Research on Personalized Teaching Strategy of Computer Education Based on Big Data Analysis

Junfei Jin*
Zhengzhou Software Vocational and Technical College, China
Corresponding author: jinjunfei@163.com

Abstract

With the rapid development of information technology, big data analysis has become the core driving force for educational change. In the field of computer education, personalized teaching strategy significantly improves the teaching quality and learning effect by adapting to the individual differences of students. Based on big data analysis technology, this paper discusses the implementation path of personalized teaching in computer education: through the construction of a learning behavior data analysis model, the dynamic customization of teaching content, relying on intelligent recommendation algorithms, optimizing personalized learning path; combined with a multi-dimensional evaluation system, improving the teaching feedback mechanism. At the same time, for the current challenges of computer education, such as rapid technology iteration, large differences in students' foundation, and uneven resource allocation, we propose a solution that integrates big data technology and verifies the effectiveness of the strategy through case studies. The study aims to provide theoretical references and practical paradigms for the intelligent and personalized development of computer education.

Keywords

Big Data, Computer Education, Personalized Instruction, Learning Analytics, Intelligent Recommendations

The core of personalized teaching lies in “student-centeredness”, which provides adaptive teaching support by accurately capturing learners' cognitive characteristics, learning needs, and behavioral patterns.

In traditional computer education, the “one-size-fits-all” teaching model struggles to accommodate differences in students' knowledge bases, learning paces, and interest orientations, leading some students to fall into the dilemma of “can't keep up” or “not challenged enough”.

The advancement of big data technology offers a solution to this problem by collecting multidimensional data from students' classroom interactions, online learning, and practical operations, and leveraging data mining and machine learning to analyze learning patterns, enabling scientific teaching decisions and precise educational services.
Currently, computer education is undergoing a critical transition from “standardized training” to “personalized development”. On the one hand, the widespread adoption of cloud computing and artificial intelligence has facilitated the digitization of teaching resources, providing a robust foundation for data collection. On the other hand, the industry's demand for computer talent has become increasingly diverse, requiring an

education system capable of nurturing individuals with both theoretical depth and practical innovation skills.

In this context, exploring big data-driven personalized teaching strategies is not only an inevitable response to the challenges of educational equity and quality but also a key step in advancing the high-quality development of computer education.

Mechanisms for applying big data analytics in personalized instruction

Multi-dimensional collection and modeling of learning behavior data

Learning behavior data is the “cornerstone: of personalized teaching, and its completeness and accuracy directly determine the effectiveness of the strategy. In the computer education scenario, the data collection covers the whole scenario of “online offline”, specifically including: basic attribute data: students’ professional background, admission scores, prerequisite courses, etc., which are used to build the initial learning portrait; process data: video viewing hours, code submission times, forum interaction frequency, homework correction trajectory, etc., which reflect the real-time learning status; and process data: video viewing hours, code submission times, forum interaction frequency, homework correction track, etc., which reflect the real-time learning status; and learning behavior data. Process data: online platform video viewing hours, code submission times, forum interaction frequency, homework error correction track, etc., reflecting real-time learning status; outcome data: test scores, project acceptance results, skill certification levels, etc., assessing the degree of knowledge mastery; context data: learning device type, login time, geographic location, etc., to assist in analyzing the impact of the learning environment.

After integrating multi-source information through the construction of a “data lake”, it is necessary to use cluster analysis, association rule mining, and other techniques for modeling[1]. For example, using the K-means algorithm to cluster students' programming practice data, different learning disabilities such as “grammatically weak”, “logically deficient”, and “application deficient” can be identified. By analyzing the correlation between “video viewing time” and “correct rate of homework” through association rules, we can find that “students who watch videos for more than 30 minutes and take notes can increase the correct rate of homework by 27%”, which can provide a good basis for teaching and learning[2]. This can provide a basis for teaching intervention.

Table 1. Learning behavior data types and analysis dimensions.

Datatype	Collection sources	Core analysis dimensions	Application Scenario
Basic Attribute Data	School registration system, questionnaires	Starting point of knowledge, learning potential	Initial instructional subgroups
Process data	LMS platform, programming environment	Learning Input, Difficulty Distribution	Real-time instructional adjustments

Intelligent Recommendation of Personalized Learning Path

Based on the results of learning behavior data modeling, the intelligent recommendation algorithm can create a learning path for “thousands of people with thousands of faces.” The core idea is to use the knowledge graph as the framework, combined with students' learning progress and ability levels, to dynamically adjust the learning nodes and resource sequences. For example, in the “Python Programming” course, the system initially builds a basic pathway of “Grammar Basics-Functional Programming-Object-Oriented-Project Combat,” then customizes it for different students: for those with weak grammar skills, it adds reinforcement through “Grammar Micro classes Instant Exercises” to lower the difficulty; for students with strong logical thinking, repetitive exercises are skipped, and more challenging tasks like “Algorithm Competition Problems Contributions to Open Source Projects” are directly recommended; for application-focused students, an application-oriented module such as “data analysis case - visualization tool practice” is included.

