

Learning Analytics and Personalized Learning Resource Recommendation in a University Computer Course: An Empirical Study

Han Wang, Changqi Li

School of Big Data, Baoshan University, Baoshan, Yunnan, China

Copyright: © 2026 Author(s). This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY 4.0), permitting distribution and reproduction in any medium, provided the original work is cited.

Abstract: With the development of digital learning environments in higher education, researchers and educators have begun to give more and more attention to how to use large-scale learning process data to provide good, personalized instructional support. The subject of research for this paper is the university's computer course "Fundamentals of Programming," and a personalized learning-resource-recommendation framework based on learning analytics has been developed and applied to collect behavior data, implement Bayesian knowledge tracing, create learner profiles, and perform hybrid resource ranking. A one-semester quasi-experimental study was carried out on 102 first-year undergraduate students; one group ($n = 52$) was given personalized resource recommendations as the experimental group, and the other group ($n = 50$) continued to use the traditional fixed course materials as the control group. Based on the above results, the experimental group performed better than the control group in the post-test ($t = 5.63$, $P < 0.001$, Cohen's $d = 1.10$), showed more active learning behavior, and reached a recommendation precision@5 of 0.78; this is much higher than that of the popularity-based baseline (0.41). Therefore, it has been decided that learning analytics will be employed to integrate process data with personalized instruction in university computer courses, and thus a reproducible technical path and empirical evidence for precision teaching have been obtained in this field.

Keywords: Learning analytics; Personalized recommendation; University computer course; Knowledge tracing; Empirical study

Online publication: September 30, 2026

1. Introduction

University computer courses, such as programming basics, data structures, and database principles, are generally abstract in nature, have strong logical dependencies among the points of knowledge, and show a wide range of students' prior knowledge. Teachers in the traditional classroom generally teach the same content and exercises for all students, fail to consider individual differences, and thus advanced students are often under-stimulated; at the same time, misunderstandings among struggling students may be carried over to the next topic. With the spread of Learning Management Systems (LMS) and online judge platforms, times

have changed; now, every click, quiz attempt, code submission, and forum interaction can be recorded as detailed process data ^[1]. Learning analytics is defined as the measurement, collection, analysis, and reporting of data on learners and their environments to understand how they learn and improve teaching; thus, this type of data can be used to make teaching decisions ^[2].

At the same time, the quantity of available digital learning materials has been expanding rapidly. Video lectures, interactive tutorials, coding exercises, and open courseware are all good options, but there are too many choices, which is called information overload. Recommender systems have been introduced in education to match resources with the needs of learners ^[3]. The first educational recommenders were based on content similarity or collaborative filtering from e-commerce, and they did not consider the pedagogical structure of knowledge or changes in the state of learners ^[4]. Recently, some studies have begun to combine knowledge graphs, deep neural networks, and human-centered design to make more accurate, understandable, and educationally effective recommendations ^[5,6].

Nevertheless, three gaps remain in the literature. First, most of the recommendation research has been carried out on public MOOC datasets, and there is a shortage of empirical evidence from real, semester-long university computer courses^[7]. Second, learner modeling and resource recommendation are often separate modules, so the recommendations cannot be quickly updated in light of changes in the learner's knowledge state. Third, there are no behavioral observations of learning activities or the corresponding levels of achievement; thus, the explanation for the observed effects is weak ^[8]. To address the problems mentioned above, this paper will construct an integrated learning-analytics and recommendation framework and carry out an experiment in a real university computer course over the entire semester. RQ1: Is the increase in students' grades after personalized resource recommendation higher than the increase in the traditional fixed-content model? RQ2: What changes occur in students' learning behavior after the intervention? RQ3: How far is the proposed hybrid recommendation algorithm from the popularity-based baseline?

2. Related work

Learning analytics in higher education has gone from descriptive dashboards to predictive and prescriptive applications. Khor and K ^[1] have systematically studied and found that the first kinds of support for personalized learning based on learning analytics are early warning prediction, adaptive content delivery, and reflective feedback, and they have also conducted more classroom-based empirical studies. Alfredo and others ^[5] have studied how to use learning analytics for students and other interested parties and found that participation and understanding need to be organized in these systems. Gao and others ^[8] have divided the learning paths in an introductory programming course according to scores, participation, and code metrics, and found that process data can reveal whether a student is doing well or struggling long before the end of the semester. Zhang and others ^[9] have also found that multimodal learning analytics can show how scaffolding and all kinds of examples work together to affect programming learning behavior.

