PersonaPath: Towards Knowledge-Centric Personalized Learning Path Planning

Yu Liu[1], Zeming Liu[1†], Tianle Zhang[1], Zihao Cheng[1], Yuhang Guo[2], Kehai Chen[3], Min Zhang[3], Yunhong Wang[1], Haifeng Wang[4]

[1]School of Computer Science and Engineering, Beihang University, Beijing, China
[2]Beijing Institute of Technology [3]Harbin Institute of Technology (Shenzhen) [4]Baidu Inc.
[†]Corresponding author Email: liuyuu@buaa.edu.cn, zmliu@buaa.edu.cn

Abstract

Adaptive learning systems commonly formulate learning path planning as Exercise-Centric (EC) recommendation, where the next step is inferred from item-level interaction logs. Evaluating goal-oriented guidance additionally requires explicit learner goals and curriculum-scale prerequisites: learners with similar exercise records may need different paths toward their targets. We therefore study Knowledge-Centric (KC) personalized learning path planning, where a planner must reason over learner profiles, mastery states, and prerequisite knowledge structures to decide which textbook, unit, and concept should be studied next. To support this setting, we introduce **PersonaPath**, a benchmark that pairs 2,000 fine-grained learner personas with a hierarchical knowledge graph of 347 textbooks, 1,751 units, and 4,092 concepts across 77 subjects. We evaluate representative LLMs on PersonaPath. Results show that even the strongest LLM reaches only a 29.5% final pass rate in Basic Education, and that the main bottleneck lies in adaptivity, where no model exceeds 44.7% in tailoring paths to individual learners[1].

1 Introduction

Learning path planning decides what a learner should study next and in what order (Li et al., 2024; Tu et al., 2025; Hedi et al., 2025). By adapting this sequence to each learner's current knowledge and goals, it makes study more efficient and better personalized (Li et al., 2023). Previous methods typically follow an Exercise-Centric (EC) recommendation paradigm (Liu et al., 2019; Zhang et al., 2024), modeling each learner from item-level logs, such as which exercises were attempted and answered correctly, and recommending the next exercise to practice.

Interaction content supports learner-state estimation and sequential recommendation, while explicit goals specify which knowledge a path should ultimately reach. For example, in Figure 1A, two students attempt the same exercises and produce identical correctness records. Both solve the vector-coordinate item (E1) but miss vector addition (E2) and dot product (E3). A recommendation based on this record alone can assign them the same sequence of vector decomposition (E4), component-wise addition (E5), and component-wise dot product (E6). Yet their goals diverge, with Student A targeting straight-line equations and Student B targeting vector concepts. The illustrated sequence follows Student B's goal, whereas Student A needs a path connecting her current mastery to straight-line equations. Evaluating this distinction requires the learner's target and the prerequisite structure connecting it to the current knowledge state.

To bring each learner's individual needs into focus, we propose a Knowledge-Centric (KC) paradigm, which decides what knowledge a learner should study and in what order. As shown in Figure 1B, for two different learners, a KC planner reads each one's progress and goal, selects the knowledge they need from a structured knowledge space, and orders it into a learning path that respects the prerequisite relations among concepts. Planning thus shifts from which exercise to practice next to which knowledge to study next, and why. Existing benchmarks primarily support exercise recommendation and student performance prediction. They rarely jointly provide explicit learner personas, long-term goals, and curriculum-scale prerequisite structures. Evaluating KC planning requires these components together: the goal determines the destination, the mastery state identifies the learner's starting point, and the prerequisite graph constrains the routes between them.

To evaluate Knowledge-Centric personalized

[1]The data and the code are available at https://github.com/BUAA-IRIP-LLM/PersonaPath.

Proceedings of the 15th International Joint Conference on Natural Language Processing and the 5th Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics, pages 3387–3423
November 6-10, 2026 ©2026 Association for Computational Linguistics

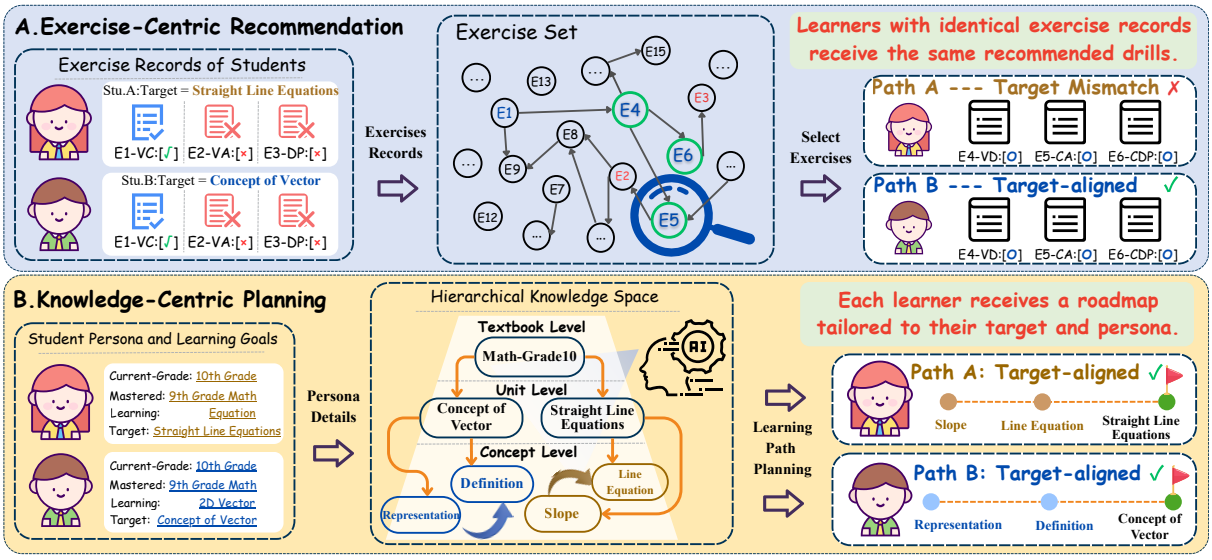


Figure 1: **Illustration of Exercise-Centric (EC) recommendation and Knowledge-Centric (KC) planning.** Panel A illustrates recommendations conditioned on correctness records alone; Panel B additionally uses explicit learner goals and mastery states. Each node (E1, E2, ...) represents an exercise item, and edges denote prerequisite relations among the knowledge points. Abbreviations: VC = Vector Coordinates, VA = Vector Addition, DP = Dot Product, VD = Vector Decomposition, CA = Component-wise Addition, and CDP = Component-wise Dot Product.

learning path planning, we propose **PersonaPath**. It contains 2,000 fine-grained learner personas from primary school to higher education. The personas are grounded in a hierarchical knowledge space of 347 textbooks, 1,751 units, and 4,092 concepts across 77 subjects, connected by prerequisite dependencies. Given a learner profile and a target unit, a model must generate a path that respects prerequisites, adapts to prior knowledge, and efficiently reaches the goal. We benchmark representative large language models on PersonaPath and find that they still struggle with KC planning, especially in adapting paths to individual learners. PersonaPath thus advances personalized learning path planning toward a knowledge-centric paradigm, helping educational systems better address each learner's individual needs.

