Research on Personalized Learning Path Recommendation System for Higher Education Based on Deep Learning

Yili Fang
[1]Department of Physical, Aesthetic & Labor Education, Zhejiang Shuren University, HangZhou 310000, Zhejiang, China
Corresponding author: Yili Fang (e-mail: Fangyili19951010@126.com).

ABSTRACT The traditional higher education teaching model often adopts a standardized teaching form of "one size fits all", which is difficult to adapt to the knowledge base, learning ability, interest preferences, and learning pace of different students, resulting in significant differences in learning efficiency and difficulty in implementing personalized teaching. With the deep development of smart education and artificial intelligence technology, deep learning provides a new technological path for building personalized learning services in higher education scenarios with its powerful feature extraction, sequence modeling, and user behavior mining capabilities.

INDEX TERMS Deep learning; Higher Education; Personalized learning; Recommended learning path

I. INTRODUCTION

A. RESEARCH SIGNIFICANCE

From the perspective of individual students, higher education students come from a wide range of sources, and there are significant differences in their knowledge reserves, subject foundations, study habits, cognitive abilities, time planning, and interest directions before enrollment. In the same professional course, some students have solid prerequisite knowledge, can quickly understand classroom content and carry out extended learning; Some students have weak prior knowledge, making it difficult for them to keep up with the normal teaching progress. Long term accumulation can lead to problems such as disinterest in learning and failing courses; Some students tend to focus on practical application and have a low acceptance of theoretical courses, and standardized learning paths cannot match their development needs. The unified learning arrangement erases the personalized characteristics of students, which cannot enable students with spare capacity to achieve higher education and expansion, nor can it provide tutoring for students with weak foundations, seriously restricting the improvement of the quality of talent cultivation in higher education.

B. CURRENT RESEARCH STATUS AT HOME AND ABROAD

The research on smart education and personalized learning started earlier in foreign countries, and developed countries in Europe and America have formed a large number of research results in the fields of learning recommendation and learning path planning based on well-established online education platforms and artificial intelligence technology. At the application level of recommendation algorithms, early foreign education recommendation systems mainly used traditional collaborative filtering algorithms and content-based recommendation algorithms. Collaborative filtering relies on user similarity and item similarity to achieve resource recommendation, and has been widely used in early MOOC platforms. However, this algorithm has serious problems with data sparsity and cold start. Content based recommendation algorithms rely on course tags and knowledge point attributes for recommendation, with strong stability, but cannot explore users' deep behavioral preferences. With the rise of deep learning technology, foreign scholars have taken the lead in introducing deep models into the field of educational recommendation. Google、Microsoft and other technology companies have partnered with overseas universities to mine students' static features (age, major, grades) using deep neural networks (DNN), and model students' temporal learning behavior using long short-term memory networks (LSTM) to achieve dynamic learning resource recommendations. Part of the research combines Graph Convolutional Networks (GCN) to construct a course knowledge graph, mining the predecessor, successor, and association relationships between knowledge points, so that recommended content conforms to the logical order of knowledge learning, rather than simply interest push.

The research on personalized learning and intelligent recommendation in China has developed rapidly with the process of education informatization. In recent years, under the promotion of national policies, the research results of in-depth learning combined with intelligent education have increased year by year. The research subjects are mainly universities, educational technology institutions and Internet education enterprises.

II. RELATED THEORETICAL AND TECHNICAL FOUNDATIONS

As the theoretical and technical support chapter of the entire text, this chapter first defines the core professional concepts involved in the research, and then systematically explains the relevant basic theories and key technologies from five dimensions: educational theory, recommendation system theory, deep learning algorithms, educational big data technology, and software development technology, providing a basis for subsequent system design, data modeling, algorithm development, and code implementation.

A. DEFINITION OF CORE CONCEPTS

The personalized learning in higher education studied in this article refers to the use of artificial intelligence technology to identify individual differences among students within the framework of the professional teaching system in universities, providing differentiated and customized learning content, learning sequence, learning pace, and learning services for different students. While ensuring the integrity of the professional knowledge system, it maximizes the adaptation of students' cognitive abilities and development needs.