Narimani and Barbera ^[4] have studied course recommendation systems in their paper and found that the two essential parts of personalization are learner profiling and course feature extraction. Liu and others ^[2] have put forward self-attention convolutional neural networks and deep metric attention models to improve the adaptation of online learning resources and enhance the responsiveness of recommendations to the needs of learners. Raj and Renumol ^[3] have put forward an adaptive learning path recommendation model that adjusts the order of learning materials in real time based on the current level of knowledge of students. Knowledge graphs have also been used to model prerequisite relationships among concepts: Chen and others

^[6] built a course knowledge graph with large language models and employed it for course recommendation, and Chen and others ^[10] have explored the application of knowledge graphs and large language models in education. Konstantinidou and others ^[11] have also studied what university staff know about AI-driven personalized learning and analytics; they have high hopes but are also worried about data governance.

Many new tools for educational research have been developed with deep learning. Wang ^[7] has combined recurrent neural networks and attention mechanisms for content recommendation, fine-tuned large language models to automatically generate feedback, and used random forests to predict early risks in college English education; he has shown that a pipeline of complementary models can be employed to achieve personalization, assessment and prediction simultaneously. Although the above multi-model architectures are very inspiring, they also increase the demand for interpretability and the amount of data required, and this is often not available in a single good package. Therefore, light-weight, transparent models are suitable for course-level deployment.

3. Framework and methodology

3.1. Overall architecture

The five levels of the proposed framework are shown in **Figure 1**. The Data Collection Layer will collect all kinds of process data from the LMS, online judge, and course forum, such as page views, video watching, quiz attempts, code submissions, and discussion posts. Organize the raw data in the learning analytics layer into behavior features and then use Bayesian knowledge tracing (BKT) to determine how well each student has mastered all the knowledge points. The Learner Model Layer does not have a fixed size and contains the knowledge-state vector, learning-behavior preference, and forgetting estimate. The recommendation engine calculates the combined score of all candidate resources and shows the top-ranked ones on the student's personal page each week. Finally, the feedback dashboard will notify teachers of problems in their classes and then address them promptly.

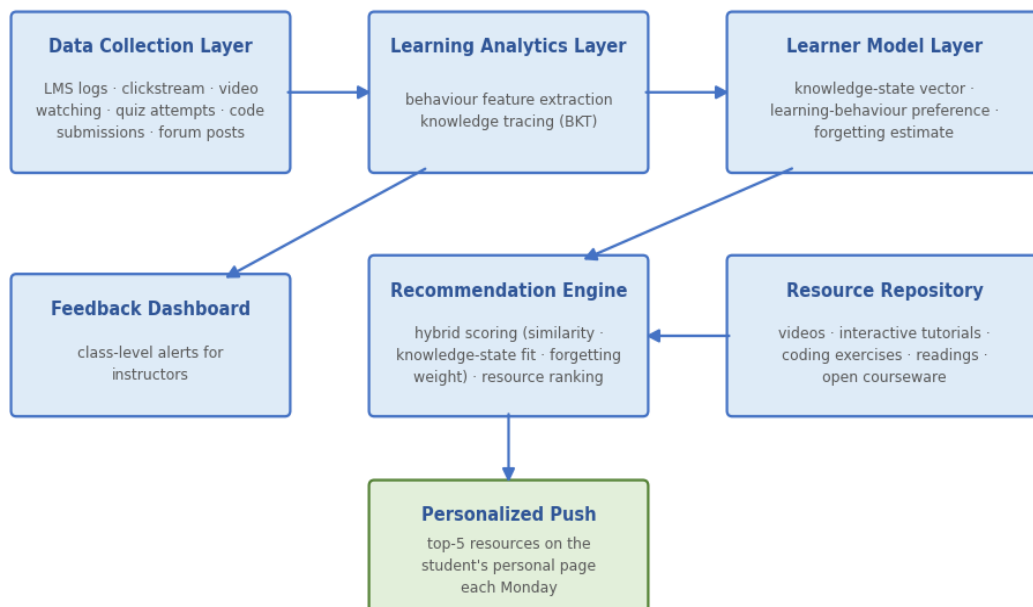


Figure 1. Architecture of the learning-analytics-driven personalized recommendation framework

