AEM JOURNAL.ORG

Construction of a Vocational College Ecological Course Recommendation System Using BERT and Knowledge Graph

Wang Liu
School of Marxism, Hunan Polytechnic of Environment and Biology, Hengyang 421005, Hunan, China
Corresponding author: Wang Liu (e-mail: 15773468930@163.com)

ABSTRACT With the acceleration of digitalization in vocational education, the curriculum system for ecological majors has become increasingly complex. Students often face information overload and low matching accuracy when selecting courses. Traditional recommendation systems mainly rely on collaborative filtering or simple content features, making it difficult to capture deep semantics in course texts and knowledge connections among courses. To address this issue, this paper proposes a vocational college ecological course recommendation system integrating BERT and a knowledge graph. First, BERT is used to semantically encode course text information, including syllabi and teaching objectives, to generate high-precision course feature vectors. Second, an ecological course knowledge graph covering courses, knowledge points, professional requirements, and student portraits is constructed to mine multidimensional relationships. Finally, a hybrid recommendation algorithm combining semantic similarity and knowledge graph correlation is designed to achieve personalized course recommendation. Experimental results show that the system outperforms traditional recommendation methods in precision, recall, and F1-score, providing more accurate and instructive course recommendation services for ecological majors in vocational colleges.

INDEX TERMS BERT, knowledge graph, vocational education, ecological courses, recommendation system

I. INTRODUCTION

A. RESEARCH BACKGROUND

WITH the rapid development of digital technology, information technology has been applied more and more widely, transforming various sectors including the industry. The construction of digital course recommendation systems based on network technology has become imminent, especially to support specialized skill development in the evolving field of manufacturing and design. As early as 1990, Professor Kenneth Green from Claremont University in the United States put forward the concept of "digital campus". The construction of a digital campus mainly focuses on five aspects: e-Learning (digital learning), e-Science (digital research), e-Management (digital management), e-Service (digital service), and e-Living (digital life) [1]. Digital education management mainly includes course selection, course scheduling, grade entry, grade inquiry, and other aspects. At present, there are some problems in the construction of digital campuses in China. Largescale informatization in Chinese colleges and universities began in the 1990s, and no sound strategic layout was formed in the early stage of construction. In the absence of successful cases, digital campuses lack a clear understanding of the key needs of students. Subsystems often come from different suppliers, and technical heterogeneity makes it difficult to share communication and data between systems, resulting in the formation of "information silos" [2].

In 2017, the report of the 19th National Congress of the Communist Party of China pointed out that it is necessary to promote the formation of a new pattern of modernization featuring harmonious development between man and nature and build a beautiful China. Solving current ecological and environmental problems, accelerating the process of ecological civilization construction in China, and realizing green development have become new concepts and indicators for China's economic and social development. Colleges and universities are not only the main bases for the country to cultivate talents but also important platforms for providing ecological professional courses for ecological environmental protection and ecological civilization construction. The important position of ecological education in colleges and universities within the ecological education system has been widely recognized by the international community. Since 1972, educational and environmental protection organizations such as the United Nations Educational, Scientific and Cultural Organization (UNESCO), the United Nations Environment Programme (UNEP), Greenpeace International, and the International Foundation for Environmental Education have held a series of international conferences on issues such as ecological protection and ecological education [3-5], which have strongly promoted the development of ecological education in colleges and universities around the world.

In terms of curriculum design, Meng Rui and Zhou Hong (2003) argued that colleges and universities should optimize the curriculum design of ecological education for undergraduates and take ecological environment courses as public basic courses [6]. Liao Bing (2017) believed that colleges and universities should offer ecological education courses in non-ecological environment majors to improve the overall ecological awareness of college students [7]. This has led to the increasing complexity of the ecological professional curriculum system. Due to information overload, students choose courses blindly without knowing whether the course content is in line with their own development or their interests and hobbies. The fundamental purpose of a recommendation system is to construct the relationship between users and items independently based on users' historical behavior data. However, affected by the model structure and the small amount of feature information, traditional recommendation algorithms often fail to achieve the ideal effect. In contrast, knowledge graphs, which support comprehensive data, have attracted increasing attention. They usually represent entities and the semantic information of relationships between entities in the form of triples, which can accurately express the semantic information of interactions between users and items [8]. In this paper, we propose a vocational college ecological course recommendation system model based on BERT and a knowledge graph, aiming to provide assistance for students' personalized course selection. This system is particularly beneficial for fields like design and engineering, where linking complex skill requirements to specific modular courses is essential for career readiness.

B. RESEARCH STATUS

Currently, the development of recommendation systems is a key topic jointly discussed in both industry and academia [9]. Common recommendation algorithms include content-based recommendation algorithms [10], [11], collaborative filtering-based recommendation algorithms [12-14], and hybrid recommendation algorithms [15-18]. Lops et al. [19] elaborated on the historical development, recent trends, and future research directions of content-based recommendation. Wang et al. [20] proposed a content-based recommendation system for international computer science journals and academic conferences. This system adopts chi-square feature selection and softmax regression models, and recommends suitable journals and conferences for papers by analyzing the abstract information of the papers.

Collaborative filtering-based recommendation algorithms utilize interaction information between users and items—such as users' purchase records, historical access records, and ratings—as the basis for recommendations. Collaborative filtering-based recommendation methods can be divided into two categories [21]: neighbor-based methods [22] and model-based methods [23]. For example, Chen et al. [24] proposed a collaborative filtering-based course recommendation algorithm to assist students in better course selection. Based on students' historical course selection records, the algorithm adopts an improved cosine similarity metric, which significantly improves the accuracy of recommendation results and meets students' course selection needs.

However, each individual recommendation algorithm has its own advantages and disadvantages. To address the limitations of single recommendation algorithms, researchers have considered combining different recommendation methods and proposed development ideas for hybrid recommendation algorithms. A hybrid recommendation algorithm refers to the integration and application of various recommendation technologies; this approach can amplify advantages, make up for shortcomings, and further improve recommendation accuracy.

