Research on the Construction of Course Resources Based on Educational Knowledge Graph

Shuguang Cui, Haixia Wang*, Hai Huang, Lei Xiang

College of Information Engineering, Hunan Automotive Engineering Vocational University, Zhuzhou, China

*csg06282023@126.com

Abstract. In view of the current problems in teaching resource construction, such as low construction efficiency, difficult resource application, and insufficiently intelligent learning, taking the "Java Programming" course as an example, the OWL language is used to construct an ontology for building a generalized ontology of the knowledge graph of the Java Programming course. The Cypher language is then utilized to generate a structured knowledge semantic network with a clear structure and definite relationships, achieving the construction and intelligent application of the course resources of the Java Programming knowledge graph. In combination with the educational knowledge graph, research on the upgrade of teaching resource attributes has been accomplished, facilitating the construction of intelligent and personalized "smart" teaching resources. Practice has proven that the course construction based on the knowledge graph can effectively improve the resource construction efficiency of the course, enhance the application status of the course resources, and provide learners with favorable learning paths.

Keywords: artificial intelligence, educational knowledge graph, teaching resource construction

1 Introduction

Digital teaching resources are the core elements of the innovative development of information-based education and teaching. Since its implementation in 2010, more than 200 national-level resource repositories have formed more than 4 million granular resources. However, while the quantity of teaching resources is already very rich, it also faces problems such as laborious construction, difficult application, and insufficient intelligence.

S. Yang et al. (eds.), *Proceedings of the 2024 Seminar on Educational Technology and Management Information Systems (ETMIS 2024)*, Advances in Social Science, Education and Humanities Research 911,
https://doi.org/10.2991/978-94-6463-674-1_16

1.1 The Knowledge Production Model is Single and the Quality is Uneven

There are three knowledge production models: excavation type, construction type, and growth type[1]. At present, due to the constraints of its display form, higher vocational teaching resource construction usually adopts excavation type and construction type. Their common point is to present knowledge as a result to learners, and require that digital teaching resources must be complete knowledge products such as courseware, micro-lessons, and virtual software. This brings great challenges to resource builders. Surveys show that all teachers think that resource construction is "very labor-consuming." Many people who have experienced the construction of resource repositories think that the teaching resources in the resource repository are "abundant but not excellent[2]."

1.2 The Growth of Knowledge is Weak, and it is Difficult to Update and Maintain Teaching Resources

At present, resource updates mainly rely on manual re-production. The dynamics and growth of knowledge are very weak. It is not only time-consuming and labor-intensive, but also the resource content is always lagging behind the development of the industry and the times. Close integration with the industry is the characteristic of vocational education. Teaching resources that cannot be updated in time and reflect changes in the industry and positions cannot be recognized. Resources have poor adaptability and lack feedback channels, which directly leads to unsatisfactory resource application effects.

1.3 Inefficient Knowledge Connection and Organization, and Insufficient Logicality in the Organization of Teaching Resources

The existing teaching resources have unclear organization systems and loose relationships[3]. Resource repositories and excellent courses still use the "knowledge tree" formed by the tree-shaped course syllabus as the basis for organizing teaching resources, following the linear logic of one-way growth[4], rather than forming a "network-like" knowledge structure. This makes it impossible to efficiently and intelligently organize and manage the endogenous elements such as the knowledge structure and corresponding ability levels of digital teaching resources themselves, and it is difficult to construct a good learning path for learners.

2 Related Work

2.1 Research Status of Related Technical Frameworks of Educational Knowledge Graph

In the research of general knowledge graphs, Liu Qiao et al. proposed a technical framework for knowledge graphs, which can be divided into three levels: information extraction layer, knowledge fusion layer, and knowledge processing layer[5]. Xu Zenglin et al. proposed that the technologies involved in knowledge graphs include four aspects: knowledge extraction, knowledge representation, knowledge fusion, and knowledge reasoning[6]. Qi Guilin et al. proposed that the key technologies for constructing knowledge graphs include entity relationship recognition technology, knowledge fusion technology, entity linking technology, etc[7]. The construction methods of knowledge graphs include completely manual construction by experts, construction using crowdsourcing (such as Wikipedia and Freebase), semi-automatic construction (such as presetting rules or regular expressions manually first and then performing automatic construction from semi-structured information), and completely automatic construction from unstructured information[8].