This work makes three contributions:

- We identify an evaluation gap in personalized learning: existing benchmarks rarely combine explicit learner personas, long-term goals, and curriculum-scale prerequisites for assessing goal-directed knowledge paths.
- To address this, we propose KC path planning and construct **PersonaPath**, a benchmark that integrates learner personas, target goals, and prerequisite curriculum structures to evaluate LLMs on KC tasks.
- We evaluate representative baselines on PersonaPath and show that LLMs still struggle with KC planning, especially in personalized adaptation. Ablations examine how mastery information, contextual noise, and step-by-step feedback affect path quality.

2 Related Work

2.1 Learning Path Recommendation

Most Learning Path Recommendation (LPR) studies (Zhang et al., 2021) operationalize personalized learning path planning through the Exercise-Centric (EC) paradigm, where paths are inferred from item-level interaction content, such as exercise attempts, correctness records, hints, and response sequences (Liu et al., 2019; Zhang et al., 2024). Existing approaches generally fall into two categories: similarity-based heuristics and Deep Reinforcement Learning (DRL) frameworks. Traditional methods often treat LPR as a combinatorial optimization problem, using techniques such as collaborative filtering (Yu et al., 2018) or meta-heuristic search, including Ant Colony Optimization (Niknam and Thulasiraman, 2020), immune algorithms (Bian et al., 2019), and genetic algorithms (Elshani and Nuçi, 2021), to discover paths based on similar learner cohorts. More recently, DRL-based frameworks have become a prominent line of work (Li et al., 2023), viewing learning as

Benchmark	Attribute	Task	TPG	PSN	BED	HKS	RS
Junyi Academy (Chang et al., 2015)	Exercise-Centric	Question Recommendation	✗	✗	⚔	✗	✗
ASSISTments2009 (Feng et al., 2009)	Exercise-Centric	Question Recommendation	✗	✗	⚔	✗	✗
ASSISTments2012 (Wang et al., 2015)	Exercise-Centric	Question Recommendation	✗	⚔	⚔	✗	✗
OLI Engineering Statics (University, 2011)	Exercise-Centric	Question Recommendation	✗	✗	⚔	✗	✗
Synthetic Data from DKT (Piech et al., 2015)	Exercise-Centric	Question Recommendation	✗	✗	⚔	✗	✗
PersonaPath (Ours)	**Knowledge-Centric**	**Learning Path Planning**	✓	✓	✓	✓	✓

Table 1: Comparison between PersonaPath and other benchmarks. "TPG", "PSN", "BED", "HKS", and "RS" represent Textbook-level Prerequisite Graph, Persona, Breadth of Educational Disciplines, Hierarchical Knowledge Structure, and Rich Semantics, respectively. "✗" indicates the absence of a dimension, while "⚔" indicates that only part of it is included.

a sequential decision-making process. These models use architectures such as RNNs or graph-based agents to maximize a cumulative reward defined by the learner's performance on subsequent interactions (Zhou et al., 2018; Zhang et al., 2024). Such approaches already incorporate learner states and sequential decisions, including goal-oriented recommendation over graph structures (Li et al., 2023; Liu et al., 2023).

Related benchmarks outside education evaluate dependency-aware API planning, preference-sensitive tool selection, and real-world travel planning (Wang et al., 2024; Cheng et al., 2025; Deng et al., 2025). PersonaPath brings these concerns into curriculum planning, where actions must respect prerequisite knowledge and adapt to learner states.

Knowledge Tracing (KT) estimates and updates mastery from response sequences (Corbett and Anderson, 1995; Piech et al., 2015), while Cognitive Diagnosis (CD) infers concept-level proficiency from response data (Cheng et al., 2019). These methods provide learner-state estimates that can inform a downstream planner. PersonaPath supplies the mastery state as an input and evaluates the resulting prerequisite-aware path toward an explicit target. Its evaluation focus is the selection and ordering of knowledge given that state. Appendix D discusses the connection in more detail.

2.2 Evaluation Benchmarks and Datasets

Standard benchmarks in this domain primarily consist of large-scale interaction logs recorded from online learning platforms, including the **ASSISTments** family (Feng et al., 2009; Wang et al., 2015), **Junyi Academy** (Chang et al., 2015), and **OLI Engineering Statics** (University, 2011), together with synthetic response data generated for Deep Knowledge Tracing (Piech et al., 2015). These resources support student performance prediction and learner-state modeling from item-level records. We describe each dataset in detail in Appendix A.2. As summarized in Table 1, the gap for KC evaluation is the joint availability of explicit learner personas, long-term goals, and curriculum-scale prerequisites. Skill labels and interaction sequences provide useful evidence about mastery; evaluating a goal-directed path additionally requires a structured curriculum connecting the current state to the target. PersonaPath pairs learner personas and goals with a textbook–unit–concept hierarchy and prerequisite relations to support this evaluation.

3 PersonaPath

3.1 Task Definition

PersonaPath evaluates Knowledge-Centric personalized learning path planning. Given a learner profile, a mastery state, a target unit, and a hierarchical knowledge space, the model must generate a sequence of knowledge concepts that helps the learner reach the target while respecting curriculum prerequisites. The task evaluates how a planner uses the supplied knowledge structure and learner state to select its next action.

Knowledge Space Definition. We define the knowledge space as a Hierarchical Knowledge Graph, denoted by $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{H})$. The vertex set $\mathcal{V}$ is hierarchically partitioned into three levels: Textbooks $\mathcal{B} = \{b_1, ..., b_K\}$, Units $\mathcal{U} = \{u_1, ..., u_M\}$, and Atomic Concepts $\mathcal{C} = \{c_1, ..., c_N\}$. The edge set $\mathcal{E} \subseteq \mathcal{B} \times \mathcal{B}$ defines the directed prerequisite order between textbooks. The set $\mathcal{H}$ encodes the vertical inclusion structure, where each atomic concept c belongs to a unique unit u, and each unit u belongs to a unique textbook b. Prerequisite relations at the unit and concept levels are induced from $\mathcal{E}$ via $\mathcal{H}$.