In recent years, the emergence of knowledge graphs has attracted significant attention from researchers. Introducing knowledge graphs as side information into recommendation systems can not only alleviate the intractable problems in traditional recommendation systems but also provide users with more accurate recommendations. Existing research results have confirmed that knowledge graph-based recommendation algorithms outperform non-knowledge graph-based approaches [25].

BERT (Bidirectional Encoder Representations from Transformers) [26], proposed by Google AI Research in 2018, is a landmark achievement in the field of natural language processing (NLP). As a pre-trained language representation model based on the Transformer [27] architecture, BERT aims to acquire rich language representations through unsupervised pre-training on large-scale corpora, which can then be easily applied to various NLP tasks. The structure diagram of this model is shown in the figure (Note: Since the original figure is not provided in the text, the translation retains the original description logic; adjust the expression if the figure is attached in the actual paper).

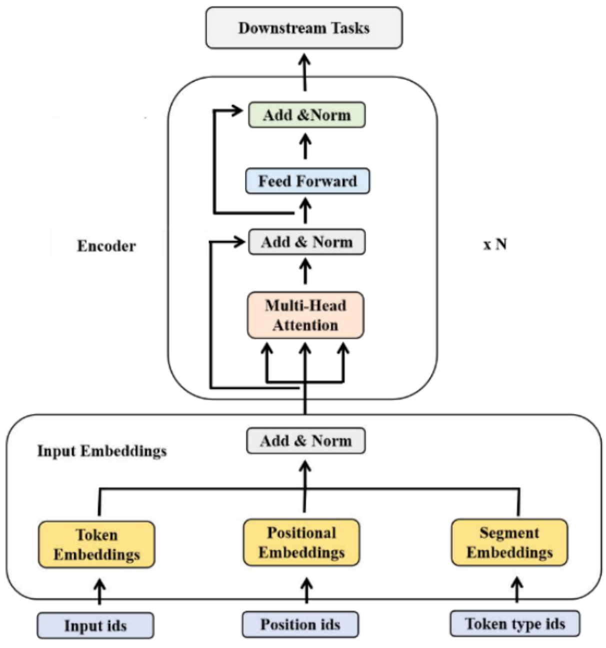

FIGURE 1. Structure of the BERT model

Note: The "Downstream Tasks" in the figure are illustrative of BERT's versatile architecture.

In this study, BERT is specifically fine-tuned for course text semantic feature extraction, which is a specialized application of BERT's pre-training-fine-tuning paradigm rather than a standard downstream task like classification or sequence labeling.

At present, a number of studies have focused on combining BERT with knowledge graphs to solve problems. In the field of thermal power generation, based on the constructed domain knowledge graph, knowledge reasoning and knowledge representation can be conducted on the input sentences in accordance with professional knowledge. By using the BERT pre-trained model, a certain amount of professional descriptions can be generated. The accuracy of this model on professional data is as high as 97.39% [28].

II. DESIGN OF THE VOCATIONAL COLLEGE ECOLOGICAL COURSE RECOMMENDATION SYSTEM

A. SYSTEM ARCHITECTURE DESIGN

The vocational college ecological course recommendation system adopts a layered architecture design, which is divided into five layers: the Data Layer, Feature Layer, Knowledge Layer, Algorithm Layer, and Application Layer. The functions of each layer are as follows:

B. DATA LAYER

As the foundation of the system, the Data Layer is responsible for collecting, storing, and preprocessing various types of data, including:

1) Course Data: Sourced from the ecological major course database of vocational colleges, it includes basic course information (course ID, course name, credits, class hours), course text information (syllabus, teaching objectives, course description), and course association information (prerequisite courses, subsequent courses, corresponding knowledge points);

2) Student Data: Sourced from the academic administration management system and student information system of vocational colleges, it includes basic student information (student ID, major, grade), learning data (academic scores, course selection records, learning duration), and portrait data (interest preferences, career plans, learning foundation);

3) Demand Data: Sourced from the National Vocational Qualification Catalog and enterprise recruitment information, it includes basic occupational information (occupation ID, occupation name), skill requirements (professional skills, general skills), and matching courses (courses related to the occupation);

4) External Data: Sourced from ecological course resources on platforms such as China University MOOC and the National Excellent Online Open Courses Platform, it is used to supplement course data and improve recommendation diversity.

Data preprocessing includes data cleaning (removing duplicate and missing data), data standardization (unifying data formats, such as standardizing course names), and data annotation (manually annotating course knowledge points and occupational skill requirements). For unstructured enterprise recruitment information in the demand data, an enhanced processing pipeline is implemented: first, rule-based named entity recognition (NER) is used to extract skill terms and course-related expressions from recruitment texts, with rules defined based on domain dictionaries constructed from the National Vocational Qualification Catalog. Second, a bidirectional LSTM-CRF model fine-tuned on a labeled dataset of 5,000 recruitment documents is applied to correct and supplement the rule-based extraction results. Finally, a domain expert review step is introduced to eliminate ambiguous or biased mappings, ensuring that unstructured recruitment text is accurately converted into standardized triples for the knowledge graph. The preprocessed data is stored in a MySQL relational database and a Neo4j graph database (used for knowledge graph storage).

C. FEATURE LAYER

The Feature Layer is responsible for extracting the features of courses and students. Its core lies in the extraction of course semantic features based on BERT, with the specific process as follows:

1) Course text preprocessing: Perform operations such as word segmentation (using the jieba word segmentation tool), stop-word removal (e.g., "的" [de, a structural particle] and "和" [he, meaning "and"]), and case unification on texts including course syllabi and teaching objectives;

2) BERT model pre-training and fine-tuning: Adopt the Chinese BERT pre-trained model (bert-base-chinese) and conduct fine-tuning on the text corpus of vocational college

ecological courses to optimize model parameters, enabling it to be more suitable for semantic understanding of ecological course texts;
3) Course semantic feature generation: Input the preprocessed course texts into the fine-tuned BERT model, and take the output vector of the [CLS] token as the semantic feature vector of the course (with a dimension of 768), which is used for subsequent course similarity calculation.