2.2 The Current Research Status of the Application of Educational Knowledge Graph in the Construction of Teaching Resources

In the field of higher vocational education, some scholars have paid attention to the lack of existing teaching resources and the application research of educational knowledge mapping. Bilal Abu Salih[9] highlight the specific KG functionalities, knowledge extraction techniques, knowledge base characteristics, resource requirements, evaluation criteria, and limitations. Mzwri K[10] has proposed a method to enhance the knowledge base of open-domain question-answering chatbots by leveraging feature snippets, knowledge graphs, and organic search, combined with data science and natural language models. This approach enables chatbots to dynamically access a wide range of up-to-date knowledge across the web.Hu Guangyong[11] combined the characteristics of vocational technical skills education and constructed a framework for the knowledge map of vocational education software technology professional knowledge and skill system. Based on the analysis of the basic characteristics of online course knowledge graphs, Guo Hongwei proposed two ways to construct online course knowledge graphs in universities [12], but his work focused on manual annotation models and the construction process lacked "intelligence". He Wentao explored how to apply the educational knowledge graph to the construction of teaching materials, and proposed a new approach to analyzing the knowledge content of teaching materials based on the knowledge graph as the data foundation [13]. However, its work is limited to the construction of static knowledge maps based on the content of textbooks, and does not involve the work of dynamic event maps that represent teaching events or activities as the object of representation. Therefore, there are deficiencies in the application of "intelligence".

3 Method

3.1 Content Recognition and Encoding Method

The process of organizing knowledge can be automated through the use of artificial intelligence. The construction process of the knowledge graph mainly includes data acquisition, knowledge extraction (pattern layer and data layer), knowledge fusion, etc. Combining the two key elements of knowledge organization, namely "knowledge unit" and "organization method", to ensure that all analysis units are attributed, the priori knowledge organization process categories are determined as follows: knowledge extraction (pattern layer includes concept/term, attribute/relation extraction, data layer includes entity, attribute/relation extraction), knowledge fusion (entity alignment and entity linking), artificial intelligence technology for "machine learning", "algorithm model", "natural language processing tool" and other artificial intelligence technology, algorithm, tool involved in the knowledge organization process. The coding method for the analysis unit was finally determined as "knowledge organization process: artificial intelligence technology". During the process of identifying analysis units, it is necessary to unify the diverse analysis units being identified, such as converting Chinese-language algorithm models into English abbreviations; Different orders of combined algorithms can lead to inconsistent results. To facilitate statistical analysis and eliminate the influence of the original algorithm order, the results are adjusted to a uniform order based on the coding results.

3.2 Research Methods of Entity Recognition and Relationship Extraction based on Deep Learning

Entity extraction refers to accurately identifying and extracting named entities with practical significance from datasets in specific fields or large-scale resources in the open domain of the Internet by using artificial intelligence technology. Deep learning allows computational models composed of multiple processing layers to learn data representations with multiple abstraction levels. By using the backpropagation algorithm to discover complex structures in large data sets, it can automatically learn various semantic features in data and avoid the complexity of manual feature extraction, thus being widely used in the process of entity recognition and extraction. For entity recognition, an automatic role labeling method based on Hidden Markov Model (HMM) is adopted. When labeling, the close connection between words is applied. The HMM sequence determines the labeling by using the transition probability between states and the emission probability of words. The word2vec representation method is used to train word vectors to replace the structural components of named entities. For entity relationship extraction, a relationship extraction model based on convolutional neural network is attempted. Sentence features are extracted through convolutional neural network, and then the sentence features are connected with entities and the features of entities. Finally, the features are classified by neural network.