A learning path is an ordered learning sequence followed by learners to achieve established learning goals, consisting of knowledge points, course resources, exercise tasks, and learning stages arranged in a certain logical combination. The personalized learning path in higher education is not the push of a single course or resource, but a complete and phased learning process. From the perspective of constituent elements, the learning path includes four core elements: firstly, knowledge nodes, namely core learning content such as courses, knowledge points, and experimental projects; The second is the logical relationship, which refers to the relationships between knowledge nodes such as predecessor, successor, parallel, and extension; The third is the learning process, which includes previewing, attending classes, practicing, reviewing, answering questions, expanding, and other complete learning processes; The fourth is the temporal rule, which includes time constraints such as learning duration, learning order, and stage assessment nodes.

B. BASIC THEORIES RELATED TO EDUCATION

Educational theory is the fundamental basis for designing learning paths, determining the structure, sequence, and stage settings of learning paths. This article mainly relies on four educational theories: constructivist learning theory, cognitive load theory, mastery learning theory, and cognitive development theory to conduct research.

Constructivist learning theory holds that learning is not a passive process of receiving knowledge, but rather a process in which learners actively construct a knowledge system based on their existing knowledge reserves. Different learners have different existing knowledge structures, so their understanding and absorption efficiency of new knowledge also vary. This theory provides core theoretical support for personalized learning: firstly, the design of learning paths must fully consider students' existing knowledge foundation and avoid the occurrence of "knowledge gaps"; Secondly, the learning process should include layered tasks and interactive elements to guide students to actively explore knowledge, rather than simply pushing content in one direction; Thirdly, allow students to independently adjust their learning pace and order, and unleash their subjective initiative. In path planning, this article combines constructivist theory to prioritize matching students' existing knowledge levels and gradually introduce new knowledge.

The cognitive load theory was proposed by psychologist Sverer, which states that the working memory capacity of the human brain is limited, and external information during the learning process can generate cognitive load. When the load exceeds the carrying range, learning efficiency will significantly decrease. Cognitive load is divided into intrinsic cognitive load, extrinsic cognitive load, and associative cognitive load. The internal load is determined by the complexity of the knowledge itself, while the external load is determined by the presentation form and learning order of the learning content.

Based on this theory, this article follows two principles when designing the learning path: firstly, breaking down high complexity knowledge points into multiple low difficulty sub nodes, pushing them step by step, and reducing the internal cognitive load; The second is to optimize the sorting and presentation format of learning content, plan paths in order from easy to difficult and from shallow to deep, reduce ineffective external cognitive load, and improve learning efficiency.

C. BASIC THEORY OF RECOMMENDATION SYSTEM

A recommendation system is an information service system that automatically pushes interesting or valuable content to users based on their needs, historical behavior, and item attributes. The learning path recommendation studied in this article belongs to the category of domain specific recommendation systems. This section introduces the three basic contents of recommendation systems: classification, core evaluation indicators, and traditional recommendation algorithms. Please refer to Table 1 for details.

TABLE 1. Comparison of core performance indicators of different recommendation algorithms

Algorithm	Precision@10	Recall@10	F1-Score@10
Traditional Collaborative Filtering	0.621	0.583	0.601
Single Deep Neural Network (DNN)	0.715	0.682	0.698
The Proposed Hybrid Deep Learning Model	0.826	0.803	0.814

According to the recommendation objectives and application scenarios, recommendation systems can be divided into

three categories. In this article, the system integrates three types of recommendation logic:
1. Content based recommendation: Recommendations are made based on the attributes, tags, and features of the item itself. In this study, students were matched with learning resources that matched the content based on attributes such as course knowledge points, difficulty, type, and major. The advantage was that there was no cold start problem, but the disadvantage was that the recommendation results were highly homogenized.
2. Collaborative filtering recommendation: divided into user based collaborative filtering and item based collaborative filtering. Based on user collaborative filtering, find students with similar interests and behaviors, and push learning content for similar students; Collaborative filtering based on items is used to find course resources with high similarity and achieve associated push notifications. This algorithm mines user behavior preferences, but is greatly affected by data sparsity.
3. Sequence recommendation: Recommend based on time-series behavior data to predict the user's next behavior and choices. In this study, sequence recommendation is used to model students' continuous learning behavior and plan a gradual learning path, which is the core of dynamic path generation.