3.2. Knowledge tracing

Bayesian knowledge tracing is used to model the learner's knowledge state. Let $P(L_t)$ be the probability that a student has mastered a knowledge point at time t . Based on the observed response of the student, update the mastery probability according to equation (1), where $P(T)$ is the learning transition probability, $P(G)$ is the guessing probability, and $P(S)$ is the slipping probability:

$$P(L_{t+1}) = P(L_t | obs) + [1 - P(L_t | obs)]xP(T) \quad (1)$$

where $P(L_t | obs)$ is the posterior obtained from the standard BKT conditionalization on a correct or incorrect observation. The parameters were initially set to $P(L_0)=0.30$, $P(T)=0.12$, $P(G)=0.20$ and $P(S)=0.10$ according to the general rule, and then fitted using expectation-maximization on the first four weeks of data. A knowledge point is considered mastered when $P(L) > 0.85$.

3.3. Hybrid recommendation scoring

Based on the combination of content relevance, knowledge-state fitness, and recency, order the Candidate Resources. The semantic relevance of learner u and resource r is given by the cosine similarity of their embedding vectors, as shown in equation (2):

$$sim(u, r) = (v_u \cdot v_r) / (\|v_u\| \|v_r\|) \quad (2)$$

The total recommendation score is given by equation (3), and $KS(u, r)$ is the degree to which the resource is in line with a student's unmastered knowledge points; $W(t) = \exp(-\lambda \times \delta_t)$ is an Ebbinghaus-type forgetting weight that gives more weight to recently weakened knowledge, and α , β and γ are weighting coefficients set to 0.4, 0.4 and 0.2 respectively, determined through grid search on validation data:

$$R(u, r) = \alpha sim(u, r) + \beta KS(u, r) + \gamma W(t) \quad (3)$$

Every Monday, the engine selects the top five resources that students have rated most highly and presents them together with a brief reason, such as "Recommended because your understanding of loop structures is estimated to be 0.62."

4. Empirical design

In the autumn of 2025, I took the Fundamentals of Programming class, which is a mandatory first-year course at a comprehensive university in southwestern China. Two parallel classes were taught by the same teacher and had the same syllabus, textbooks, and tests. As shown in **Table 1**, there were 52 students in the experimental group and 50 in the control group; the two groups were comparable in terms of gender, previous programming experience, and entrance examination scores at the start. The study was conducted in accordance with the ethical guidelines of the institution. All data were anonymized, and informed consent was obtained from all participants prior to data collection.

Table 1. Baseline comparison between the experimental and control groups

Indicator	Experimental ($n = 52$)	Control ($n = 50$)	Test
Gender (male/female)	31 / 21	30 / 20	$\chi^2 = 0.004, P = 0.951$

Prior programming experience (%)	38.5	36.0	$\chi^2 = 0.071, P = 0.790$
Entrance score (mean $\pm$ SD)	76.3 $\pm$ 8.1	75.8 $\pm$ 8.6	$t = 0.30, P = 0.763$
Pre-test score (mean $\pm$ SD)	68.4 $\pm$ 9.2	67.9 $\pm$ 8.8	$t = 0.29, P = 0.774$

It was about a year. Based on the structure in Section 3, the experimental group received personalized material recommendations every week, and the control group was given the same set of materials from a single, fixed, and ordered list. Both groups attended the same classes and took the same weekly tests. The sources of the data are: LMS logs (approximately 187,000 events), 12 weekly quizzes, standardized pre- and post-tests, and system logs of recommendation impressions and clicks. Academic achievement is the post-test score (maximum of 100 points), behavior is quantified by the number of active learning behaviors per student per week, and algorithmic quality is shown as precision@5 on held-out clicks. Parallel forms of the pre-test and post-test were created by two experienced teachers to cover the same 46 knowledge points and had the same difficulty; the post-test had a Cronbach's alpha of 0.87 and showed good internal consistency. Weekly quizzes were used to gauge how well students had grasped the material and to monitor changes in their knowledge; finally, all students completed a five-point Likert scale questionnaire to rate their own satisfaction with the helpfulness and convenience of the recommended service. All statistical tests were carried out at the 0.05 level of significance, and the P -values are presented as effect sizes.

5. Results

5.1. Academic performance

Table 2 presents the pre-test and post-test results. There was no significant difference between the two groups in the pre-test, but the experimental group scored higher after the intervention (82.6 ± 7.4) than the control group (74.1 ± 8.1); that is, $t(100) = 5.63, P < 0.001$, and the effect size was relatively large (Cohen's $d = 1.10$). The mean gain of the experimental group was 14.2 points, about 2.3 times that in the control group (6.2 points), so RQ1 was positively answered.