4 Implementation Steps

Step 1: Collect domain knowledge

Taking "Java Programming" as the object, from the perspective of organizing knowledge and skill points, combine manual work with web crawling technology to collect data from data sources such as professional documents, teaching resources on teaching platforms, professional management platforms, textbooks, and open knowledge bases such as online encyclopedias. Discuss with relevant domain experts and course experts, comprehensively and reasonably divide the course knowledge, extract entities and entity attributes, and construct a knowledge graph dataset for the course.

Step 2: Define the relationship between knowledge points

Use knowledge modeling technology to define the subordination relationship between knowledge. Use specialized vocabulary to express superordinate-subordinate relationship, coordinate relationship, constituent elements of a concept or components of a function, attribute characteristics of a concept, examples of a concept, classification of a concept, calculation formula of a concept, principle process steps, logical relationship between two principles, support relationship, etc. The superordinate-subordinate relationship is formally described as: Let all Java classes constitute a set C. $Object$ is the superclass of all classes. For any custom subclass C_i.That is, C_i satisfies $C_i <: Object$ (here "$<:$" represents inheritance relationship). The formal description of coordinate relationship is: Suppose there is a parent class FatherClass and there are two subclasses SubClass1 and SubClass2. The coordinate relationship between SubClass1 and SubClass2 can be expressed as:SubClass1 <: FatherClass and SubClass2 <: FatherClass and SubClass1!= SubClass2.

Step 3: Construct the knowledge graph of Java programming course

Using OWL language suitable for fine-grained Chinese domain ontology description, according to the ontology characteristics to be constructed and the principle of progressive differentiation, under the guidance of domain experts and course experts, fully sort out and utilize existing teaching resources. Combined with the automatic construction method based on ontology learning and using machine learning related technologies, mine the relationships between entities and construct an ontology of the knowledge graph of Java programming course with generalization ability.

a. Entity Recognition Based on the Hidden Markov Model (HMM)

(1)Definition of the HMM Model

Suppose we have the following roles (states): $S1$:Entity, $S2$:Attribute, $S3$:Relation. The observation set (vocabulary) can be the words extracted from the text, for example: $O1$:class, $O2$:extends, $O3$:hasField, $O4$:String, $O5$:int.

(2) Prepare the Training Data Suppose there are the following labeled training data: Sequence 1: $S1 -> O1\,(\text{class}), S2 -> O3\,(\text{hasField}), S1 -> O4\,(\text{String})$

Sequence 2: $S1 -> O1\,(\text{class}), S2 -> O2\,(\text{extends}), S1 -> O5\,(\text{int})$

(3) Train the HMM Model The initial state probability distribution ( ) can be estimated by calculating the frequency at which each state appears at the initial position.

$$\begin{aligned}
\pi = [&\\
&P(S1) = 2/2 = 1.0,\\
&P(S2) = 0/2 = 0.0,\\
&P(S3) = 0/2 = 0.0\\
]&
\end{aligned}$$

The state transition probability matrix A can be estimated by calculating the frequency of transitioning from one state to another.

$$
\begin{aligned}
&\mathrm{A} = [\\
&\quad [\mathrm{P(S1 \mid S1)} = 0/1 = 0.0, \mathrm{P(S2 \mid S1)} = 1/1 = 1.0, \mathrm{P(S3 \mid S1)} = 0/1 = 0.0],\\
&\quad [\mathrm{P(S1 \mid S2)} = 1/1 = 1.0, \mathrm{P(S2 \mid S2)} = 0/1 = 0.0, \mathrm{P(S3 \mid S2)} = 0/1 = 0.0]\\
&]
\end{aligned}
$$

The observation probability matrix B can be estimated by calculating the frequency of generating a certain observation under each state.