III. SYSTEM REQUIREMENT ANALYSIS AND OVERALL ARCHITECTURE DESIGN

This chapter is based on the theoretical foundation of the previous text and combines with real teaching scenarios in higher education to conduct a comprehensive system requirements analysis, clarifying the system's usage roles, functional requirements, and non functional requirements; On this basis, complete the overall system architecture, hierarchical architecture, functional module division, and design the overall database structure and core data tables to build the overall framework for subsequent data modeling, algorithm development, and system implementation, as shown in Table 2.

TABLE 2. System core roles and functional permission configuration

Role	Core Operation Permissions	Data Access Scope	System Configuration Permission
Student	View learning paths, learn resources, complete assessments	Personal-only learning data	No configuration permission
Teacher	Manage teaching resources, view class learning data	Data of taught courses and classes	No system configuration permission
Teaching Administrator	Review resources, manage course system, view global statistics	Full college/university teaching data	Partial basic configuration permission

This system is designed for smart teaching scenarios in universities, and is divided into four core roles: students, teachers, teaching administrators, and super administrators, based on the organizational structure and teaching processes of universities. There are significant differences in the usage goals, operational permissions, and functional requirements of different roles, which are analyzed one by one as follows:

Students are the most core user group of the system and also the service objects of personalized learning paths, covering all grades and majors of students in universities. The core needs of students include logging into the system to view personal information, viewing recommended personalized learning paths, online learning course resources, completing supporting exercises, checking learning progress and knowledge mastery, autonomously adjusting learning paths, and querying historical learning records. Permission scope: Only personal related data can be operated, browsing and learning assigned learning content, and course information, system configuration, and other people's data cannot be modified.

Teachers include professional course teachers, public course teachers, and practical training guidance teachers, mainly responsible for course teaching and teaching data analysis. Teacher core needs: manage the courses and supporting learning resources taught, view the learning paths of all students in the class, monitor the overall learning progress and knowledge weaknesses of students, view course learning statistics reports, publish homework and supplementary learning resources, and adjust the order of course knowledge points according to the overall situation of the class. Permission scope: Manage the course data taught by oneself, view the learning data of students in the class, without advanced permissions such as system configuration, user addition, and permission modification.

Teaching administrators are subordinate to the Academic Affairs Office and the College Teaching Office of the university, responsible for the overall management of the school's curriculum system, professional training programs, and teaching resources. Core requirements: Maintain information on majors, grades, and classes throughout the school, manage all course information and system architecture, review teaching resources, view statistical analysis reports on teaching data across the school/college, and monitor the overall operation status of the system. Permission scope: Manage basic teaching data, course system data, view global teaching statistics data, do not have the system low-level configuration permission of a super administrator,as shown in Table 3 below..

TABLE 3. System core roles and functional permission configuration

Role	Core Operation Permissions	Data Access Scope	System Configuration Permission
Student	View learning paths, learn resources, complete assessments	Personal-only learning data	No configuration permission
Teacher	Manage teaching resources, view class learning data	Data of taught courses and classes	No system configuration permission
Teaching Administrator	Review resources, manage course system, view global statistics	Full college/university teaching data	Partial basic configuration permission

A. SYSTEM REQUIREMENTS ANALYSIS

Functional requirements refer to the business functions that the system must implement. Based on the usage scenarios of four types of roles, the system functions are divided into seven core functional modules, and the functional details are broken down one by one:

(1) User login and permission management module

1. Account login: Supports student ID/employee ID+ password login, supports verification code anti fraud, and distinguishes login entrances for different roles;
2. Account management: Super administrators can add, delete, modify, freeze, and reset passwords for accounts;
3. Permission allocation: Role based access control (RBAC) assigns different operational permissions to different roles;
4. Personal Center: All roles can view, modify personal basic information, and change login passwords.

In addition to basic login and account management functions, the system has added three auxiliary functions: remote login reminder, password strength verification, and automatic session timeout exit. When an account logs in with an unfamiliar IP or device, the system pushes internal messages to remind users and prevent account theft; Password setting requires the combination of letters, numbers, and special symbols to enhance password security; For accounts that have not been operated for a long time, the session will be automatically terminated to protect the security of backend data. At the same time, it supports batch account import and export, adapting to the batch account creation scenarios of new students and faculty joining each semester in universities, greatly improving the efficiency of administrators' work.