Table 2. Comparison of test performance between the two groups

Measure	Experimental	Control	Statistics
Pre-test (mean $\pm$ SD)	68.4 $\pm$ 9.2	67.9 $\pm$ 8.8	$t = 0.29, P = 0.774$
Post-test (mean $\pm$ SD)	82.6 $\pm$ 7.4	74.1 $\pm$ 8.1	$t = 5.63, P < 0.001, d = 1.10$
Gain score	14.2 $\pm$ 6.8	6.2 $\pm$ 5.9	$t = 6.34, P < 0.001$

5.2. Learning behavior

Figure 2 shows the weekly development of active learning behavior. Both groups started at the same level in week 1, but the experimental group showed a continuous increase after the recommendation engine was stable in week 3, reaching 226 behaviors per student by week 12, and the control group only reached about 142 and then stopped growing. Repeated-measures ANOVA indicated a significant group-by-time interaction ($F = 21.47, P < 0.001$). Based on Behavioral Sequence Analysis, the experimental students have been able to complete more cycles of the “watch recommended video–attempt exercise–review feedback” loop, and their past efforts have led to deeper interest and self-regulated learning^[9]. The results answer RQ2.

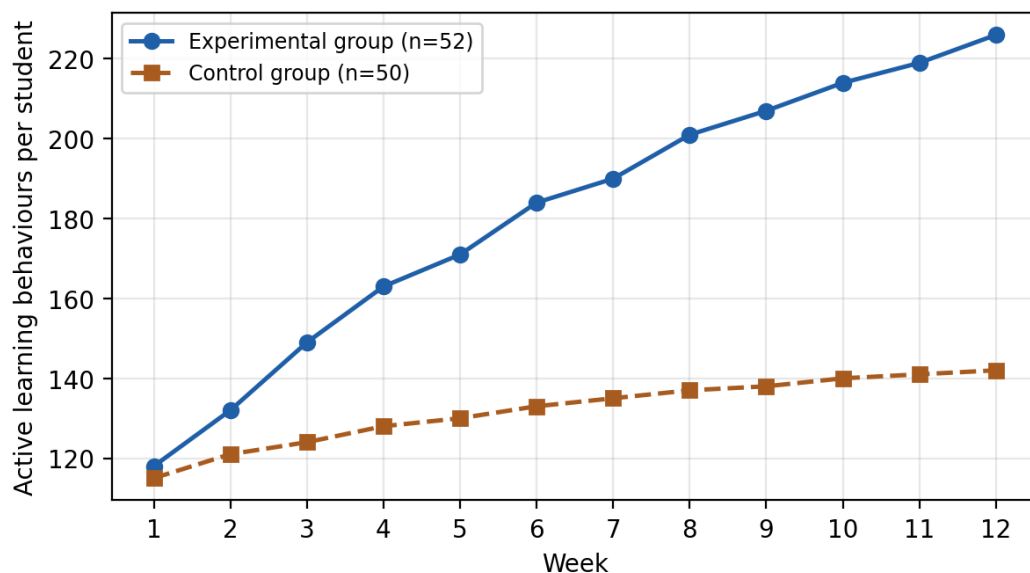


Figure 2. Weekly active learning behaviors of the two groups

5.3. Recommendation quality

RQ3: Compare the hybrid algorithm with the popularity-based baseline based on held-out click data from the past four weeks. The precision@5 and recall@5 of the hybrid engine were 0.78 and 0.46, respectively, and they surpassed the baseline values of 0.41 and 0.22. In addition, 71.3 percent of the pushed resources were for knowledge points with an estimated mastery of less than 0.70, so the engine focused on actual learning deficiencies rather than popular materials. The click-through rate of the personalized push was 64.8%, and the end-of-course questionnaire (5-point scale) showed that people found it quite useful (mean = 4.31) and easy to use (mean = 4.18).

6. Discussion

The improvement in this study is in line with the results of the meta-analysis that analytics-driven personalization can improve performance, and given that its effect size ($d = 1.10$) is relatively large compared with other studies on dashboards alone, this paper supports the idea that closing the loop between analytics and actionable advice is necessary. The behavior data are in line with the path analysis results of Gao *et al.*^[8] and the multi-modal results of Zhang *et al.*; that is, personalized prompts have expanded both the frequency and the order of study behavior through repeated application. BKT-based knowledge tracing with semantic and forgetting-aware scoring will be employed to extend the learning-path recommendation line proposed by Raj and Renumol and the neural adaptation method of Liu *et al.*, while still meeting the model interpretability requirements set by the university staff^[11].