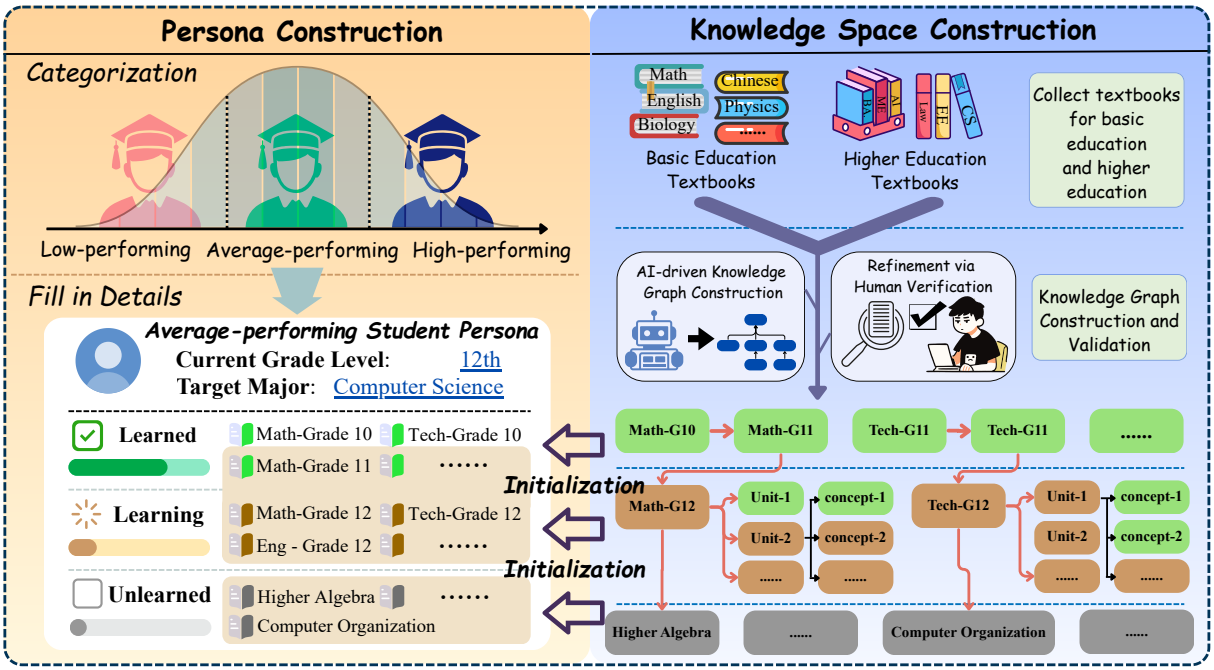


Figure 2: The construction pipeline of **PersonaPath**. The benchmark first builds explicit learner personas and a hierarchical knowledge space, then initializes each learner's mastery state so that KC planners can be evaluated on goal-oriented, prerequisite-aware, and persona-adaptive learning paths.

Cognitive State and Knowledge Acquisition. We distinguish between the static attributes of knowledge and the dynamic cognitive state of the learner. Let $D : \mathcal{C} \to [0, 1]$ denote the inherent difficulty function associated with atomic concepts. Conversely, learner proficiency is tracked via a dynamic mastery vector $\mathbf{M}_t \in [0, 1]^{|\mathcal{U}|}$, defined at the unit level at time step t. The planning process operates as follows: at each step t, given the learner's state $\mathcal{S}_t = (\mathbf{P}, \mathbf{M}_t)$ (where $\mathbf{P}$ represents the static persona profile), the agent selects a knowledge concept $a_t \in \mathcal{C}$. The environment then simulates the cognitive update of the corresponding parent unit u via a transition function (see Appendix A.1 for details of the calculation): $\mathbf{M}_{t+1}[u] \leftarrow \text{Sim}(\mathbf{M}_t[u], D(a_t))$. A detailed explanation of the hierarchical transition mechanism is provided in Appendix A.5. Thus, the action and state have different granularities: concepts are the selectable skills, while units are the coordinates of the mastery vector. Concepts within a unit share its mastery value but retain their individual difficulty values. The updated unit mastery conditions the next concept selection. This representation tracks 1,751 unit states over a planning space of 4,092 concepts.

Planning Objective. The goal is to generate an optimal learning path $\mathcal{P} = [a_1, a_2, ..., a_T]$ for a specified target unit $g_{target} \in \mathcal{U}$. The process terminates when the learner's mastery of the target unit meets the educational standard: $\mathbf{M}_T[g_{target}] \geq \tau$, where τ is a predefined proficiency threshold.

3.2 Data Construction

The data construction pipeline is designed around the three inputs required by KC planning: explicit learner goals, prerequisite knowledge structure, and learner-specific mastery states. As illustrated in Figure 2, the pipeline consists of *Persona Construction*, *Knowledge Space Construction*, and *Persona Initialization*. These components support evaluating whether a model selects appropriate content for a learner at each step toward the target.

Persona Construction. Following the synthetic student modeling approach of Piech et al. (2015), we define a learner persona by a specific academic goal (e.g., a target major at university) and a structured knowledge mastery state. This design makes the planning target explicit: the model must decide how to bridge the gap between the learner's current state and the target knowledge. To simulate varied learning abilities, we categorize learners into *Low-performing*, *Average-performing*, and *High-performing* archetypes in an approximate 1:3:1 ratio, which corresponds to tertile bins of the standard normal ability distribution commonly used in Item Response Theory (Lord, 2012). These

archetypes govern the learner's simulated learning efficiency and difficulty tolerance, enabling the benchmark to evaluate planning performance across different learner profiles. Furthermore, we position each learner persona at a specific *Current Grade Level*, which splits the curriculum into three regions: *Learned* (already mastered), *Learning* (under active study), and *Unlearned* (next objectives). This grade anchor fixes both the planning horizon and the learner's initial knowledge state.

Knowledge Space Construction. To provide the prerequisite structure required by KC planning, we built a knowledge base from authoritative textbooks: Basic Education textbooks from *People's Education Press*[2] and Higher Education resources from the *Smart Education of China* platform[3]. Textbooks were ordered by subject following the standard teaching sequence for Basic Education and the internal learning path of each major for Higher Education, with Basic Education subjects serving as prerequisites for related Higher Education majors (e.g., Chemistry and Biology before Clinical Medicine). To transform these textbooks into a structured Knowledge Graph, we adopted a hybrid human-AI workflow. We used DeepSeek-V3 as the extraction engine because of its performance on entity and relation extraction (Zhao et al., 2025; Xu et al., 2024) and its proficiency in Chinese educational text. Following a hierarchical extraction strategy (Chen et al., 2025b; Lu et al., 2019), the model first parsed the tables of contents to build the structural hierarchy and then identified atomic concepts within each unit to form a fine-grained Knowledge Graph. Domain experts manually reviewed the output to correct structural errors and extraction inaccuracies.

Persona Initialization. We initialize mastery values so that each persona becomes an actionable planning state rather than a static user description. For *Learned* content, mastery scores are assigned according to the persona's proficiency archetype (e.g., *Low-performing* or *High-performing*), with distribution details in Appendix A. For *Learning* content, mastery levels are randomly initialized within the interval $(0, 1)$. To simulate the cumulative nature of learning, we implement a dynamic knowledge transfer mechanism within the test environment. When a learner moves to a new textbook, the initial mastery for upcoming concepts is not zero but is derived from the proficiency levels attained in the corresponding prerequisite subjects.

[2]https://www.pep.com.cn/

[3]https://smartedu.tbook.com.cn/

3.3 Quality Control

Quality control focuses on the elements that determine whether KC paths can be evaluated reliably: prerequisite validity, entity correctness, and persona consistency. For prerequisite relations, we combined automated and manual checks: graph algorithms inspected the Directed Acyclic Graph (DAG) structure for circular dependencies, following the precedence axioms in Knowledge Space Theory (Doignon and Falmagne, 2012). A panel of two domain experts also reviewed the pedagogical logic of the links and verified the semantic accuracy of the LLM-generated entities. To assess consistency, we calculated Inter-Annotator Agreement (IAA) on a validation subset reviewed by both experts, using Cohen's Kappa (Cohen, 1960). To improve the coherence of initialized personas, we implemented sanity checks to avoid logical inconsistencies. These constraints verify that (1) *High-performing* personas have correspondingly high mastery scores in previously *Learned* content and (2) a persona positioned at a higher *Current Grade Level* has completed all curriculum requirements from preceding educational stages.

Expert Evaluation of Personas. Two experienced educators independently evaluated 100 randomly sampled personas using a 5-point Likert scale. They assessed four dimensions: curriculum consistency, progression plausibility, mastery plausibility, and persona coherence. The study evaluates whether the generated learner profiles and initialized mastery states form pedagogically plausible, internally coherent representations. We report the mean rating for each dimension and inter-rater reliability using ICC(2,2), the two-way random-effects, absolute-agreement coefficient for the average of the two raters.