$$
\begin{aligned}
&\mathrm{B} = [\\
&\quad [\mathrm{P(O1 \mid S1)} = 1/1 = 1.0, \mathrm{P(O2 \mid S1)} = 0/1 = 0.0,\\
&\quad\ \mathrm{P(O3 \mid S1)} = 0/1 = 0.0, \mathrm{P(O4 \mid S1)} = 1/1 = 1.0,\ \mathrm{P(O5 \mid S1)} = 0/1 = 0.0],\\
&\quad [\mathrm{P(O1 \mid S2)} = 0/1 = 0.0, \mathrm{P(O2 \mid S2)} = 1/1 = 1.0,\\
&\quad\ \mathrm{P(O3 \mid S2)} = 1/1 = 1.0, \mathrm{P(O4 \mid S2)} = 0/1 = 0.0, \mathrm{P(O5 \mid S2)} = 0/1 = 0.0]\\
&]
\end{aligned}
$$

(4) Use the Viterbi Algorithm for Tagging

The Viterbi algorithm is used to find the most likely state sequence given an observation sequence. The specific steps are as follows:

Initialization:

$$
\begin{aligned}
&\delta_1(i) = \pi_i B_i(O_1)\\
&\psi_1(i) = 0
\end{aligned}
$$

Recursion:

$$
\begin{aligned}
&\delta_t(j) = \max_{1<=i<=N} (\delta_{t-1}(i) A_{ij}) B_j(O_t)\\
&\psi_t(j) = \arg\max_{1<=i<=N} (\delta_{t-1}(i) A_{ij})
\end{aligned}
$$

Termination:

$$
\begin{aligned}
&\mathrm{P}^* = \max_{1<=i<=N} \delta_{\mathrm{T}}(\mathrm{i})\\
&\mathrm{q}_{\mathrm{T}}^* = \arg\max_{1<=i<=N} \delta_{\mathrm{T}}(\mathrm{i})
\end{aligned}
$$

Path Backtracking:

$$
\mathrm{q}_{\mathrm{t}}^* = \psi_{\mathrm{t+1}}(\mathrm{q}_{\mathrm{t+1}}^*)
$$

(5) Optimization and Improvement of the HMM Model Using Smoothing Techniques to Improve Parameter Estimation: Words or state transitions that do not appear in the training data of the HMM model may lead to zero probabilities, which affects

the generalization ability of the model. We adopt Add-One Smoothing: adding one to all counts to avoid zero probabilities.

$$P(w_i \mid w_{i-1}) = \frac{C(w_{i-1}, w_i) + 1}{\sum_{w'} (C(w_{i-1}, w') + 1)}$$

Enhancing Part-of-Speech Information with n-gram Features to Improve the Discrimination of Different Entity Types:

$$\mathrm{P}(y_i \mid x_i, pos_i) = \frac{C(x_i, pos_i, y_i)}{C(x_i, pos_i)}$$

b.Adopting Piecewise Convolutional Neural Networks (PCNNs) of Multi-Instance Learning for Entity Relation Extraction

(1) Data Preparation Word Segmentation:

Split a sentence into words or phrases:For example, the sentence "Java is a programming language" is segmented into ["Java", "is", "a", "programming", "language"].

Word Indexing: Map each word to a unique index. For example, "Java" is mapped to 1, "is" is mapped to 2, and so on.

Relation Label Encoding: Map relation labels (such as "programming language", "development", "acquisition") to unique numerical labels. For example, "programming language" is mapped to 0, "development" is mapped to 1, and "acquisition" is mapped to 2.

(2) Model Construction

Input Layer:Accept the word index sequence of a sentence. Assuming the maximum length of a sentence is max_len, the shape of the input layer is (batch_size, max_len), where batch_size is the batch size.

Embedding Layer: Convert word indices into word vectors. The input dimension of the embedding layer is the vocabulary size vocab_size, and the output dimension is the embedding dimension embedding_dim.

Piecewise Convolutional Layer: Divide the sentence into three parts according to the position information of entities. Use multiple convolutional kernels to extract local features. The convolutional kernel adopted in this experiment is 5. The number of each convolutional kernel is set to 100. The ReLU activation function is used. Piecewise Pooling Layer: Select the maximum value from the convolution results of each segment and concatenate the maximum pooling results of the three segments. Use the Softmax activation function and output the probability of each relation category.