Three indications of teaching practice follow. First, instructors of university computer courses should make weekly recommendations part of their teaching routine, not an optional extra; only after the engine has been running stably and some weeks have passed will any effect be observed. Second, the interpretability of the model is also a problem; instructors do not trust it and are thus unwilling to follow the gap-oriented explanations provided by Push. Third, the quality of the resources used by recommenders is relatively low;

thus, the course team should optimize the resource library first. Based on the research of learning analytics, achievement, behavior sequences, and algorithmic indicators can be combined in a single Design to build a replication model for other STEM courses.

7. Conclusion

We developed, implemented, and empirically tested a learning-analytics-driven personalized resource recommendation framework for a university computer course in this paper. The framework helped the students improve their grades, form some good study habits, and receive some targeted advice to address their problems. In the future, we will expand the framework to several courses, integrate knowledge graphs generated by large language models, and study the duration of the observed effect.

Funding

This work was supported by the Joint Special Fund for Basic Research of Local Undergraduate Universities in Yunnan Province (Grant No. 202101BA070001-200).

Disclosure statement

The authors declare no conflict of interest.

References

- [1] Khor ET, KM, 2024, A Systematic Review of the Role of Learning Analytics in Supporting Personalized Learning. *Education Sciences*, 14(1): 51. <https://doi.org/10.3390/educsci14010051>
- [2] Liu N, Li Y, Guo Y, 2024, Optimization of Online Learning Resource Adaptation in Higher Education Through Neural Network Approaches. *International Journal of Interactive Mobile Technologies*, 18(11): 92–107. <https://doi.org/10.3991/ijim.v18i11.49779>
- [3] Raj NS, Renumol VG, 2024, An Improved Adaptive Learning Path Recommendation Model Driven by Real-Time Learning Analytics. *Journal of Computers in Education*, 11(1): 121–148. <https://doi.org/10.1007/s40692-022-00250-v>
- [4] Narimani A, Barbera E, 2024, Extracting Course Features and Learner Profiling for Course Recommendation Systems: A Comprehensive Literature Review. *International Review of Research in Open and Distributed Learning*, 25(1): 197–225.
- [5] Alfredo R, Echeverria V, Jin Y, et al., 2024, Human-Centred Learning Analytics and AI in Education: A Systematic Literature Review. *Computers and Education: Artificial Intelligence*, 6: 100215. <https://doi.org/10.1016/j.caeai.2024.100215>
- [6] Chen X, Yin C, Chen H, et al., 2024, Course Recommendation System Based on Course Knowledge Graph Generated by Large Language Models, *Proceedings of the 2024 IEEE International Conference on Teaching, Assessment and Learning for Engineering (TALE)*. IEEE, 1–8.
- [7] Wang H, 2025, Leveraging Machine Learning for Personalized Learning, Automated Feedback, and Predictive Analytics in College English Education. *International Journal of Information and Communication Technology*,

26(16): 16–37. <https://doi.org/10.1504/IJICT.2025.146375>

- [8] Gao Z, Yan H, Liu J, et al., 2025, Tracing Distinct Learning Trajectories in Introductory Programming Course: A Sequence Analysis of Score, Engagement, and Code Metrics for Novice Computer Science vs. Math Cohorts. *International Journal of STEM Education*, 12. <https://doi.org/10.1186/s40594-025-00546-2>
- [9] Zhang L, Wang X, Chen W, et al., 2026, Using Multimodal Learning Analytics to Understand the Combined Effects of Dynamic Self-Regulated Scaffoldings and Learning from Examples on Computer Programming: An Empirical Study. *International Journal of STEM Education*, 13: 3. <https://doi.org/10.1186/s40594-025-00591-x>
- [10] Chen G, Song T, Wang Q, et al., 2025, Knowledge Graph and Large Language Model Integration with Focus on Educational Applications: A Survey. *Neurocomputing*, 2025: 131230.
- [11] Konstantinidou A, Nisiforou EA, Vrasidas C, 2026, Enhancing Personalized Learning with Artificial Intelligence and Analytics: University Staff's Insights. *Interactive Learning Environments*, 2026: 1–21. <https://doi.org/10.1080/10494820.2026.2619907>

Publisher's note

Bio-Byword Scientific Publishing remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.