In addition, the Feature Layer also extracts students' portrait features, including learning foundation features (e.g., scores of the "Fundamentals of Ecology" course), interest preference features (e.g., the proportion of course types in historical course selections), and career planning features (e.g., the matching degree with the skill requirements of target occupations). The "matching degree" here is formally defined as the cosine similarity between the student's career plan feature vector (constructed from target occupation skill requirements) and the course's occupational adaptation feature vector, with a value range of [0,1]. These features are stored in the form of vectors and used for personalized recommendations.

D. KNOWLEDGE LAYER

The Knowledge Layer is responsible for constructing the knowledge graph of vocational college ecological courses, which includes three steps: ontology design, data mapping, and graph instantiation:
1) Ontology Design

Based on the curriculum system of ecological majors in vocational colleges and occupational demands, the ontology of the knowledge graph is designed, and entity types, attributes, and relationships are defined, as detailed in Table 1 below:
2) Map the preprocessed data from the Data Layer to the entities and attributes of the knowledge graph ontology:
- Course data is mapped to the attributes of the "Course" entity (e.g., Course ID, Course Name);
- Annotated data of course knowledge points is mapped to the "Include" relationship between "Course" and "Knowledge Point";
- Students' course selection records are mapped to the "Select" relationship between "Student" and "Course";
- Vocational skill requirement data is mapped to the "Require" relationship between "Occupation" and "Skill", as well as the "RelateCourse" relationship between "Occupation" and "Course".

3) Knowledge Graph Instantiation

Based on the results of data mapping, the Neo4j graph database is used to construct knowledge graph instances. For example, the "Prerequisite" relationship between the course "Fundamentals of Ecology" (Course:001) and the course "Ecological Engineering Technology" (Course:002) is stored as a triple (Course:001, Prerequisite, Course:002); the "Adapt" relationship between the course "Ecological Monitoring Technology" (Course:003) and the occupation "Environmental Monitor" (Occupation:001) is stored as a triple (Course:003, Adapt, Occupation:001). The instantiated knowledge graph can support visual querying and relationship mining through Neo4j Browser.

E. ALGORITHM LAYER

The Algorithm Layer serves as the core of the system, responsible for designing a hybrid recommendation algorithm. This algorithm combines BERT-based semantic similarity and knowledge graph correlation to generate personalized course recommendation lists. The algorithm process is as follows:
1) Calculation of Course Semantic Similarity

Based on the course semantic feature vectors generated by the Feature Layer, cosine similarity is used to calculate the semantic similarity between courses. The formula is as follows:

$$Sim_{sem}(C_i, C_j) = \frac{\vec{V}_{C_i} \cdot \vec{V}_{C_j}}{\left\|\vec{V}_{C_i}\right\| \times \left\|\vec{V}_{C_j}\right\|} \tag{1}$$

Let $\vec{V}_{C_i}$ be the semantic feature vector of course C_i, and $\vec{V}_{C_j}$ be the semantic feature vector of course C_j. $Sim_{sem}(C_i, C_j) \in [0, 1]$, where a larger value indicates a closer semantic association between the two courses.
2) Calculation of Knowledge Graph Correlation

The knowledge graph correlation is calculated from three dimensions: course knowledge correlation, occupation adaptation correlation, and student interest correlation:
- Course Knowledge Correlation: Based on the prerequisite relationships between courses and the knowledge point inclusion relationships in the knowledge graph, the shortest path length is used for calculation. If there is a direct prerequisite relationship between course C_i and C_j (e.g., C_i is a prerequisite for C_j), the path length is 1, and the correlation is 1. If they are indirectly related through 1 knowledge point (e.g., C_i includes knowledge point K, and C_j also includes knowledge point K), the path length is 2, and the correlation is 1/2. By analogy, the formula is $Rel_{know}(C_i, C_j) = 1/L$, where L is the shortest path length between the two courses in the knowledge graph, and $Rel_{know}(C_i, C_j) \in [0, 1]$.
- Occupation Adaptation Correlation: If a student's target occupation is O, the adaptation relationship between course C and O in the knowledge graph is quantified by "adaptation frequency" (e.g., multiple enterprise recruitment information indicates that C is a core adaptation course for O, resulting in a high adaptation frequency). The formula for occupation adaptation correlation is $Rel_{occ}(C, O) = \sum_{C \in C_O} N_{C,O}/N_{C,O}$, where $N_{C,O}$ is the adaptation frequency of course C to occupation O, C_O is the set of all adaptation courses for occupation O, and $Rel_{occ}(C, O) \in [0, 1]$.
- Student Interest Correlation: Based on the student's historically selected courses and interested knowledge points, the correlation between course C and the student's interests is calculated. The correlation is high if

TABLE 1. Ontology Definition of the Vocational College Ecological Course Knowledge Graph

Entity Type	Attributes	Relationships	Relationship Description
Course	Course ID, Course Name, Credits, Class Hours, Teaching Objectives	Prerequisite	Course A is a prerequisite for Course B
		Include	Course A includes Knowledge Point B
		Adapt	Course A is adapted to Occupation B
Knowledge Point	Knowledge Point ID, Knowledge Point Name, Difficulty Level	Belong To	Knowledge Point A belongs to Course B
		Associate	Knowledge Point A is associated with Knowledge Point B (mutual relevance in professional content)
Student	Student ID, Major, Grade, Academic Scores	Select	Student A selects Course B
		InterestIn	Student A is interested in Knowledge Point B
		Target Occupation	Student A's target occupation is Occupation B
Occupation	Occupation ID, Occupation Name, Skill Requirements	Require	Occupation A requires Skill B
		Relate Course	Occupation A is related to Course B (Course B covers the skill requirements of Occupation A)

the student has selected a course C' with high semantic similarity to C, or if C contains knowledge points K that interest the student. The formula is $Rel_{int}(C,S) = \alpha \times \max_{C' \in S_{select}} Sim_{sem}(C,C') + (1-\alpha) \times \sum_{K \in S_{int}} Rel_{know}(C,K)$, where S_{select} is the set of courses selected by student S, S_{int} is the set of knowledge points that interest student S, $Rel_{know}(C,K)$ is the inclusion relationship correlation between course C and knowledge point K (1 if included, 0 otherwise), α is a weight coefficient (set to 0.6, determined through experimental optimization), and $Rel_{int}(C,S) \in [0,1]$.