(2) Basic Teaching Data Management Module

This module is designed for administrators and teachers to maintain basic teaching system data in universities

1. Organizational structure management: Maintain the data of the school, college, major, grade, and class structure at four levels;
2. Student information management: Import, query, and edit basic student information;
3. Teacher Information Management: Maintain information such as teacher ID, name, and course of instruction;
4. Curriculum system management: Maintain course information, course classification, and pre - and post course relationships according to the professional training plan;
5. Knowledge point management: Break down knowledge points for each course, maintain the name, difficulty, chapter, and associated knowledge points of each knowledge point.

This module strictly benchmarks the academic management standards of universities, and the curriculum system fully conforms to the talent training plans of various majors. It supports the setting of course semester, credits, assessment methods, study types, and other attributes. Knowledge point decomposition supports multi-level layering, refining according to the four level structure of "course chapter section knowledge point". At the same time, manual annotation of precursor dependencies, parallel associations, and extended relationships between knowledge points provides raw data for knowledge graph construction. For large-scale data maintenance scenarios, the system provides Excel template import function, supporting data format verification and error data annotation. Management personnel can correct data based on error information to avoid inefficiency and errors caused by manual entry of each item.

Non functional requirements determine the operational performance, stability, and compatibility of the system, and are an important guarantee for the implementation and application of the system.

1. Performance requirement: The system supports concurrent online access by at least 3000 users from universities; Page response time $\leq$ 2 seconds; The response time for deep learning algorithms to generate learning paths is ≤ 3 seconds; The response time for large-scale data import and statistical report generation is ≤ 5 seconds. For peak access periods such as exam weeks and school start seasons, the system relies on load balancing mechanisms to distribute requests and avoid page lag and interface timeouts.
2. Stability requirement: The annual normal operation rate of the system should be $\geq 99.5\%$, and the downtime for a single failure should not exceed 30 minutes; Automatic scheduled backup of data to prevent data loss. The system sets up primary and backup service nodes, which automatically switch to backup nodes when the primary node fails, achieving seamless switching in case of failure.
3. Compatibility requirements: The front-end page is compatible with mainstream browsers such as Chrome, Edge, Firefox, etc; Adapt to screens of different resolutions on both computer and tablet devices. The page adopts a responsive layout and can be displayed normally on screens with resolutions ranging from 1024P to 2K. All core functions can be fully utilized on mobile tablets.
4. Scalability requirements: The system adopts a modular design, which can quickly add functional modules such as training management, subject competitions, and online Q&A in the future; The algorithm module supports the addition of deep learning models, which can be flexibly iterated and upgraded. The code architecture reserves extension interfaces, and new functional modules can seamlessly integrate into the existing system without the need to refactor the underlying code, as shown in Table 4.

TABLE 4. Core rules for classification and preprocessing of multi-source educational data

Data Category	Core Data Source	Key Preprocessing Rule	Main Application Scenario
Static Structured Data	University Educational Administration System	One-hot encoding for categorical features, min-max normalization for numerical values	Student basic portrait construction
Dynamic Temporal Data	System front-end buried point collection	Time series alignment, abnormal value elimination, behavior sequence splicing	Learning behavior feature extraction
Knowledge Graph Data	Manual annotation + system automatic construction	Node relationship verification, topology optimization, knowledge embedding vectorization	Learning path logical constraint

B. OVERALL SYSTEM ARCHITECTURE DESIGN

Combining the B/S architecture, modular thinking, data flow and technology stack, this article designs a five layer hierarchical architecture, from top to bottom: front-end display layer, application service layer, algorithm service layer, data support layer, and infrastructure layer. The hierarchical architecture decouples each module, reducing the difficulty of development, maintenance, and iteration. The overall architecture logic is clear and the division of labor is clear.

The infrastructure layer is the hardware and basic software environment that runs the system, providing underlying support for the entire system, including server clusters, operating system (LinuxCentOS7), network devices, firewalls, load balancing devices, database services (MySQL8.0, MongoDB 5.0), and caching services (Redis 6.0). This layer ensures system hardware stability, network security, and reliable data storage.