3.4 Quality and Data Statistics

We quantitatively assessed the structural integrity and logical validity of the **PersonaPath** dataset. Regarding prerequisite relations, structural analysis verified strict DAG compliance across **347** textbook nodes and **411** prerequisite edges, confirming zero circular dependencies. Manual verification on the validation subset yielded a Cohen's Kappa of **0.93**, which falls within the "almost perfect" agreement range (Landis and Koch, 1977), indicating strong expert consensus. After expert cor-

A: Textbook Statistics				
Stage	Subject	Books	Units	Concepts
Primary	8	71	416	930
Middle	12	58	286	680
High	12	78	344	881
University	45	140	705	1,601
Total	**77**	**347**	**1,751**	**4,092**

B: Persona Statistics		
Category	Basic Education	Higher Education
Low-performing	193	200
Average-performing	620	609
High-performing	187	191
Total	**1,000**	**1,000**

Table 2: Overview of the benchmark statistics.

Dimension	Mean / 5	ICC(2,2)
Curriculum consistency	4.68	0.85
Progression plausibility	4.37	0.82
Mastery plausibility	4.11	0.74
Persona coherence	4.33	0.79

Table 3: Expert evaluation of 100 randomly sampled personas. Two educators independently rated each persona on a 5-point Likert scale. ICC(2,2) measures agreement for their average rating.

rection, the verified prerequisite relations reached a precision of **99.5**%. For content correctness, the evaluation confirmed the semantic accuracy of the LLM-generated entities, ensuring a reliable knowledge environment. Initialized personas are filtered by two logical sanity rules: (1) *High-performing* personas must hold mastery scores above **0.85** in learned content; (2) personas placed at a higher *Current Grade Level* must satisfy chronological consistency (no missing prerequisites). The released personas satisfy these rules by construction.

Table 3 reports the expert evaluation of the sampled personas. Mean ratings range from 4.11 to 4.68 out of 5, with ICC(2,2) values from 0.74 to 0.85. Curriculum consistency receives the highest rating (4.68), while mastery plausibility has the lowest mean (4.11) and inter-rater reliability (0.74). These judgments support the pedagogical plausibility and internal coherence of the synthetic profiles.

Table 2 presents the statistical breakdown of the benchmark's two core components. Part A depicts a vertically structured knowledge graph where the number of concepts grows with educational progression, expanding from **930** concepts at the Primary level to **1,601** at the University level. Part B summarizes a cohort of **2,000** learners, evenly balanced between Basic and Higher Education. These learners are stratified into *Low-performing*, *Average-performing*, and *High-performing* archetypes in an approximate **1:3:1** ratio, supporting the evaluation of personalization across different learner proficiency levels.

4 Experiments

4.1 Experimental Setup

Baselines. We evaluate ten open-source LLMs covering different architectures and parameter scales (1B to 30B+). The evaluation set includes: (1) the **Pangu series** (Chen et al., 2025a) (1B, 7B); (2) the **Llama series** (Grattafiori et al., 2024) (3.2-1B, 3.1-8B); (3) the **Qwen series** (Yang et al., 2025) (Qwen3-4B-2507, Qwen3-30B-A3B); (4) the **DeepSeek series** (Guo et al., 2025), comprising the reasoning-optimized R1-Distill models (7B, 14B) and DeepSeek-V3.1; and (5) **InnoSpark** (Song et al., 2025), specifically InnoSpark-7B, an education-focused model whose training includes supervised fine-tuning and multi-stage reinforcement learning. To examine whether reasoning chains help, we evaluate these models under both *Zero-shot* and *Chain-of-Thought* (CoT) settings (Wei et al., 2022).

Interactive Planning and Retrieval. The main evaluation uses a closed-loop protocol: each concept selection is followed by a simulated mastery update, and the updated state is supplied for the next decision (Figure F.6). We also evaluate a retrieval-augmented Qwen3-4B-2507$_{Zero-shot}$ on Basic Education. It retrieves the top-3 knowledge chunks using `BAAI/bge-small-zh-v1.5` embeddings and FAISS, then prepends them to the planning prompt. Table 5 reports the results; Appendix B.4 gives the retrieval configuration.

Implementation Details. Experiments were conducted on a high-performance computing cluster with Huawei Ascend 910B1 NPUs (64 GB HBM) and Kunpeng-920 CPUs (192 cores, AArch64 architecture), running Huawei Cloud EulerOS 2.0. We used the `vLLM` library (v0.9.2) for inference, with all models loaded in bfloat16 precision. We set the sampling temperature to 0.1 and the maximum generation length to 32,768 tokens, while retaining the default vLLM settings for other parameters.

4.2 Evaluation Metrics

We evaluate each path along three dimensions. *Validity* contains *Prerequisite Violation* (no concept appears before its antecedents, per Knowledge Space Theory (Doignon and Falmagne, 2012)) and *Hallucination* (no entity falls outside the knowledge space).

Adaptivity contains *Persona Alignment* (consistency with the learner's archetype and prior experience, motivated by the Zone of Proximal Development (Vygotsky, 1978)) and *Difficulty Adaptability*. We measure the latter using Cog-Gap (Zhang et al., 2024), the average absolute difference between mastery and concept difficulty along the path:

$$\text{Cog-Gap} = \frac{1}{n}\sum_{t=1}^{n} \left|\mathbf{M}_t[u] - D(a_t)\right|. \quad (1)$$

Here, n is the number of evaluated concept selections, and u denotes the parent unit of the selected concept a_t at each step. The unit mastery $\mathbf{M}_t[u]$ and concept difficulty $D(a_t)$ both lie in $[0, 1]$, so Cog-Gap also lies in $[0, 1]$. A smaller value indicates closer alignment between the learner's current mastery and the selected difficulty. The absolute difference captures both under-challenging and over-challenging choices. Because mastery is updated during a path, the same concept difficulty can yield different gaps at different steps. This metric complements prerequisite validity by assessing whether the selected content fits the learner's current state.

Efficiency contains *Goal Completion* (reaching target mastery within a step budget $T_{\max}$) and *Progress Continuity* (the average proficiency increment over a five-step sliding window must exceed a threshold). A path receives a *Final Pass* only when all three hold, $Pass_{final} = \mathbb{I}(\text{Validity}) \cdot \mathbb{I}(\text{Adaptivity}) \cdot \mathbb{I}(\text{Efficiency})$. Full definitions and thresholds are in Appendix B.1.

4.3 Results

Table 4 presents the performance of various LLMs across different learning stages and prompt types on PersonaPath. Three findings emerge.

Performance drops consistently from Basic to Higher Education across almost every metric. Notably, the Final Pass Rate of the top-performing DeepSeek-V3.1 drops by approximately half, falling from 29.5% to just 14.6%. The drop appears across all model scales: the runner-up Qwen3-30B-A3B (CoT) declines from 27.7% to 10.8%, while the lightweight Qwen3-4B-2507 (CoT) suffers an even steeper drop from 20.8% to 5.5%. This gap suggests that these models may have been exposed to less university-level educational data during training and highlights the difficulty of handling the escalated structural complexity in higher education planning.