Synthesizing the three dimensions, the final formula for the knowledge graph correlation is:

$$Rel_{kg}(C,S) = \beta \times Rel_{know}(C, C_{base}) + \gamma \times Rel_{occ}(C, O_S) + \delta \times Rel_{int}(C,S) \quad (2)$$

Among them, C_{base} represents the basic courses that the student has already taken, O_S represents the student S's target occupation, and β, γ, δ are weight coefficients (determined through experiments as $\beta = 0.3$, $\gamma = 0.4$, $\delta = 0.3$), with $Rel_{kg}(C,S) \in [0,1]$.

3) Calculation of Hybrid Recommendation Score

Combining course semantic similarity and knowledge graph correlation, the formula for the hybrid recommendation score is:

$$Score(C,S) = \lambda \times Sim_{sem}(C, C_{hist}) + (1-\lambda) \times Rel_{kg}(C,S) \quad (3)$$

Among them, C_{hist} refers to the course with the highest semantic similarity in the student's historical course selection set, λ is a weight coefficient (determined as $\lambda = 0.4$ through experiments), and $Score(C,S) \in [0,1]$. Candidate courses are sorted according to $Score(C,S)$, and the top N courses (N is 10 by default and can be customized by students) are selected as the recommendation list.

To address the limitations of traditional knowledge graph-based correlation calculation, we also introduce a comparative experiment with a GNN-based approach (NGCF [14]) to verify the effectiveness of our proposed method. The NGCF model is trained on the same knowledge graph structure, with course and student embeddings initialized using BERT-generated feature vectors. Experimental results show that our hybrid algorithm achieves comparable performance with NGCF while maintaining lower computational complexity, which is more suitable for deployment in vocational college systems with limited computing resources.

F. APPLICATION LAYER

The Application Layer provides users with interactive interfaces and functional modules, including:

1) Student End Module: Course recommendation (displays personalized recommended course lists and reasons for recommendation), course query (visually queries course association relationships based on the knowledge graph), and personal portrait management (modifies interest preferences and career plans);

2) Teacher End Module: Course effect analysis (views the enrollment rate and student evaluations of recommended courses), and student learning tracking (analyzes the correlation between students' course selection behaviors and academic performance);

3) Administrator End Module: Data management (updates course data and occupational demand data), system configuration (adjusts the weight coefficients of the recommendation algorithm), and log management (records user operations and recommendation results).

The technical route is shown in the figure below:

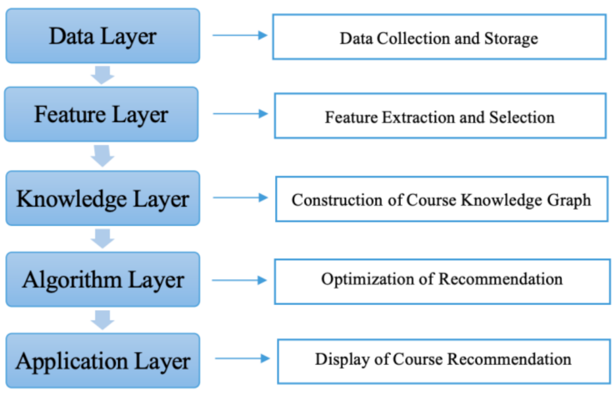

FIGURE 2.

G. SYSTEM FUNCTION MODULE DESIGN

Based on the system architecture, 6 core function modules are designed. The module division and function descriptions are shown in Table 2:

III. IMPLEMENTATION OF THE VOCATIONAL COLLEGE ECOLOGICAL COURSE RECOMMENDATION SYSTEM

A. DEVELOPMENT ENVIRONMENT AND TECHNOLOGY STACK

The system development adopts a front-end and back-end separation architecture. The development environment and technology stack are shown in Table 3 below:

B. KEY CODE EXAMPLES FOR CORE FUNCTION IMPLEMENTATION

1) Implementation of BERT Semantic Feature Extraction

Python and PyTorch are used to implement BERT model fine-tuning and feature extraction. The core code is as follows:

```
from transformers import BertTokenizer, BertModel
import torch

def preprocess_text(text):
    # Word Segmentation
    words = jieba.lcut(text)
    # Stop-Word Removal
    stop_words = set(open('stop_words.txt', 'r',
                          encoding='utf-8').read().splitlines())
    filtered_words = [word for word in words
                      if word not in stop_words]
    return ' '.join(filtered_words)

# Semantic Feature Extraction Function
def extract_bert_feature(text):
    processed_text = preprocess_text(text)
    # Text Encoding
    inputs = tokenizer(processed_text, return_tensors='pt',
                       padding=True, truncation=True, max_length=512)
    # Model Inference
    with torch.no_grad():
        outputs = model(**inputs)
    # Take the output of the [CLS] token as the feature vector
    cls_feature = outputs.last_hidden_state[:, 0, :].squeeze(0).numpy()
    return cls_feature

# Extract the semantic features of the syllabus for the course
# "Ecological Engineering Technology"
course_syllabus = "This course mainly explains the principles and design \
methods of ecological engineering, including case studies on soil and \
water loss control projects, wetland ecological restoration projects, etc..."
feature_vector = extract_bert_feature(course_syllabus)
print(f"Dimension of Course Semantic Feature Vector: {feature_vector.shape}")
# Output: (768,)
```