Server clusters are divided into four major clusters based on their functions: application servers, algorithm servers, database servers, and file servers, each performing their own duties. Deploy backend business code on the application server to handle user requests; The algorithm server independently deploys deep learning inference services to isolate algorithm computation pressure; The database server is specialized in running relational and non relational databases to ensure data read and write efficiency; The file server centrally stores teaching resources such as videos, courseware, and documents, sharing the storage pressure of the main server. The firewall enables port filtering and intrusion detection functions to resist external network attacks; The load balancing device implements request offloading to balance the running load of each server, as shown in Table 5.

TABLE 5. Summary of System Core Function Test Results

Test Item	Expected Performance Standard	Actual Test Result	Test Conclusion
Personalized learning path generation response time	$\leq$ 3 seconds	2.1 seconds	Pass
System concurrent user access capacity	$\geq$ 3000 simultaneous online users	3500 users stable operation	Pass
Learning path dynamic adjustment response time	$\leq$ 1 minute after assessment completion	0.8 seconds	Pass

The data support layer is responsible for the collection, storage, preprocessing, and synchronization of all educational data, and is the data source of the system. Divided into four sub modules:

1. Data collection module: front-end embedded point collection behavior data, interface docking with external teaching platform data, database synchronization static data;
2. Data preprocessing module: automates data cleaning, deduplication, missing value filling, normalization, and feature encoding;
3. Data storage module: MySQL stores structured business data, MongoDB stores unstructured behavior and graph data, Redis caches high-frequency access data (user login information, popular courses) to improve access speed;
4. Data synchronization module: Implement real-time synchronization of data between different databases and teaching platforms to ensure data consistency.

The data support layer sets data flow standards. After the raw data collection is completed, it is first standardized in the preprocessing module and then classified and stored in the corresponding database. Redis cache sets data expiration policies and periodically cleans up temporarily cached data to avoid memory overflow. The data synchronization adopts a combination of timed tasks and real-time triggering. The basic archive data is synchronized at regular intervals every hour, and dynamic data such as exam scores and learning behaviors are synchronized in real time to ensure complete consistency of data across multiple platforms.

The algorithm service layer is the intelligent core of the system, independent of the business system, dedicated to running deep learning models, calculating features, and generating recommendation results. It provides services to the upper layer through API interfaces. Divided into three sub modules:

1. Feature modeling module: Call DNN model to extract static features of students, LSTM model to extract temporal behavior features, GCN model to extract knowledge graph features, and complete multi-dimensional feature fusion;
2. Hybrid recommendation algorithm module: Run the trained hybrid deep learning model, receive feature data, generate personalized learning paths and resource recommendation results;
3. Strategy optimization module: Implement cold start optimization, dynamic path adjustment, abnormal data processing and other strategies to ensure the stable operation of algorithms in complex scenarios.

The algorithm service layer adopts a microservice architecture, where feature modeling, recommendation inference, and policy optimization are separated into independent interfaces that can be expanded and iterated separately. The model inference adopts a combination of batch processing and real-time processing mode, with real-time response to single user path requests and batch updating of all student portraits and knowledge features at night, balancing response speed and computational efficiency. In response to industry pain points such as cold start and sparse data for new students, the strategy optimization module is equipped with multiple sets of fallback rules to ensure that a reasonable learning path can still be output in extreme scenarios.

IV. LEARNING DATA PROCESSING AND MULTIDIMENSIONAL FEATURE MODELING

On the basis of completing the system architecture, functional modules, and database design in the previous section, this chapter focuses on the data layer and feature modeling layer, fully elaborates on the collection standards and full process preprocessing scheme of multi-source heterogeneous education data in higher education, sequentially completes the modeling of student multi-dimensional user profiles, curriculum knowledge point education knowledge graph, and finally outputs a standardized feature dataset, providing high-quality data support for the training and inference of deep learning hybrid recommendation algorithms in Chapter 5. This chapter is divided into five sections: overall data analysis, data collection system, data preprocessing, student user profile modeling, and educational knowledge graph modeling, as shown in Table 6 below.