Performance varies significantly across model series. Llama-3.2-1B and Llama-3.1-8B achieve Final Pass Rates of at most 1.3% across Basic and Higher Education. These scores describe performance on the Chinese textbooks and curriculum structures used by PersonaPath. Section 5 examines changes within each model under the same language and curriculum conditions. In addition, InnoSpark-7B, a model specialized for education, shows a distinct "specialist" profile: it outperforms similar-sized baselines in *Validity* with a 41.3% pass rate, yet lags behind the same counterparts in *Adaptivity*. This pattern suggests that domain specialization may skew the model toward curriculum-aligned outputs at the cost of the learner-conditioned reasoning required for Adaptivity.

Models show a sharp imbalance across constraints: they excel in Validity but struggle with Adaptivity. Specifically, models perform relatively well in *Validity*, exemplified by DeepSeek-V3.1's 90.9% pass rate, which suggests that current LLMs can often structure logically valid teaching schedules under the provided constraints. Conversely, *Adaptivity* remains the primary bottleneck: even the top-performing Qwen3-30B-A3B (CoT) achieves only 44.7%, while DeepSeek-V3.1 reaches 44.3%. This reveals a gap in personalization: models handle general pedagogical rules well but struggle to tailor decisions to specific learner profiles.

Effects of CoT Prompting. CoT changes performance differently across models and educational stages (Table 4). In Basic Education, Qwen3-30B-A3B's Final Pass rises from 10.0% under Zero-shot prompting to 27.7% with CoT, and Qwen3-4B-2507's rises from 3.2% to 20.8%. Their Adaptivity scores increase from 21.9% to 44.7% and from 12.5% to 43.9%, respectively. Thus, the gains for these Qwen models include better alignment with learner states as well as higher joint pass rates.

The improvements are smaller in Higher Education: Final Pass rises from 8.6% to 10.8% for Qwen3-30B-A3B and from 2.4% to 5.5% for Qwen3-4B-2507. Llama-3.1-8B shows a different response in Basic Education, with Validity declin-

	Basic Education (Test#1000)				**Higher Education** (Test#1000)			
	Validity Pass Rate	Adaptivity Pass Rate	Efficiency Pass Rate	Final Pass Rate	Validity Pass Rate	Adaptivity Pass Rate	Efficiency Pass Rate	Final Pass Rate
Pangu-1B$_{CoT}$	10.1	11.4	20.4	1.1	6.4	13.6	10.8	1.0
Llama-3.2-1B$_{Zero-shot}$	3.1	13.6	11.5	0.3	4.4	11.8	6.0	0.9
Llama-3.2-1B$_{CoT}$	3.4	9.0	9.0	0.7	0.6	2.7	1.6	0.0
Qwen3-4B-2507$_{Zero-shot}$	39.7	12.5	25.8	3.2	8.6	13.5	6.5	2.4
Qwen3-4B-2507$_{CoT}$	55.8	43.9	60.2	20.8	27.6	24.2	31.1	5.5
Pangu-7B$_{CoT}$	29.9	28.5	46.4	15.0	24.6	31.9	41.3	5.4
InnoSpark-7B$_{Zero-shot}$	41.1	11.8	23.8	2.4	19.8	16.0	16.7	2.2
InnoSpark-7B$_{CoT}$	41.3	10.7	24.2	2.9	13.4	17.4	5.0	2.0
DeepSeek-R1-Distill-7B$_{Zero-shot}$	9.8	20.2	26.6	2.5	3.8	8.6	6.1	1.1
DeepSeek-R1-Distill-7B$_{CoT}$	9.9	28.2	30.6	2.5	8.5	16.2	12.0	1.9
Llama-3.1-8B$_{Zero-shot}$	19.2	11.0	14.1	0.9	6.1	15.4	8.0	0.9
Llama-3.1-8B$_{CoT}$	14.6	6.6	12.6	0.8	2.3	12.0	2.9	0.4
DeepSeek-R1-Distill-14B$_{Zero-shot}$	42.3	34.0	53.9	10.8	35.9	27.6	40.0	4.3
DeepSeek-R1-Distill-14B$_{CoT}$	49.9	43.4	57.1	20.4	36.3	34.1	41.9	7.3
Qwen3-30B-A3B$_{Zero-shot}$	63.5	21.9	48.9	10.0	38.3	34.0	41.5	8.6
Qwen3-30B-A3B$_{CoT}$	70.8	44.7	59.4	27.7	44.1	33.5	46.0	10.8
DeepSeek-V3.1$_{Zero-shot}$	90.9	44.3	68.1	29.5	57.9	42.5	51.3	14.6

Table 4: **LLM performance on PersonaPath across Basic and Higher Education settings.** Results are reported as pass rates in percentage (%). **Basic Education** and **Higher Education** denote the two evaluation splits. The best, second-best, and third-best results are marked purple, orange, and gray, respectively.

ing from 19.2% to 14.6% and Final Pass from 1.3% to 0.8%. These within-model comparisons show that the effect of CoT depends on the model and curriculum stage. Reporting both prompting settings captures this variation across Validity, Adaptivity, and the joint planning objective.

Setting	Validity	Adaptivity	Efficiency	Final
Zero-shot	39.7	12.5	25.8	3.2
+ RAG	73.8	19.3	18.4	2.9
Δ	+34.1	+6.8	−7.4	−0.3

Table 5: Retrieval augmentation for Qwen3-4B-2507 under zero-shot prompting on 1,000 Basic Education personas. Pass rates are percentages; Δ is the change in percentage points from Zero-shot to Zero-shot + RAG.

Effects of Retrieval Augmentation. Table 5 compares Qwen3-4B-2507$_{Zero-shot}$ with its RAG variant on the same Basic Education cohort. RAG raises Validity from 39.7% to 73.8%, a gain of 34.1 percentage points, and Adaptivity from 12.5% to 19.3%. Efficiency decreases from 25.8% to 18.4%, while Final Pass changes from 3.2% to 2.9%. Retrieved context therefore improves the rate of structurally valid paths in this setting, while adaptation and efficient goal completion remain more restrictive. Since Final Pass requires all three dimensions to hold for the same path, the increase in Validity alone does not yield a higher joint pass rate. This comparison separates the benefit of supplying relevant knowledge from the overall quality of the resulting learning sequence.

The two input strategies emphasize different dimensions for the same base model. In Basic Education, Qwen3-4B-2507 with RAG obtains higher Validity than its CoT setting (73.8% versus 55.8%). CoT achieves higher Adaptivity (43.9% versus 19.3%) and Efficiency (60.2% versus 18.4%), yielding a Final Pass of 20.8% compared with 2.9% for RAG. The highest Validity and highest joint pass rate therefore occur under different settings. Evaluating both prompting and retrieval across all three dimensions captures the learner adaptation and progress requirements that a curriculum-consistency score alone would miss.

5 Analysis

We investigate three research questions through within-model comparisons under the same language and curriculum conditions. **RQ1** (Sec. 5.1) tests how explicit mastery information affects personalization. **RQ2** (Sec. 5.2) evaluates whether models can focus on the relevant curriculum structure when noisy resources are present. **RQ3** (Sec. 5.3) compares step-by-step interaction with generating a static curriculum in one pass.