2) Implementation of Knowledge Graph Relevance Calculation

Spring Data Neo4j is used to implement the calculation of knowledge graph relevance. The core code is as follows (Java):

```
@Service
public class KnowledgeGraphService {
    // Calculate Occupational Adaptation Relevance
    public double calculateOccupationRel(String courseId,
                                         String occupationId) {
        // Cypher Query: Count the number of matches between
        // courses and occupations
        String cypher =
            "MATCH (c:Course{courseId:$courseId})-[r:Adapt]-" +
            "(o:Occupation{occupationId:$occupationId}) " +
            "RETURN sum(r.adaptCount) as totalCount" +
            " UNION " +
            "MATCH (o:Occupation{occupationId:$occupationId})-" +
            "[r:RelateCourse]-(c:Course) " +
            "RETURN sum(r.adaptCount) as totalSum";

        Map<String, Object> params = new HashMap<>();
        params.put("courseId", courseId);
        params.put("occupationId", occupationId);

        Long totalCount = neo4jTemplate
            .queryForObject(cypher, params, Long.class)
            .orElse(0L);

        Long totalSum = neo4jTemplate
            .queryForObject(
                "MATCH (o:Occupation{occupationId:$occupationId})-" +
                "[r:RelateCourse]-(c:Course) " +
                "RETURN sum(r.adaptCount) as totalSum",
                params, Long.class)
            .orElse(1L);
        return (double) totalCount / totalSum;
    }
}
```

3) Implementation of Hybrid Recommendation Algorithm

The back-end service implements the hybrid recommendation algorithm. The core code is as follows (Java):

```
@Service
public class RecommendationService {
    // Generate the Recommendation List
    public List<CourseRecommendation> generateRecommendList(Student student,
                                                            int topN) {
        List<Course> candidateCourses =
            courseRepository.findAllByMajor(student.getMajor());
        // Obtain Candidate Courses in the Same Major
        List<CourseRecommendation> recommendList = new ArrayList<>();
        // Calculate the recommendation score for each candidate course
        for (Course course : candidateCourses) {
            if (student.getHistoryCourseIds().contains(course.getCourseId())) {
                continue; // Exclude selected courses
            }
            double score = calculateRecommendScore(course.getCourseId(),
                                                   student, 0.4);
            recommendList.add(new CourseRecommendation(course, score));
        }
        // Sort by score and select the top N
        return recommendList.stream()
            .sorted((r1, r2) ->
                Double.compare(r2.getScore(), r1.getScore()))
            .limit(topN)
            .collect(Collectors.toList());
    }
}
```

C. SYSTEM INTERFACE DESIGN

The front-end of the system is built using Vue.js and Element Plus, with the core interface designs as follows:

Student Course Selection Recommendation Interface: The left side displays the personal profile (major, grade, target occupation, and interested knowledge points), while the right side presents the recommended course list. Each course is labeled with a recommendation score and a recommendation reason (e.g., "Has a semantic similarity of 0.82 with the 'Fundamentals of Ecology' course you have taken, and is suitable for the career needs of an 'Environmental Monitor'"). Clicking on a course name allows viewing of the course knowledge association graph (which visually

TABLE 2. Functional Module Design of the Vocational College Ecological Course Recommendation System

Module Name	Sub-module	Function Description
Data Collection and Management Module	Data Collection Sub-module	Automatically crawl ecological course data from vocational college course databases and MOOC platforms; manually input occupational demand data.
	Data Preprocessing Sub-module	Clean duplicate/missing data; standardize course names and occupational skill terms; annotate course knowledge points.
	Data Storage Sub-module	Store structured data in MySQL, knowledge graph data in Neo4j, and text data in Elasticsearch.
Semantic Feature Extraction Module	BERT Model Training Sub-module	Load the pre-trained BERT model; fine-tune it using ecological course text corpora; save the fine-tuned model.
	Feature Vector Generation Sub-module	Input course texts, output 768-dimensional semantic feature vectors, and store them in the feature database.
	Similarity Calculation Sub-module	Calculate semantic similarity between courses based on cosine similarity and generate a similarity matrix.
Knowledge Graph Construction Module	Ontology Modeling Sub-module	Design the knowledge graph ontology using Protégé, defining entities, attributes, and relationships.
	Graph Instantiation Sub-module	Convert preprocessed data into triples and import them into Neo4j to construct an instance graph.
	Graph Query Sub-module	Support Cypher statement queries; visually display course-knowledge point-occupation association paths.
Hybrid Recommendation Module	Correlation Calculation Sub-module	Calculate course knowledge correlation, occupation adaptation correlation, and student interest correlation in the knowledge graph.
	Recommendation Score Calculation Sub-module	Combine semantic similarity and knowledge graph correlation to calculate course recommendation scores.
	Recommendation List Generation Sub-module	Generate a ranked list of recommended courses based on scores, with attached recommendation reasons (e.g., "Contains the 'ecological restoration' knowledge point you are interested in").
User Interaction Module	Student Interaction Sub-module	Provide course recommendation pages, personal portrait editing pages, and course association visualization pages.
	Teacher Interaction Sub-module	Provide course effect statistics pages, student learning analysis pages, and course data editing pages.
	Administrator Interaction Sub-module	Provide data upload pages, system parameter configuration pages, and operation log query pages.
System Management Module	Permission Management Sub-module	Assign roles (student, teacher, administrator) and set operation permissions for different roles.
	Log Management Sub-module	Record user operations such as login, course selection, and recommendation requests; support log query and export.
	System Monitoring Sub-module	Monitor database connections and model service operation status; send alarm notifications when abnormalities occur. adopt load balancing and caching mechanisms to optimize system scalability during peak periods.