TABLE 6. Comparison of core differences between this study and existing mainstream research

Research Dimension	Existing Mainstream Research	This Research	Core Advantage
User Portrait Modeling	Shallow feature mining based on grades and course selection	Three-dimensional portrait integrating static, behavioral and mastery features	More accurate characterization of students' learning state
Learning Path Logic	Interest-driven recommendation, ignoring professional knowledge system	Knowledge graph as hard constraint, conforming to cognitive law	Avoid disordered path recommendation, stronger practicability
Algorithm Adaptability	Single algorithm, poor performance in sparse data scenarios	Hybrid deep learning model with cold start optimization strategy	Stable performance in complex university teaching scenarios

A. OVERALL ANALYSIS OF MULTI-SOURCE DATA IN EDUCATION

The learning data in the context of higher education has five characteristics: multiple sources, miscellaneous types, huge size, strong timing, and complex association. Different from the general recommendation data on the Internet, the education data has strong domain specialization and logical constraints. All data in this system comes from real teaching scenarios in universities, and is classified and sorted in a comprehensive manner according to data attributes, data structures, and business purposes, clarifying the characteristics, application scenarios, and processing requirements of each type of data.

Based on the existing information system of universities, the data in this system is divided into four major source channels, and the data from each channel complements and cross verifies each other, forming a complete data loop:

1. University Academic Management System: Core static data source, providing official structured data such as teacher-student basic files, professional class structure, curriculum system, training plan, credit information, historical exam scores, course selection records, etc. This type of data has the highest authority and is the fundamental basis for building a student profile and curriculum knowledge system. The data update frequency is low and the stability is strong.
2. The front-end of this system collects data: the core dynamic behavior data source, which collects fine-grained behaviors of students throughout the entire online learning process through front-end embedding and interactive monitoring, including page clicks, video playback, pause, review, drag and drop, resource collection, download, answer operation, dwell time, etc. This type of data has the largest volume and strongest real-time performance, and is the core basis for analyzing learning habits, learning status, and knowledge mastery.
3. On campus examination and question bank system: an assessment data source that provides data on in class exercises, homework, unit tests, mid-term and final exams, answer records, scores, and error records. This type of data directly reflects students' mastery of knowledge points and is a key indicator for dynamically adjusting learning paths and updating knowledge mastery labels.
4. Third party online teaching platform: Supplement data sources, connect with general teaching platforms such as Xuetang Online and Chaoxing Learning Platform, synchronize cross platform learning records, course resources, and interactive data, break platform data silos, and achieve unified modeling of learning behaviors in all scenarios.

According to the data storage structure and format, the entire data is divided into three categories: structured data, semi-structured data, and unstructured data, and targeted processing plans are formulated:

1. Structured data: With fixed fields and formats, it can be directly stored in relational databases. Representative data: teacher-student profiles, course information, grades, class structure, answer results, etc. This type of data has high regularity and low preprocessing difficulty, mainly used for building static features.
2. Semi structured data: There is no strict data table structure, but it has unified tags and formats, existing in JSON, XML, key value and other forms. Representative data: learning path details, behavior sequences, tag sets, resource attributes, etc. This type of data needs to be parsed and formatted before extracting effective features.
3. Unstructured data: There is no fixed format, mainly consisting of files and streaming media data. Representative data: course videos, PPT presentations, document materials, images, student comments and messages, etc. This type of data is not directly input into the algorithm and is only used as a learning resource, extracting only file attributes and access behavior as features.

B. DESIGN OF A FULL DIMENSIONAL DATA COLLECTION SYSTEM

Based on the previous data sources, build a data collection system that is "multi-channel collaboration, full scene coverage, high real-time performance, and high reliability", clarify the collection methods, fields, frequency, and transmission rules, and ensure that the original data is complete, effective, and traceable.

This system adopts a combination of three mainstream technologies: database synchronization, API interface docking, and front-end buried point collection to achieve full data collection. The three methods have clear division of labor and complement each other:

1. Database synchronization collection: Targeting internal same source databases such as academic and examination systems, a master-slave data synchronization scheme is adopted, and incremental synchronization is achieved based on Binlog logs. Mainly collect static structured data such as basic information of teachers and students, course structure, and historical grades, with a synchronization frequency set to once per hour. During leisure time (2:00-4:00 am),

perform full data verification and completion. This method ensures stable transmission, high data integrity, and does not occupy operational resources of the business system.
2. API interface docking and collection: Targeting third-party online teaching platforms and off campus resource platforms, data is pulled based on the RESTful API interface. Regularly request data according to platform interface specifications, synchronize cross platform learning records and resource information, set signature verification and frequency restrictions for interface calls to prevent interface blocking and data leakage.
3. Front end embedded point collection: Targeting user interaction behavior within the system, adopting a non-invasive JS embedded point solution, deployed on all pages, video players, and answer components. Real time monitoring of every user interaction action, collecting behavior data in milliseconds, and uploading the data to the background through asynchronous requests, without blocking page rendering and normal operations. Burial points are divided into general burial points and special burial points: the general burial point collection page can be accessed, exited, clicked, and stopped; Specialized buried points refine the collection fields for core scenarios such as video playback, answering questions, and resource operations.