Fully Connected Layer: Map the concatenated feature vectors to relation categories. (3) Model Training

Loss Function: Use the cross-entropy loss function (Sparse Categorical Crossentropy) suitable for multi-classification tasks.

Optimizer: Use the Adam optimizer because it combines the advantages of momentum and adaptive learning rate.

Learning Rate: The initial learning rate is set to 0.001 and adjusted according to the training situation.

Batch Size: Set to 32. Training Epochs: Since the dataset is not large, it is set to 10.
(4) Model Optimization and Improvement

Introduce an attention mechanism to enable the model to focus on important parts of the sentence. Improve the model's ability to capture key information and reduce the interference of irrelevant information. Use the Deep Residual Network (ResNet) structure to solve the problem of gradient vanishing in deep networks and improve the optimization effect of the model for multi-classification tasks. Use Focal Loss to reduce the weights of easy-to-classify samples and increase the weights of hard-to-classify samples.

Step 4: Persistent storage of the knowledge graph of Java programming course

The multi-disciplinary knowledge entity and semantic relationship cross-modal multivariate knowledge model obtained through training is persistently stored in a visual graph database (such as Neo4j graph database). The following is the code for creating nodes and relationships:

```
String createClassNodeQuery = "CREATE (c:Class {name: 'MyClass'})";
session.run(createClassNodeQuery);
String createObjectNodeQuery = "CREATE (o:Object {name: 'obj1'})";
session.run(createObjectNodeQuery);
String createRelationshipQuery = "MATCH (c:Class), (o:Object) WHERE c.name = 'MyClass' AND o.name = 'obj1' CREATE (c)-[:IS_OBJECT_OF]->(o)";
session.run(createRelationshipQuery);
String createIntegerDataTypeNodeQuery = "CREATE (idt:IntegerDataType {name: 'int'})";
session.run(createIntegerDataTypeNodeQuery);
```

Based on graph data visualization knowledge organization languages (such as Cypher language, etc.), a structured knowledge semantic network is generated. The following uses Cypher language to express the corresponding relationship between "Data Types" and "Variables": Based on graph data visualization knowledge organization languages (such as Cypher language, etc.), a structured knowledge semantic network is generated. The following uses Cypher language to express the corresponding relationship between "Data Types" and "Variables":

```
CREATE (:Concept {name: 'Variables'})
MATCH (c1:Concept {name: 'Data Types'}), (c2:Concept {name: 'Variables'})
CREATE (c1)-[:USED_BY]->(c2)
```

Step 5: Construction and intelligent application of Java programming

Reorganize the existing teaching resources and supplement resources through the knowledge graph. Develop diverse course teaching resources centered around a specific knowledge point to enhance learners' understanding and application ability of knowledge. Form a free combination of resources based on the knowledge graph and construct a personalized Java programming course that meets the needs of students.

5 Experimental Results and Analysis

To verify the superiority of the proposed model in terms of performance, we conducted tests on the named entity recognition task using the CoNLL-2003 dataset. A comparison was made between our improved Hidden Markov Model (HMM) and the traditional HMM model. The detailed results are shown in Table 1. Judging from the results, significant improvements were achieved in terms of accuracy, precision, recall, and F1 score. Similarly, when it came to the entity relation extraction task, a comparison and analysis were carried out with the entity recognition benchmark models. The specific comparison results are presented in Table 2. As can be seen from the results, compared with the traditional Piecewise Convolutional Neural Networks (PCNNs), the improved PCNN model witnessed remarkable enhancements in accuracy, precision, recall, and F1 score. Specifically, the F1 score increased from 89.2% to 93.0%, representing an improvement of 3.8 percentage points.

Table 1. Experimental results on the named entity recognition task on CoNLL-2003.