The selection of recommendation algorithms needs to be combined with the teaching objectives: collaborative filtering algorithms are suitable for mining the common needs of the student group, content-based recommendation focuses more on matching individual interests, and reinforcement learning algorithms can optimize long-term learning paths through continuous feedback. The practice of a higher vocational college shows that after adopting the hybrid recommendation strategy, the course completion rate of students increased by 32%, and the degree of knowledge mastery increased by 28%.

Dynamic Adaptation and Optimization of Teaching Resources

The personalized adjustment of teaching resources should achieve a three-dimensional synergy of “content-method-scene.” At the content level, high-frequency error knowledge points are identified through big data analysis, and courseware and exercises are updated accordingly[3-5]. For example, when data shows that 80% of students stay more than 15 minutes at the “pointer concept” and the error rate reaches 60%, the static text analysis will be replaced with multimedia resources such as “animation demonstration interactive exercises.”

At the level of teaching methodology, differentiated strategies are implemented based on the results of learning style classification: visual learners: provide mind maps, flowcharts, and visualization tools for code execution; auditory learners: support audio explanation of knowledge points and audio review function for group discussions; kinesthetic learners: design virtual simulation experiments and programming sandbox hands-on tasks.

At the scenario level, adaptive resources are pushed in conjunction with contextual data: for students logging in on the mobile terminal, lightweight micro-lessons and fragmented exercises are recommended as a priority; for students in laboratory scenarios, equipment operation guides and troubleshooting casebooks are pushed.

Status and challenges of computer education

Development history and technology-driven features

The evolution of computer education has always been deeply bound to technological innovation, and can be divided into four stages: Foundation period (1950s-1980s): hardware principles and assembly language as the core, the teaching goal is to train “technical operators”, the representative courses are Principles of Computer, Machine Language Programming; Popularization period (1990s-2010s): with the popularization of PCs and the Internet, the curriculum was expanded to include software development and network technology, emphasizing “application ability”, with the addition of C programming and local area programming. Machine Language Programming; popularization period (1990s-2010s): with the popularization of PCs and the Internet, the curriculum was expanded to include software development and network technology, emphasizing “application ability”, and adding “C Programming” and “Local Area Network Construction”; integration period (2010s-2020s): big data and cloud computing promote interdisciplinary integration, introducing statistics, management, and other new technologies. Integration period (2010s-2020s): big data and cloud computing promote interdisciplinary integration, introduce statistics and management knowledge, and cultivate “composite innovators”, with representative courses such as Introduction to Big Data Analysis and Cloud Platform Architecture; Intelligence period (2020s-present): AI and education are deeply integrated, focusing on personalization and creativity, with the addition of new courses such as Application of Machine Learning for Teaching and Educational Data Mining[6]. .

Current Core Challenges

(1) The contradiction between technology iteration and lagging teaching content

Computer technology is updated by one generation every 18 months on average, while the revision cycle of the course syllabus often reaches 3-5 years, resulting in a disconnect between the skills mastered by the students and the industrial needs. For example, when enterprises commonly use Python for data analysis, some institutions still use C++ as the only programming teaching language.

(2) Conflict between students' basic differences and the homogenization of teaching

Students in the same class may have extreme differences between “zero programming foundation” and “competition award-winning experience”, and the traditional “uniform progress” teaching results in about 30% of students being taught in the same class. As a result of traditional “uniform progress” teaching, about 30% of the students “can't keep up” and 20% of the students “can't learn enough”.

(3) Constraints of uneven distribution of practice resources. High-quality laboratories and enterprise-level development environments are mostly concentrated in key institutions, while the per capita practice hours of students in local institutions are only 1/3 of the former, which affects the transformation of theory into practice.

(4) Data Privacy and Security Risks Learning behavior data contains a large amount of personal information, which may lead to privacy leakage if not properly managed, such as the practice records of 100,000 students being made public due to loopholes in an online platform.

Pathways and Cases for Implementing Individualized Instructional Strategies

Accurate customization of teaching content

Relying on the knowledge map and learning diagnosis system, the “dynamic generation” of teaching content is realized. Take the “Principles of Database” course as an example:

(1) Knowledge mapping: the course is broken down into four core modules: “Relational Model - SQL Syntax - Index Optimization - Transaction Processing”, with each module containing 3-5 key knowledge points to form a visual knowledge network.

(2) Diagnostic assessment: the beginning of the school year, through the adaptive test to locate students' weak points, such as a student in the “index optimization” module correct rate of only 40%, the system automatically marked as a key to strengthen the content.

(3) Dynamic push: real-time adjustment of the content difficulty according to the progress of learning. when the student correctly completes the basic questions for the third consecutive time, it will automatically push the advanced cases (such as the “Index Optimization” module correctly, and the system will automatically push the advanced cases)[7]. When students complete the basic questions correctly three times in a row, the system automatically pushes advanced cases (e.g., “optimizing database performance of e-commerce platform”). Conversely, the system provides remedial resources such as “animation analysis and attribution of wrong questions”.