Based on the subsequent feature modeling requirements, define the required collection fields for various types of data to ensure the completeness of feature dimensions:
1. Basic file data collection fields: user ID, account, name, gender, major, class, enrollment year, pre course grades, total enrollment score, role type.
2. Course and Knowledge Point Data Collection Fields: Course ID, Course Name, Course Difficulty, Credits, Course Type, Pre Course ID, Knowledge Point ID, Knowledge Point Name, Knowledge Point Difficulty, Pre Knowledge Point ID.
3. Learning behavior data collection fields: user ID, course ID, knowledge point ID, resource ID, behavior type, start time, end time, dwell time, operation action (play/pause/look back/bookmark), device type, IP address.
4. Assessment answer data collection fields: user ID, knowledge point ID, question ID, answer, answer result (correct/incorrect), answer time, answer time, and test paper type.

The collected raw data is first stored in the Redis temporary cache queue, and then batch transferred to the preprocessing module to avoid data loss and disorder in high concurrency scenarios. Real time behavioral data adopts a queue first in, first out rule to ensure that the temporal sequence of behaviors is not disrupted; Static archive data updates the latest values. All data in the transmission process are desensitized. Privacy fields such as mobile phone number and ID number are masked to protect user privacy from the transmission link.

V. CONCLUSION

This article addresses the issues of homogenization in traditional higher education teaching models, data sparsity, cold start, and insufficient logical learning paths in existing learning recommendation systems. A personalized learning path recommendation system for higher education based on deep learning has been designed and implemented. The research combines educational theories such as constructivism and cognitive load, integrates multiple deep learning algorithms such as DNN, LSTM, and GCN, completes multi-source educational data collection, preprocessing, student profiling, and curriculum knowledge graph construction, and builds a layered intelligent teaching platform. Through testing and comparative experiments, it has been verified that the system can accurately mine students' learning characteristics and knowledge weaknesses, dynamically generate personalized learning paths that conform to cognitive laws, and significantly improve recommendation accuracy, recall rate, and other indicators compared to traditional recommendation algorithms. The system has complete functions and stable operation, which can effectively solve practical problems such as overload of teaching resources and difficulty in implementing personalized teaching in universities. This study still has limitations such as insufficient interdisciplinary knowledge association mining and the need to improve extreme sample adaptability. In the future, the algorithm model can be further optimized to expand multi terminal linkage and intelligent interaction functions, and continue to promote the landing and application of deep learning technology in personalized teaching scenarios in higher education.

ACKNOWLEDGMENT

REFERENCES

[1] G. K. Murtaza and C. Libin, "Ethical Challenges and Drawbacks of AI-Driven Personalization in Higher Education: A Review," Asian Journal of Current Research, pp. 16–25, 2026.
[2] T. Tao, "The optimization strategy of personalized english teaching in higher education based on fuzzy control," Discover Artificial Intelligence, vol. 6, no. 1, pp. 214–214, 2026, doi: 10.1007/S44163-025-00766-Z.
[3] W. Yanlong, "The Construction and Effect Evaluation of Personalized English Teaching Models in Higher Education under the Background of Artificial Intelligence," Education Research and Innovation, vol. 1, no. 8, 2025, doi: 10.62639/SSPERI30.20250108.
[4] C. Sun, S. Huang, B. Sun, et al., "Personalized learning path planning for higher education based on deep generative models and quantum machine learning: a multimodal learning analysis method integrating transformer, adversarial training and quantum state classification," Discover Artificial Intelligence, vol. 5, no. 1, pp. 29–29, 2025, doi: 10.1007/S44163-025-00252-6.
[5] M. C. Campos, "The Impact of Artificial Intelligence on Personalized Learning in Higher Education: A Systematic Review," Trends in Higher Education, vol. 4, no. 2, pp. 17–17, 2025, doi: 10.3390/HIGHEREDU4020017.
[6] X. Wu, X. Xie, and H. Mo, "Research on Big Data-driven Personalized Teaching Models in Higher Education," Journal of Big Data and Computing, vol. 2, no. 4, 2024, doi: 10.62517/JBDC.202401404.
[7] J. Cai and Y. Li, "Fuzzy association rule mining for Personalized English Language Teaching from higher education," Journal of Computational Methods in Sciences and Engineering, vol. 24, no. 6, pp. 3617–3631, 2024, doi: 10.1177/14727978241296748.
[8] F. Teng, "Fuzzy Association Rule Mining for Personalized Chinese Language and Literature Teaching from Higher Education," International