DataSet	Model	Accuracy	Precision	Recall	F1 Score
CoNLL-2003	HMM	85.2%	84.5%	83.8%	84.1%
CoNLL-2003	Ours	90.1%	89.5%	89.0%	89.2%

Table 2. Experimental results on the relation extraction task on Chinese DuIE

DataSet	Model	Accuracy	Precision	Recall	F1 Score
CoNLL-2003	BiLSTM-CRF	89.5%	88.7%	88.2%	88.4%
CoNLL-2003	PCNN	90.1%	89.5%	89.0%	89.2%
CoNLL-2003	Ours	93.5%	93.1%	92.8%	93.0%

6 Conclusions

The construction of Java programming course resources based on knowledge graph effectively integrates various kinds of knowledge of Java programming course, including grammar rules, programming concepts, design patterns, etc., forming a structured and systematic knowledge system. Through the knowledge graph, teachers can organize teaching contents more conveniently. According to different teaching goals and students' needs, they can quickly extract relevant knowledge points and cases to enrich teaching resources. At the same time, students can also conduct autonomous learning with the help of knowledge graph, clearly understand the associations and hierarchical structures between knowledge, and improve learning efficiency and effectiveness. It has opened up a new way for the teaching reform and innovative development of Java programming courses.

Acknowledgment

This work is supported by "Research on the Construction of Higher Vocational Curriculum Resources Based on Educational Knowledge Graph from the Perspective of “Artificial Intelligence +” —Taking the Course of “Java Programming” as an Example." (Number: HQJY2313) and which is the Educational and teaching reform research project at school level in 2023 of Hunan Automotive Engineering Vocational University.

References

1. Downes,S. Connectivism and Connective Knowledge[EB/OL]. https://www.downes.ca/files/books/ Connective_Knowledge- 19May2012.pdf,2012.
2. Liu Xingshu, Guan Zhiwei, Yin Wanjian, et al. Current situation, problems and strategies of the construction of teaching resource database for higher vocational majors—Taking the teaching resource database of automotive intelligent technology major as an example[J]. Chinese Vocational and Technical Education, 2021,(29):33-39.
3. Chen Mingxuan, Feng Xueqing. Current situation and optimization strategies of digital education resource supply in China[J]. E-Education Research, 2020, 41(6): 46-52.
4. Tian Jing, Shi Weiping. Towards symbiosis: Problem reflection and reconstruction path of curriculum system of higher vocational professional groups[J]. Vocational and Technical Education, 2020, 41(20): 45-49.
5. Liu Qiao, Li Yang, Duan Hong, et al. Survey on knowledge graph construction technology[J]. Journal of Computer Research and Development, 2016(3):582-600.
6. Qi Guilin, Gao Huan, Wu Tianxing. Research progress of knowledge graph[J]. Information Engineering, 2017(1):4-25.
7. Xu Zenglin, Sheng Yongpan, He Lirong, Wang Yafang. Survey on knowledge graph technology[J]. Journal of University of Electronic Science and Technology of China, 2016(4):589-606.
8. Zhu Muyijie, Bao Bingkun. Research progress on the development and construction of knowledge graph[J]. Journal of Nanjing University of Information Science & Technology (Natural Science Edition), 2017, 9(06):575-582. DOI:10.13878/j.cnki.jnuist.2017.06.002.
9. Bilal Abu Salih,Salihah Alotaibi.A systematic literature review of knowledge graph construction and application in education.[J].Heliyon,2024,10(3):e25383-e25383.
10. Mzwri K,Szabo T M .Internet Wizard for Enhancing Open-Domain Question-Answering Chatbot Knowledge Base in Education[J].Applied Sciences,2023,13(14).
11. Hu Guangyong. Research on the construction of knowledge graph of professional knowledge and skill system[J]. Industry and Information Technology Education, 2020(12):123-127.
12. Guo Hongwei. Research on the construction of knowledge graph of online courses in colleges and universities based on intelligent education—Taking the history of Chinese medicine as an example[J]. China Educational Technology, 2021,(2):123-130.
13. He Wentao, Li Mengqing, Lu Lu, et al. The analysis approach and application value of textbook knowledge content based on knowledge modeling graph[J]. Journal of Distance Education, 2021, 39(1):104-112.