Adaptive adjustment of teaching methods

Adopts the mode of “intelligent grouping flexible progress” to adapt to individual differences[8-10]. The practice of a software vocational college is as follows: dynamic grouping: based on the initial assessment and learning style analysis, students are divided into the “solid foundation group” (focusing on conceptual comprehension), the “ability enhancement group” (focusing on case practice), the “innovation challenge group” (focusing on case practice), the “innovation challenge group” (focusing on case practice), and the “innovation challenge group” (focusing on case practice). Innovation and Challenge Group“ (focusing on project development), with the ratio of each group being about 4:4:2; Flexible schedule: the Foundation Group completes 2 knowledge points 3 basic exercises per week, the Enhancement Group completes 3 knowledge points 1 case study, and the Challenge Group completes 4 knowledge points 1 small project, allowing the group to adjust the tempo on its own; Cross-group collaboration: ”Mixed Tasks" are set up every month to provide the students with the opportunity to work on a variety of tasks. “Mixed tasks”, such as allowing students in the challenge group to lead the foundation group to complete a simple system development, promoting mutual learning. After one semester of implementation, the college's course pass rate increased from 78% to 92%, with a student satisfaction rate of 90%[11].

Diversified Innovation of Evaluation Methods

A “process multi-dimensional” evaluation system is constructed to replace the traditional single examination.

(1) Real-time feedback: code quality analysis tools are embedded in the programming exercises, prompting real-time “naming specification problems” and “algorithm complexity optimization suggestions”, and recording modification trajectories.

(2) Capability Mapping: Generate radar charts from four dimensions of “knowledge mastery”, “practical ability”, “innovative thinking”, and “collaboration and

communication” to visualize the strengths and weaknesses.

(3) Predictive Evaluation: Analyzing learning data through machine learning models, we predict students who are likely to fail a course 4 weeks in advance, and push “one-on-one tutoring focus on key points” intervention programs.

Conclusion

Big data for computer education personalized teaching provides a new paradigm of “technology-enabled” learning, through the analysis of learning behavior, intelligent path recommendation, dynamic resource adaptation, and other strategies, effectively cracking the dilemma of traditional teaching's homogeneity. Future development needs to focus on three aspects: first, strengthening data literacy training for teachers to enhance their ability to use analytical tools; second, establishing data security norms to protect students' privacy in collection and application; and third, promoting the in-depth fusion of “technology education” to develop smarter adaptive learning systems.

With the maturity of 5G and AI technologies, personalized teaching will evolve in the direction of “ubiquitous” and “immersive”, such as simulating programming scenarios through VR, and identifying the learning state by using affective computing, etc. This will not only improve the quality of computer education but also enhance the quality of computer education. This will not only improve the quality of computer education but also cultivate students' independent learning ability, lay the foundation for lifelong learning, and ultimately realize the goal of “letting every student get the most suitable education”.

Funding: This research received no external funding.

Acknowledgments: The authors would like to express sincere gratitude to the technicians who contributed to this research.

References

[1] Rahmawati, R., Nursalim, N., Alfiah, A., Hasyim, A., Fawait, A. B. (2024). Impact of using big data analysis in increasing the personalization of learning. Journal of Computer Science Advancements, 2(2), 54-72.

[2] Ikegwu, A. C., Nweke, H. F., Anikwe, C. V. (2024). Recent trends in computational intelligence for educational big data analysis. Iran Journal of Computer Science, 7(1), 103-129.

[3] Grillenberger, A., Romeike, R. (2014). Big data–challenges for computer science education. In International Conference on Informatics in Schools: Situation, Evolution, and Perspectives, (29-40).

[4] Paul, P. K., Ghose, M. K. (2017). A novel educational proposal and strategies toward promoting cloud computing, big data, and human–computer interaction in engineering colleges and universities. In Advances in Smart Grid and Renewable Energy: Proceedings of ETAEERE-2016 (93-102).

[5] Picciano, A. G. (2012). The evolution of big data and learning analytics in American higher education. Journal of Asynchronous Learning Networks, 16(3), 9-20.

[6] Sezer, O. B., Dogdu, E., Ozbayoglu, A. M. (2017). Context-aware computing, learning, and big data in internet of things: a survey. IEEE Internet of Things Journal, 5(1), 1-27.

[7] Thompson, G. (2017). Computer adaptive testing, big data, and algorithmic approaches to education. British Jurnal of Sociology of Education, 38(6), 827-840.

[8] Najafabadi, M. M., Villanustre, F., Khoshgoftaar, T. M., Seliya, N., Wald, R., Muharemagic, E. (2015). Deep learning

applications and challenges in big data analytics. Journal of Big Data, 2(1), 1.

[9] Kaur, P., Sharma, M., Mittal, M. (2018). Big data and machine learning based secure healthcare framework. Procedia Computer Science, 132,1049-1059.

[10] Plageras, A. P., Psannis, K. E., Stergiou, C., Wang, H., Gupta, B. B. (2018). Efficient IoT-based sensor big data collection, processing, and analysis in smart buildings. Future Generation Computer Systems, 82, 349-357.

[11] Sadovaya, V. V., Korshunova, O. V., Nauruzbay, Z. Z. (2016). Personalized education strategies. International Electronic Journal of Mathematics Education, 11(1), 199-209.