5.1 Necessity of Explicit Knowledge State Modeling

We constructed a comparative setting by removing the *Mastery* field from the learner persona while keeping all other configurations constant (prompt template in Figure G.26). This ablation turns the input into a coarse learner description. The model still sees the learning goal and curriculum context, but no longer knows which units the learner has actually mastered. It therefore isolates whether explicit knowledge-state modeling contributes to KC planning beyond generic curriculum sequencing.

Model	Validity Pass Rate	Adaptivity Pass Rate	Efficiency Pass Rate	Final Pass Rate
Qwen3-4B-2507$_{CoT}$	56.0(↑ 0.2)	21.1(↓ 22.8)	50.9(↓ 9.3)	7.4(↓ 13.4)
Pangu-7B$_{CoT}$	22.7(↓ 7.2)	16.8(↓ 11.7)	45.1(↓ 1.3)	1.4(↓ 13.6)
DeepSeek-R1-Distill-14B$_{CoT}$	49.8(↓ 0.1)	26.3(↓ 17.1)	56.2(↓ 0.9)	10.0(↓ 10.4)
Qwen3-30B-A3B$_{CoT}$	67.8(↓ 3.0)	18.6(↓ 26.1)	53.0(↓ 6.4)	8.5(↓ 19.2)

Table 6: Performance of LLMs on PersonaPath (Basic Education) with coarse-grained profiles that omit knowledge mastery.

Table 6 presents the results for Basic Education; Higher Education results are in Appendix B. Removing mastery information reduces Adaptivity by 26.1 percentage points for Qwen3-30B-A3B (CoT), while Validity changes much less. With the goal and curriculum held fixed, the larger change in Adaptivity shows the role of the supplied mastery state in learner-specific content selection. The model needs this information to decide what a learner should skip, review, or study next.

5.2 Robustness Against Contextual Noise

In the main experiments, the candidate pool contains only domain-relevant textbooks. However, real-world retrieval systems can introduce noisy, out-of-domain items. We therefore designed a controlled noise-injection experiment in which irrelevant textbook titles were added at varying proportions. Table 7 presents performance under this noisy setting in Basic Education; Higher Education results are in Appendix B.

Model	Validity Pass Rate	Adaptivity Pass Rate	Efficiency Pass Rate	Final Pass Rate
Qwen3-4B-2507$_{CoT}$	12.6(↓ 43.2)	34.3(↓ 9.6)	37.4(↓ 22.8)	3.5(↓ 17.3)
Pangu-7B$_{CoT}$	2.7(↓ 27.2)	27.7(↓ 0.8)	22.5(↓ 23.9)	1.8(↓ 13.2)
DeepSeek-R1-Distill-14B$_{CoT}$	12.5(↓ 37.4)	25.6(↓ 17.8)	29.1(↓ 28.0)	4.5(↓ 15.9)
Qwen3-30B-A3B$_{CoT}$	15.9(↓ 54.9)	25.4(↓ 19.3)	23.7(↓ 35.7)	6.5(↓ 21.2)

Table 7: Performance of LLMs on PersonaPath (Basic Education) in the presence of contextual noise.

Contextual noise degrades performance broadly, most notably in Validity and Efficiency. Qwen3-30B-A3B (CoT) loses 54.9 percentage points in Validity and 35.7 points in Efficiency. Since learners are profiled for the target domain, selecting out-of-domain textbooks can introduce prerequisite violations and divert steps from the target. The planner must select the relevant part of the knowledge structure for the learner's goal.

5.3 Effects of Dynamic Interaction

We contrast the step-by-step paradigm with one-shot generation, where the model plans the entire curriculum in a single turn. Table 8 presents results for Basic Education; Higher Education results are in Appendix B.

Model	Validity Pass Rate	Adaptivity Pass Rate	Efficiency Pass Rate	Final Pass Rate
Qwen3-4B-2507$_{CoT}$	86.6(↑ 30.8)	20.6(↓ 23.3)	38.1(↓ 22.1)	5.7(↓ 15.1)
Pangu-7B$_{CoT}$	48.0(↑ 18.1)	24.0(↓ 4.5)	16.0(↓ 30.4)	5.4(↓ 9.6)
DeepSeek-R1-Distill-14B$_{CoT}$	77.9(↑ 28.0)	19.9(↓ 23.5)	23.3(↓ 33.8)	4.2(↓ 16.2)
Qwen3-30B-A3B$_{CoT}$	84.4(↑ 13.6)	15.9(↓ 28.8)	30.2(↓ 29.2)	3.6(↓ 24.1)

Table 8: Performance of LLMs on PersonaPath (Basic Education) with one-shot path generation.

Table 8 shows a divergence between structural validity and adaptation. Static generation increases Validity by 30.8 percentage points for Qwen3-4B-2507 (CoT), while Adaptivity decreases by 28.8 points for Qwen3-30B-A3B (CoT) and Efficiency by 33.8 points for DeepSeek-R1-Distill-14B. In the interactive setting, each decision can use the mastery changes produced by previous concepts. The one-shot planner fixes its sequence before receiving these updates, reducing its ability to adjust content selection as the learner progresses.

6 Conclusion

We study Knowledge-Centric personalized learning path planning, where models use explicit learner goals, mastery states, and curriculum prerequisites to select and order knowledge. To evaluate this setting, we introduce **PersonaPath**, a benchmark that pairs 2,000 learner personas with a hierarchical knowledge graph built from 347 textbooks and organized into units and concepts. Expert ratings support the pedagogical plausibility and internal coherence of the sampled personas. Experiments on representative LLMs show that current models still struggle with KC planning, especially in adapting paths to learner states. The ablations identify mastery information and step-by-step feedback as important inputs for personalized planning.

Limitations

PersonaPath uses synthetic learner states and mastery updates. Expert evaluation covers the pedagogical plausibility and internal coherence of 100 personas, not the relationship between benchmark scores and real learner outcomes. Longitudinal traces and classroom studies could measure observed progress and teacher interventions, while broader expert evaluation could assess generated paths.

The benchmark is grounded in Chinese textbooks and curricula. Cross-language and cross-curriculum transfer requires adapting the source materials and validating prerequisite structures and simulation settings. Controlled studies should distinguish language and curriculum effects from planning performance; Appendix C describes the adaptation procedure.

Ethics Statement

The learner personas used in PersonaPath are synthetic and generated from statistical archetypes (Low-performing, Average-performing, and High-performing) to simulate diverse cognitive states. No real-world student data or Personally Identifiable Information (PII) were collected, stored, or processed in the construction of this benchmark. This design reduces privacy risks associated with real learner data.

The knowledge substrate of our benchmark is derived from 347 textbooks sourced from authoritative platforms, including People's Education Press and the Smart Education of China platform. These materials were used strictly for the purpose of constructing an academic benchmark, extracting knowledge graphs and prerequisite relations. We do not distribute the full raw text of copyrighted books; instead, we release the structured knowledge graphs and derived metadata necessary for reproducing the experiments.

The observed hallucinations and prerequisite violations motivate teacher review of generated learning paths before classroom use. Educational applications should retain teacher oversight of content selection and learning goals.

Acknowledgments

Thanks for the insightful comments and feedback from the reviewers. This work was supported by the National Natural Science Foundation of China (No. 62406015).

References

Albert Bandura. 1977. Self-efficacy: toward a unifying theory of behavioral change. *Psychological review*, 84(2):191.

