*, 184. https://doi.org/10.3390/ijgi8040184

[9] Hoffart, J., Suchanek, F. M., Berberich K., *et al.* (2013). YAGO2: A spatially and temporally enhanced knowledge base from Wikipedia. *Artificial Intelligence*, *194*, 28–61.

[10] Mahdisoltani, F., Biega, J., & Suchanek, F. M. (2014). Yago3: A knowledge base from multilingual wikipedias. *Proceedings of the 7th Biennial Conference on Innovative Data Systems Research (CIDR)*.

[11] Chatzakis, M., Mountantonakis, M., & Tzitzikas, Y. (2021). RDFsim: Similarity-based browsing over dbpedia using embeddings. *Information*, *12(11)*, 440. https://doi.org/10.3390/info12110440

[12] Carlson, A., Betteridge, J., Kisiel, B., *et al.* Toward an architecture for never-ending language learning. *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence: Vol. 24* (pp. 1306–1313).

[13] Mitchell, T., Cohen, W., Hruschka, E., *et al*. (2018). Neverending learning. *Communications of the ACM*, *61(5)*, 103–115.

[14] Etzioni, O., Cafarella, M., Downey, D., *et al.* (2004). Webscale information extraction in knowitall:(preliminary results). *Proceedings of the 13th International Conference on World Wide Web* (pp. 100–110).

[15] Wu, W., Li, H., Wang, H., *et al.* (2012). Probase: A probabilistic taxonomy for text understanding. *Proceedings of the2012 ACM SIGMOD International Conference on Management of Data* (pp. 481–492).

[16] Liu, J. N., Liu, H. Y., Chen, X. H., *et al.* (2020). The Construction of knowledge graph towards multi-source geospatial data. *Journal of Geo-information Science*, *22(7)*, 1476–1486. doi:10.12082/dqxxkx.2020.190565

[17] Monika, S., Jokar Arsanjani, J., Klammer, R., *et al*. (2014). Integrating and generalising volunteered geographic information. *Prcoeedings of the Abstracting Geographic Information in a Data Rich World* (pp. 119–155).

[18] Wu, Z., Xu, Y., Yang, Y., Zhang, C., Zhu, X., & Ji, Y. (2017). Towards a semantic web of things: A hybrid semantic annotation, extraction, and reasoning framework for cyber-physical system. *Sensors, 17(2)*, 403. https://doi.org/10.3390/s17020403

[19] Ballatore, A., Bertolotto, M., & Wilson, D. C. (2013). Geographic knowledge extraction and semantic similarity in OpenStreetMap(Article). *Knowledge and Information Systems*, *37(1)*, 61–81.

[20] Hoffart, J., Suchanek, F. M., Berberich, K. & Weikum, G. (2013). YAGO2: A spatially and temporally enhanced knowledge base from Wikipedia. *Artificial Intelligence*, *194*, 28–61.

[21] Hu, W., Li, H., Sun, Z., Qian, X., Xue, L., Cao, E., & Qu, Y. (2016). Clinga: Bringing Chinese physical and human geography in linked open data. *Proceedings of the International Semantic Web Conference* (pp. 104–112).

[22] Zhang, J. J., Zhang, L., Zhong, H. T., *et al.* (2023). Construction and application of lithofacies paleogeography knowledge graphs. *Geological Journal of China Universities*, *29(3)*, 345–358. (in Chinese). doi:10.16108/j.issn1006-7493.2023027

[23] Ma, Z. G., Ni, R. Y., & Yu, K. H. (2020). Recent advances, key techniques and future challenges of knowledge graph. *Chinese Journal of Engineering*, *42(10)*, 1254–1266.

[24] Guan, S. P., Jin, X. L., Jia, Y. T., Wang, Y. Z., & Cheng, X. Q. (2018). Knowledge reasoning over knowledge graph: A survey. *Journal of Software*, *29(10)*, 2966–2994.

[25] Banko, M., Cafarella, M. J., Soderland, S., Broadhead, M., & Etzioni, O. (2007). Open information extraction from the web. *International Joint Conference on ArtificialIntelligence*, 2670–2676.

[26] Wu, F. & Weld, D. S. (2010). Open information extraction using wikipedia. *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics* (pp. 118–127).

[27] Geographic remote sensing ecological network scientific data registration and publication system. Land use data of Hunan Province in 2020. Retrived October 17, 2023, from www.gisrs.cn

[28] Geographic remote sensing ecological network scientific data registration and publication system. Spatial distribution data of vegetation types. Retrived October 17, 2023, from www.gisrs.cn

[29] Data Center for Resources and Environmental Sciences, Chinese Academy of Sciences. Spatial distribution data of soil erosion in China. Retrived October 17, 2023, from http://www.resdc.cn

[30] Zhang, Z., Cai, J., Zhang, Y., *et al.* (2020). Learning hierarchy-aware knowledge graph embeddings for link prediction. *Proceedings of the AAAI Conference on Artificial Intelligence: Vol. 34* (pp. 3065–3072).
[31] Sun, Z., *et al.* (2019). RotatE: Knowledge graph embedding by relational rotation in complex space. *arXiv preprint*, arXiv:1902.10197.