Journal of Computational Intelligence Systems, vol. 17, no. 1, pp. 266–266, 2024, doi: 10.1007/S44196-024-00676-5.
[9] R. Sajja, Y. Sermet, M. Cikmaz, et al., "Artificial Intelligence-Enabled Intelligent Assistant for Personalized and Adaptive Learning in Higher Education," Information, vol. 15, no. 10, pp. 596–596, 2024, doi: 10.3390/INFO15100596.
[10] D. Jiao, "AI-Driven Personalization in Higher Education: Enhancing Learning Outcomes through Adaptive Technologies," Adult and Higher Education, vol. 6, no. 6, 2024, doi: 10.23977/ADUHE.2024.060607.
[11] Z. Lin, "A systematic review of personalized learning in higher education: learning content structure, learning materials sequence, and learning readiness support," Interactive Learning Environments, vol. 31, no. 10, pp. 7053–7073, 2023, doi: 10.1080/10494820.2022.2061006.
[12] A. Mannion, R. Coyne, C. Ferrari, et al., "Personalized System of Instruction in Higher Education: A Systematic Review," Journal of Behavioral Education, vol. 34, no. 2, pp. 1–41, 2023, doi: 10.1007/S10864-023-09530-8.
[13] D. Sara, C. Cayla, M. Michelle, et al., "Individualized Support in Higher Education Using Assistive Technology (AT)," AMERICAN JOURNAL OF OCCUPATIONAL THERAPY, vol. 74, no. 4, 2020, doi: 10.5014/ajot.2020.74S1-PO1310.
[14] "Personalized Learning Environment in Higher Education through Big Data and Blended Learning Analytics," International Journal of Recent Technology and Engineering, vol. 8, no. 3, pp. 6236–6239, 2019, doi: 10.35940/ijrte.c5778.098319.
[15] B. T and R. S, "Personalized Learning Environment in Higher Education through Big Data and Blended Learning Analytics," International Journal of Recent Technology and Engineering (IJRTE), vol. 8, no. 3, pp. 6236–6239, 2019.
[16] L. Olexandr and S. Inna, "SECOND LANGUAGE TEACHING STRATEGIES WITHIN THE FRAMEWORK OF INDIVIDUALISATION AND DIFFERENTIATION IN HIGHER EDUCATIONAL INSTITUTIONS," Journal of Teaching English for Specific and Academic Purposes, vol. 6, no. 1, pp. 107–107, 2018, doi: 10.22190/jtesap1801107l.
[17] P. Hendrad, B. H. Santoso, and A. Z. Hasibuan, "Use Clustering Data of Student High School for Placement in Personalization E-Learning on Higher Education," International Journal of Innovative Science and Modern Engineering (IJISME), vol. 4, no. 10, pp. 7–12, 2017.
[18] A. Risquez and S. Moore, "Exploring feelings about technology integration in higher education Individuation and congruence," Journal of organizational change management, vol. 26, no. 2, pp. 326–339, 2013.
[19] N. Becket and M. Brookes, "The potential benefits and challenges of personalising UK higher education," Journal of Hospitality, Leisure, Sport & Tourism Education, vol. 11, no. 1, pp. 21–28, 2012, doi: 10.1016/j.jhlste.2012.02.004.
[20] B. Goldschmid and L. M. Goldschmid, "Individualizing Instruction in Higher Education: A Review," Higher Education, vol. 3, no. 1, pp. 1–24, 1974, doi: 10.1007/BF00153989.