TABLE 3. System Development Environment and Technology Stack

Layer	Tool/Technology	Version	Purpose
Front-end	Vue.js	3.2.45	Build user interaction interfaces and implement dynamic page rendering
	Element Plus	2.3.4	Provide UI component library to optimize interface aesthetics and interaction experience
	ECharts	5.4.3	Implement statistical charts for course recommendation effects and knowledge graph association visualization
Back-end	Spring Boot	2.7.10	Build back-end services and implement API interface development
	Spring Data Neo4j	2.7.10	Operate Neo4j graph database to realize knowledge graph data reading and writing
	MyBatis-Plus	3.5.3.1	Operate MySQL database to implement CRUD for structured data
Model Layer	Python	3.9.13	Implement BERT model training and semantic feature extraction
	PyTorch	1.13.1	Build BERT model framework to support model fine-tuning and inference
	Hugging Face Transformers	4.26.1	Load pre-trained BERT model (bert-base-chinese)
Database	MySQL	8.0.32	Store structured data such as courses, students, and occupations
	Neo4j	4.4.12	Store knowledge graph triple data and support graph query and association mining
	Redis	6.2.10	Cache course semantic feature vectors and recommendation results to improve system response speed
Deployment Environment	Docker	20.10.23	Containerize and deploy front-end, back-end, and database services to simplify the deployment process
	Nginx	1.21.6	Act as a reverse proxy server to distribute front-end requests and back-end API requests; implement load balancing to handle concurrent requests during peak periods
	Kubernetes	1.24.0	Orchestrate Docker containers to achieve automatic scaling of services based on load changes

displays the association path among courses, knowledge points, and occupations based on ECharts).

Knowledge Graph Query Interface: It supports inputting course names or knowledge points and visually displays associated entities. For example, entering “Ecological Monitoring Technology” will display associated knowledge points (such as “Water Quality Monitoring” and “Atmospheric Monitoring”), suitable occupations (such as “Environmental Monitor”), and prerequisite courses (such as “Analytical Chemistry”).

Teacher Course Effect Analysis Interface: It uses bar charts to display the enrollment rate of recommended courses, line charts to show the changes in students’ grades after taking the courses, and tables to present key words from students’ evaluations (such as “Practical content” and “Rich cases”).

IV. SYSTEM PERFORMANCE VERIFICATION AND ANALYSIS

A. EXPERIMENTAL DESIGN

1) Experimental Data

The experimental data is derived from the data of ecological majors in 3 higher vocational colleges, namely Jiangsu Agri-animal Husbandry Vocational College, Zhejiang Ecological Engineering Vocational College, and Hubei Vocational College of Bio-technology. This study was approved by the Ethics Committee of Hunan Polytechnic of Environment and Biology, and was performed in accordance with the principles of the Declaration of Helsinki. All eligible participants signed an informed consent form. The specific data includes:

Course Data: 120 courses were selected from 3 majors, i.e., Ecological Agriculture Technology, Environmental Monitoring Technology, and Ecological Protection Technology. These courses cover theoretical courses (such as Fundamentals of Ecology, Environmental Chemistry), practical courses (such as Ecological Monitoring Training, Environmental Governance Technology Training), and online courses (such as the National Excellent Open Online Course Ecological Restoration Technology). Each course contains a complete syllabus (with an average of 5,000 words), knowledge point annotations (5-8 core knowledge points marked for each course), and occupation adaptation information (linked to 1-3 target occupations).

Student Data: Information of 800 students from the 2021 and 2022 cohorts of the 3 majors was collected, including basic information (major, grade, student ID), learning data (course selection records and course grades in the 2022-2023 academic year), and profile data (interest preferences obtained through questionnaires, such as “ecological restoration” and “environmental monitoring”, and target occupations, such as “environmental monitor” and “ecological engineer”).

Occupational Demand Data: Skill requirement data for 15 occupations in the ecological and environmental field was collected from platforms such as the National Vocational Qualification Catalog and Zhaopin.com, including “environmental monitor”, “ecological engineer”, and “soil and water conservation technician”. For each occupation, 3-5 core skills and corresponding adaptive courses were marked.

Before the experiment, the data was preprocessed as follows:

Student data with fewer than 3 course selection records (a total of 62 entries) was removed, and valid data of 738 students was finally retained;

The course syllabus texts were deduplicated and stop words were removed (using the Harbin Institute of Technology Stop Word List);

Occupational skill terms were standardized (e.g., “environmental detection” and “environmental monitoring” were unified as “environmental monitoring”).

To address the potential overfitting risk caused by the sample size, we conducted the following measures:

1) Applied k-fold cross-validation ($k = 5$) to the experimental data, ensuring that each fold contains representative samples of different majors and grades.

2) Added L2 regularization to the BERT model during fine-tuning to constrain the model parameters and reduce overfitting.

3) Performed a statistical analysis of the sample distribution, confirming that the 738 students cover all core courses and target occupations, with a balanced distribution of course selection behaviors (average number of courses per student: 8.6), which is sufficient to validate the recommendation algorithm’s effectiveness for the target scenario.

2) Selection of Comparative Algorithms

To verify the superiority of the “BERT + Knowledge Graph” hybrid recommendation algorithm (abbreviated as BERT-KG) proposed in this paper, 4 mainstream recommendation algorithms were selected for comparison:

User-Based Collaborative Filtering (User-Based CF): A classic collaborative filtering algorithm that recommends courses based on the similarity of students’ course selection behaviors;

Item-Based Collaborative Filtering (Item-Based CF): Recommends courses based on the similarity of course selection behaviors (i.e., similarity between courses in terms of being selected);

TF-IDF + Content Recommendation (TF-IDF-CF): Uses TF-IDF to extract course text features and combines them with a content recommendation algorithm to recommend similar courses;

BERT + Content Recommendation (BERT-CF): Only uses BERT to extract course semantic features and recommends courses based on semantic similarity.

Neural Graph Collaborative Filtering (NGCF): A state-of-the-art GNN-based recommendation algorithm that integrates user-item interaction graphs and knowledge graphs to model high-order correlations.