Cun-Ling Bian, De-Liang Wang, Shi-Yu Liu, Wei-Gang Lu, and Jun-Yu Dong. 2019. Adaptive learning path recommendation based on graph theory and an improved immune algorithm. *KSII Transactions on Internet & Information Systems*, 13(5).

Haw-Shiuan Chang, Hwai-Jung Hsu, Kuan-Ta Chen, and 1 others. 2015. Modeling exercise relationships in e-learning: A unified approach. In *EDM*, pages 532–535.

Hanting Chen, Yasheng Wang, Kai Han, Dong Li, Lin Li, Zhenni Bi, Jinpeng Li, Haoyu Wang, Fei Mi, Mingjian Zhu, Bin Wang, Kaikai Song, Yifei Fu, Xu He, Yu Luo, Chong Zhu, Quan He, Xueyu Wu, Wei He, and 5 others. 2025a. Pangu embedded: An efficient dual-system llm reasoner with metacognition. *Preprint*, arXiv:2505.22375.

Ruirui Chen, Weifeng Jiang, Chengwei Qin, Bo Xiong, Fiona Liausvia, Dongkyu Choi, and Boon Kiat Quek. 2025b. Are large language models effective knowledge graph constructors? *Preprint*, arXiv:2510.11297.

Song Cheng, Qi Liu, Enhong Chen, Zai Huang, Zhenya Huang, Yiying Chen, Haiping Ma, and Guoping Hu. 2019. DIRT: Deep learning enhanced item response theory for cognitive diagnosis. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management (CIKM)*, pages 2397–2400.

Zihao Cheng, Hongru Wang, Zeming Liu, Yuhang Guo, Yuanfang Guo, Yunhong Wang, and Haifeng Wang. 2025. ToolSpectrum: Towards personalized tool utilization for large language models. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 20679–20699.

Jacob Cohen. 1960. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20(1):37–46.

Albert T. Corbett and John R. Anderson. 1995. Knowledge tracing: Modeling the acquisition of procedural knowledge. *User Modeling and User-Adapted Interaction*, 4(4):253–278.

Bin Deng, Yizhe Feng, Zeming Liu, Qing Wei, Xiangrong Zhu, Shuai Chen, Yuanfang Guo, and Yunhong Wang. 2025. RETAIL: Towards real-world travel planning for large language models. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 14870–14902.

Jean-Paul Doignon and Jean-Claude Falmagne. 2012. *Knowledge spaces*. Springer Science & Business Media.

Lumbardh Elshani and Krenare Pireva Nuçi. 2021. Constructing a personalized learning path using genetic algorithms approach. *arXiv preprint arXiv:2104.11276*.

Mingyu Feng, Neil Heffernan, and Kenneth Koedinger. 2009. Addressing the assessment challenge with an online system that tutors as it assesses. *User modeling and user-adapted interaction*, 19(3):243–266.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, and 1 others. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, and 1 others. 2025. Deepseek-r1 incentivizes reasoning in llms through reinforcement learning. *Nature*, 645:633–638.

Tebourbi Hedi, Sana Nouzri, Yazan Mualla, and Abdeljalil Abbas-Turki. 2025. Artificial intelligence agents for personalized adaptive learning. *Procedia Computer Science*, 265:252–259.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. 2023. Survey of hallucination in natural language generation. *ACM Computing Surveys*, 55(12):1–38.

Yuxin Jiang, Yufei Wang, Xingshan Zeng, Wanjun Zhong, Liangyou Li, Fei Mi, Lifeng Shang, Xin Jiang, Qun Liu, and Wei Wang. 2024. Follow-Bench: A multi-level fine-grained constraints following benchmark for large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL)*.

J Richard Landis and Gary G Koch. 1977. The measurement of observer agreement for categorical data. *Biometrics*, 33(1):159–174.

Hang Li, Tianlong Xu, Chaoli Zhang, Eason Chen, Jing Liang, Xing Fan, Haoyang Li, Jiliang Tang, and Qingsong Wen. 2024. Bringing generative ai to adaptive learning in education. *Preprint*, arXiv:2402.14601.

Qingyao Li, Wei Xia, Li'ang Yin, Jian Shen, Renting Rui, Weinan Zhang, Xianyu Chen, Ruiming Tang, and Yong Yu. 2023. Graph enhanced hierarchical reinforcement learning for goal-oriented learning path recommendation. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, pages 1318–1327.

Qi Liu, Shiwei Tong, Chuanren Liu, Hongke Zhao, Enhong Chen, Haiping Ma, and Shijin Wang. 2019. Exploiting cognitive structure for adaptive learning. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 627–635.

Zeming Liu, Ding Zhou, Hao Liu, Haifeng Wang, Zheng-Yu Niu, Hua Wu, Wanxiang Che, Ting Liu, and Hui Xiong. 2023. Graph-grounded goal planning for conversational recommendation. *IEEE Transactions on Knowledge and Data Engineering*, 35(5):4923–4939.

Frederic M Lord. 2012. *Applications of item response theory to practical testing problems*. Routledge.

Weiming Lu, Yangfan Zhou, Jiale Yu, and Chenhao Jia. 2019. Concept extraction and prerequisite relation learning from educational data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 9678–9685.

Ian R McKenzie, Alexander Lyzhov, Michael Pieler, Alicia Parrish, Aaron Mueller, Ameya Prabhu, Euan McLean, Aaron Kirtland, Alexis Ross, Alisa Liu, and 1 others. 2023. Inverse scaling: When bigger isn't better. *arXiv preprint arXiv:2306.09479*.

Mehdi Niknam and Parimala Thulasiraman. 2020. Lpr: A bio-inspired intelligent learning path recommendation system based on meaningful learning theory. *Education and Information Technologies*, 25(5):3797–3819.

LF Obukhova and IA Korepanova. 2009. The zone of proximal development: A spatiotemporal model. *Journal of Russian & East European Psychology*, 47(6):25–47.

Shalini Pandey and George Karypis. 2019. A self-attentive model for knowledge tracing. In *Proceedings of the 12th International Conference on Educational Data Mining (EDM)*, pages 384–389.

Chris Piech, Jonathan Bassen, Jonathan Huang, Surya Ganguli, Mehran Sahami, Leonidas J Guibas, and Jascha Sohl-Dickstein. 2015. Deep knowledge tracing. *Advances in neural information processing systems*, 28.

Siyu Song, Wentao Liu, Ye Lu, Ruohua Zhang, Tao Liu, Jinze Lv, Xinyun Wang, Aimin Zhou, Fei Tan, Bo Jiang, and Hao Hao. 2025. Cultivating helpful, personalized, and creative ai tutors: A framework for pedagogical alignment using reinforcement learning. *Preprint*, arXiv:2507.20335.

Yaxin Tu, Jili Chen, and Changqin Huang. 2025. Empowering personalized learning with generative artificial intelligence: Mechanisms, challenges and pathways. *Frontiers of Digital Education*, 2(2):19.

Carnegie Mellon University. 2011. Oli engineering statics - fall 2011. DataShop @CMU. Accessed: 2026-01-01.

Karthik Valmeekam, Matthew Marquez, Alberto Olmo, Sarath Sreedharan, and Subbarao Kambhampati. 2023. PlanBench: An extensible benchmark for evaluating large language models on planning and reasoning about change. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 36.

Lev S Vygotsky. 1978. *Mind in Society: The Development of Higher Psychological Processes*. Harvard University Press.