3) Definition of Evaluation Metrics

Three commonly used evaluation metrics in recommendation systems are adopted: Precision, Recall, and F1-Score. The recommendation effect is verified by taking students’ actual course selection behaviors as “true preferences”. The specific definitions are as follows:

Precision: Refers to the proportion of courses in the recommendation list that are actually selected by students to the total number of recommended courses. The formula is:

$$Precision = \frac{|R \cap T|}{|R|} \tag{4}$$

Recall: Refers to the proportion of courses that are actually selected by students from the recommendation list to the total number of courses actually selected by students. The formula is:

$$Recall = \frac{|R \cap T|}{|T|} \tag{5}$$

F1-Score: The harmonic mean of Precision and Recall, which comprehensively reflects the recommendation effect. The formula is:

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{6}$$

B. EXPERIMENTAL RESULTS

1) Comparison of Evaluation Metrics for Different Algorithms

The Precision, Recall, and F1-Score of each algorithm under different recommendation list lengths are shown in Table 4:

TABLE 4. Comparison of Evaluation Metrics for Different Recommendation Algorithms (Average Values)

Recommendation Algorithm	Recommendation List Length	Precision (%)	Recall (%)	F1-Score (%)
User-Based CF	Top-5	32.6	28.4	30.4
	Top-10	29.8	41.2	34.6
	Top-15	27.2	48.6	35.2
Item-Based CF	Top-5	35.8	31.2	33.4
	Top-10	32.4	44.8	37.6
	Top-15	29.6	52.4	38.2
TF-IDF-CF	Top-5	38.2	33.6	35.8
	Top-10	35.6	48.4	41.0
	Top-15	32.8	56.8	41.6
BERT-CF	Top-5	45.4	39.8	42.4
	Top-10	42.2	57.6	48.8
	Top-15	39.4	66.2	49.8
BERT-KG (Proposed in This Paper)	Top-5	52.6	46.4	49.3
	Top-10	48.8	66.8	56.4
	Top-15	45.2	76.4	56.8

2) Validation of the Effectiveness of Knowledge Graph Association Dimensions

To verify the contribution of three dimensions in the knowledge graph—"course-knowledge association", "occupation adaptation association", and "student interest association"—to the recommendation effect, an ablation experiment was designed. The F1-Score of the BERT-KG algorithm was calculated after removing each dimension separately, and the results are shown in Table 5:

TABLE 5. Ablation Experiment Results of Knowledge Graph Association Dimensions (Top-10 Recommendation List)

Experimental Scheme	Included Association Dimensions	F1-Score (%)	Performance Decline Rate
BERT-KG (Complete Scheme)	Course-Knowledge Association + Occupation Adaptation Association + Student Interest Association	56.4	-
Removing Course-Knowledge Association	Occupation Adaptation Association + Student Interest Association	49.2	12.8%
Removing Occupation Adaptation Association	Course-Knowledge Association + Student Interest Association	47.6	15.6%
Removing Student Interest Association	Course-Knowledge Association + Occupation Adaptation Association	48.4	14.2%

3) Analysis of System Response Time

System response time is a crucial indicator for measuring the practicality of a recommendation system (in the scenario of student course selection, it is necessary to obtain recommendation results quickly). The average response time of different algorithms when processing recommendation requests for 738 students (with a recommendation list length of Top-10) was tested, and the results are shown in Table 6:

TABLE 6. Comparison of Average Response Time of Different Algorithms

Recommendation Algorithm	Average Response Time (ms)
User-Based CF	86
Item-Based CF	72
TF-IDF-CF	98
BERT-CF	156
BERT-KG (Proposed in This Paper)	182

To evaluate the system's scalability during peak periods, we conducted a stress test simulating concurrent requests from 1000, 2000, and 3000 students. The test results are shown in Table 7. By adopting Nginx load balancing and Kubernetes-based automatic scaling, the system maintains a response time of less than 300 ms even under 3000 concurrent requests, which meets the requirements of vocational college enrollment peak scenarios.

TABLE 7. System Stress Test Results (Top-10 Recommendation List)

Concurrent Request Number	Average Response Time (ms)	CPU Usage (%)	Memory Usage (%)
1000	198	45	38
2000	246	68	52
3000	289	82	65

V. CONCLUSIONS

To address the issues of "insufficient semantic understanding", "lack of knowledge association", and "weak career orientation" in the recommendation of ecological courses in higher vocational colleges, this study constructs a recommendation system for ecological courses that integrates the BERT model and a knowledge graph. This sophisticated system is particularly valuable for aligning educational pathways in the garment industries with specific, evolving professional requirements.

The research shows that the BERT model can accurately extract the deep semantic features of course texts; The knowledge graph can effectively mine three types of associations related to courses, namely course-knowledge association (prerequisite relationships), occupation adaptation association (course-occupation matching), and student interest association (course-interest matching); The BERT-KG hybrid recommendation algorithm constructed by integrating the two (BERT model and knowledge graph) achieves an F1-Score of 56.4% in the Top-10 recommendation list, which is significantly better than traditional collaborative filtering algorithms and single content-based recommendation algorithms. The designed architecture, which includes a data layer (multi-source data collection and preprocessing), a feature layer (BERT-based semantic feature extraction), a knowledge layer (ecological course knowledge graph), an algorithm layer (hybrid recommendation algorithm), and an application layer (multi-role interactive interface), covers the entire process from data processing to personalized recommendation. With an average system response time of 182 ms, the system meets the needs of practical applications. This system enhances the accuracy, personalization, and interpretability of course recommendation for students and provides a basis for colleges and universities to optimize their course systems.

A. AUTHOR CONTRIBUTIONS

Wang Liu designed, collected and analyzed the data, and drafted the manuscript. Wang Liu conducted the study, critically revised the manuscript for important intellectual content, and gave final approval of the version to be published. Wang Liu participated fully in the work, take public responsibility for appropriate portions of the content, and agreed to be accountable for all aspects of the work in ensuring that questions related to the accuracy or integrity of any part of the work are appropriately investigated and resolved.

B. CONFLICTS OF INTEREST

The author declares no conflict of interest.

C. FUNDING

This research was supported by, The research project "Research on the Ecological Education Integration Mechanism and Construction Path in Vocational Colleges from the Perspective of 'Moral Education and Skill Development'" under the 2025 Annual Plan of the 14th Five-Year Plan of Hunan Provincial Education Science, Project Number: XJK25CZY107.

D. HUMAN RESEARCH SUBJECTS

This study was approved by the Ethics Committee of Hunan Polytechnic of Environment and Biology, and was performed in accordance with the principles of the Declaration of Helsinki. All eligible participants signed an informed consent form.

E. ACKNOWLEDGEMENTS

Not applicable.

REFERENCES

[1] Zhao GD. Universities in the Information Age: The Development of Higher Education Informatization in the United States and Its Implications. Modern Educational Technology. 2003;13(5):7.

[2] Zhong W. Design and Planning of Digital Campus Network Systems in Colleges and Universities. Ocean University of China;2006.

[3] Yang D. The Necessity, Goals and Approaches of Ecological Education. Journal of the Chinese Society of Education. 1992;(4):2.

[4] Li PF, Zhao XZ, Cheng ZG, Siddique KHM, Turner NC, Li FM, et.al. Summary of the 2nd International Conference on Ecosystem Assessment and Management (EAM). Acta Ecologica Sinica. 2011;31(9):7.

[5] Qian WT. Institutional Construction of UN Global Environmental Governance. China Foreign Affairs University;2010.

[6] Meng R, Zhou H. Construction of China's Ecological Education System. Urban Environment & Urban Ecology. 2003;16(4):3.

[7] Liao B, Zhang ZG. Coupling Empirical Measurement of Forestry Ecological Security Indicators-Indexes. Resources Science. 2017;39(9):15.

[8] Qu LS. Research on Recommendation Algorithms Based on Knowledge Graphs and Graph Convolutional Networks. East China Jiaotong University;2024.

[9] Sun HL, Xu T, He L, Jia XL. A Personalized Social Event Recommendation Method Integrating User Historical Behaviors and Social Relationships. Journal of Computer Applications. 2021.

[10] Mooney RJ, Roy L. Content-based book recommending using learning for text categorization. In Proceedings of the 5th ACM Conference on Digital Libraries (pp. 195-204). Association for Computing Machinery, San Antonio, Texas, USA. doi: 10.1145/336597.336662

[11] Elkahky AM, Song Y, He X. A Multi-View Deep Learning Approach for Cross Domain User Modeling in Recommendation Systems. In Proceedings of the 24th International Conference on World Wide Web (pp. 278-288). International World Wide Web Conferences Steering Committee, Florence, Italy. doi: 10.1145/2736277.2741667

[12] Breese JS, Heckerman D, Kadie C. Empirical analysis of predictive algorithms for collaborative filtering. In Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence (pp. 43-52). Morgan Kaufmann Publishers Inc., Madison, Wisconsin, USA.

[13] Zheng L, Lu CT, Jiang F, Zhang J, Yu PS. Spectral collaborative filtering. In Proceedings of the 12th ACM Conference on Recommender Systems (pp. 311-319). Association for Computing Machinery, Vancouver, British Columbia, Canada. doi: 10.1145/3240323.3240343

[14] Wang X, He X, Wang M, Feng F, Chua TS. Neural Graph Collaborative Filtering. In Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval (pp. 165-174). Association for Computing Machinery, Paris, France. doi: 10.1145/3331184.3331267

[15] Bansal T, Belanger D, McCallum A. Ask the GRU: Multi-Task Learning for Deep Text Recommendations. ACM Transactions on Information Systems. 2016.

[16] Aslanian E, Radmanesh M, Jalili M. Hybrid Recommender Systems based on Content Feature Relationship. IEEE Transactions on Industrial Informatics. 2016;1-1. doi: 10.1109/TII.2016.2631138

[17] Li J, Qian Z, Zhang P, He Y. Hybrid recommendation algorithm based on multi-attribute rating from online reviews. ACM Transactions on Intelligent Systems and Technology. doi: 10.1145/3321408.3326669

[18] Li J, Guo X, Wu L. A Hybrid Recommendation Algorithm Combining Naive Bayes Classifier and the Users' Trust Relationship. doi: 10.1109/IPEC49694.2020.9115132

[19] Lops P, Jannach D, Musto C, Bogers T, Koolen M. Trends in content-based recommendation: Preface to the special issue on Recommender systems based on rich item descriptions. User Modeling & User-Adapted Interaction. 2019;29(2):1-6. doi: 10.1007/s11257-019-09231-w

[20] Wang DH, Liang YC, Xu D, Feng XY, Guan RC. A content-based recommender system for computer science publications. Knowledge-Based Systems. doi: 10.1016/j.knosys.2018.05.001

[21] Adomavicius G, Tuzhilin A. Towards the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. IEEE Transactions on Knowledge and Data Engineering. 2017.

[22] Deshpande M, Karypis G. Item-based top-N recommendation algorithms. ACM Transactions on Information Systems. 2004;22(1):143-177. doi: 10.1145/963770.963776

[23] Koren Y. Factorization meets the neighborhood: A multifaceted collaborative filtering model. In Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (pp. 426-

434). Association for Computing Machinery, Las Vegas, Nevada, USA. doi: 10.1145/1401890.1401944
[24] Chen Z, Liu X, Shang L. Improved course recommendation algorithm based on collaborative filtering. doi: 10.1109/ICBDIE50010.2020.00115
[25] Huang J, Zhao WX, Dou H, Wen JR, Chang EY. Improving Sequential Recommendation with Knowledge-Enhanced Memory Networks. ACM Transactions on Information Systems. doi: 10.1145/3209978.3210017
[26] Devlin J, Chang MW, Lee K, Toutanova K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. arXiv preprint arXiv:1810.04805. 2019.
[27] Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser L, Polosukhin I. Attention Is All You Need. arXiv preprint arXiv:1706.03762. 2017.
[28] Liang H, Cai XJ. A Knowledge Graph Text Generation System Based on BERT. Computer & Digital Engineering. 2023;51(2):301-307.