Hongru Wang, Rui Wang, Boyang Xue, Heming Xia, Jingtao Cao, Zeming Liu, Jeff Z. Pan, and Kam-Fai Wong. 2024. AppBench: Planning of multiple APIs from various APPs for complex user instruction. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 15322–15336.

Yutao Wang, Neil T Heffernan, and Cristina Heffernan. 2015. Towards better affect detectors: effect of missing skills, class features and common wrong answers. In *Proceedings of the fifth international conference on learning analytics and knowledge*, pages 31–35.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, and 1 others. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

Derong Xu, Wei Chen, Wenjun Peng, Chao Zhang, Tong Xu, Xiangyu Zhao, Xian Wu, Yefeng Zheng, Yang Wang, and Enhong Chen. 2024. Large language models for generative information extraction: A survey. *arXiv preprint arXiv:2312.17617*.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, and 1 others. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Runlong Yu, Yunzhou Zhang, Yuyang Ye, Le Wu, Chao Wang, Qi Liu, and Enhong Chen. 2018. Multiple pairwise ranking with implicit feedback. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, pages 1727–1730.

Haotian Zhang, Shuanghong Shen, Bihan Xu, Zhenya Huang, Jinze Wu, Jing Sha, and Shijin Wang. 2024. Item-difficulty-aware learning path recommendation: From a real walking perspective. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 4167–4178.

Jiani Zhang, Xingjian Shi, Irwin King, and Dit-Yan Yeung. 2017. Dynamic key-value memory networks for knowledge tracing. In *Proceedings of the 26th International Conference on World Wide Web (WWW)*, pages 765–774.

Qian Zhang, Jie Lu, and Guangquan Zhang. 2021. Recommender systems in e-learning. *Journal of Smart Environments and Green Computing*, 1(2):76–89.

Kaikai Zhao, Zhaoxiang Liu, Xuejiao Lei, Jiaojiao Zhao, Zhenhong Long, Zipeng Wang, Ning Wang, Meijuan An, Qingliang Meng, Peijun Yang, Minjie Hua, Chaoyang Ma, Wen Liu, Kai Wang, and Shiguo Lian. 2025. Quantifying the capability boundary of deepseek models: an application-driven performance analysis. *arXiv preprint arXiv:2502.11164*.

Yuwen Zhou, Changqin Huang, Qintai Hu, Jia Zhu, and Yong Tang. 2018. Personalized learning full-path recommendation model based on lstm neural networks. *Information sciences*, 444:135–152.

A Benchmark Details

A.1 Cognitive State Transition Dynamics

This section provides the formal definition of the environment transition function $\text{Sim}(\mathbf{M}_t[u], D(a_t))$, which simulates the dynamic update of a learner's proficiency based on their interaction with an atomic concept a_t. Let $u \in \mathcal{U}$ be the parent unit of concept a_t. The transition consists of two stochastic phases: *Performance Simulation* and *Mastery Gain Computation*.

A.1.1 Performance Simulation (IRT Model)

We model the probability of a learner successfully mastering concept a_t using the logistic IRT model (Lord, 2012). Given the learner's current unit mastery $\mathbf{M}_t[u]$ and the concept's inherent difficulty $D(a_t)$, the probability of success $P(Y_t = 1)$ is defined as:

$$P(Y_t = 1|\mathbf{M}_t, a_t) = \frac{1}{1 + \exp\left(-k \cdot \Delta_{\text{gap}}\right)} \quad (2)$$

where:

- $\Delta_{\text{gap}} = \mathbf{M}_t[u] - D(a_t) + \delta_{\mathbf{P}}$ denotes the effective ability gap adjusted by the learner's persona.
- $Y_t \in \{0, 1\}$ is the binary outcome of the interaction (1 for success, 0 for failure).
- k is the discrimination parameter, controlling the sensitivity of the outcome to the ability-difficulty gap.
- $\delta_{\mathbf{P}}$ is a bias term derived from the learner's static persona $\mathbf{P}$.

A.1.2 Difficulty-Aware Gain Function

The core of the transition dynamics is the *Difficulty-Aware Gain Function*. Unlike constant-gain models, we posit that the magnitude of proficiency improvement Δm_t depends on both the complexity of the task and the learner's relative performance.

Case 1: Successful Interaction ($Y_t = 1$). A successful learning event yields a gain proportional

to the concept's difficulty. We also incorporate a **Challenge Bonus** based on the Zone of Proximal Development (ZPD) theory (Obukhova and Korepanova, 2009): if a learner successfully masters a concept harder than their current proficiency ($D(a_t) > \mathbf{M}_t[u]$), the gain is amplified.

$$\Delta m_t^+ = \underbrace{(\lambda_1 D(a_t) + \lambda_0)}_{\text{Info. Utility}} \cdot \underbrace{\left(\frac{\eta_{\mathbf{P}}}{\eta_{\text{ref}}}\right)}_{\text{Persona Scaling}} \cdot \underbrace{(1 + \gamma \cdot \mathbb{I}(D(a_t) > \mathbf{M}_t[u]))}_{\text{Breakthrough Bonus}} \quad (3)$$

where λ_1 and λ_0 are difficulty utility coefficients, $\eta_{\mathbf{P}}$ is the individual learning rate from persona $\mathbf{P}$, and γ is the bonus coefficient for cognitive breakthroughs.

Case 2: Failed Interaction ($Y_t = 0$). Failure results in a marginal gain, simulating the "familiarity effect" derived from trial and error:

$$\Delta m_t^- = \eta_{\mathbf{P}} \cdot \epsilon \quad (4)$$

A.1.3 State Update Rule

Finally, the unit-level mastery vector is updated for the next time step $t + 1$:

$$\mathbf{M}_{t+1}[u] = \min\left(\mathbf{M}_t[u] + \Delta m_t, 1.0\right) \quad (5)$$

where Δm_t is determined by the sampled outcome Y_t.

Symbol	Value	Description
k	4.0	IRT discrimination factor
λ_1, λ_0	0.25, 0.1	Coefficients for difficulty-based reward
η_{ref}	0.15	Reference learning rate for normalization
γ	0.1	Challenge bonus factor (10%)
ϵ	0.05	Failure compensation factor

Table A.1: Hyperparameters used in the simulation function.

A.1.4 Sensitivity Analysis of Simulation Hyperparameters

The cognitive state transition function (Equations 2–5) relies on a set of hyperparameters (Table A.1) whose values were selected empirically. To evaluate the sensitivity of our conclusions to these values, we varied each parameter individually while holding the others at their default values and measured the effect on learning trajectories across all three proficiency archetypes (Low-performing, Average-performing, and High-performing). We performed 1,000 Monte Carlo runs per configuration.

Parameter Selection Rationale. The hyperparameters in our simulation were selected based on established principles from Item Response Theory and educational modeling:

- **Discrimination parameter (k):** Following IRT conventions (Lord, 2012), k is set within $[3.0, 5.0]$. A lower k yields poor discrimination because success probabilities flatten near 50% regardless of ability, while an excessively high k creates an unrealistic step function, causing abrupt pass/fail outcomes based on marginal ability-difficulty differences.

- **Difficulty utility coefficients (λ_1, λ_0):** These govern the base mastery gain per successful interaction. Their values are calibrated so that an average learner requires approximately 3–5 interactions to master a unit, reflecting realistic learning trajectories observed in educational practice.

- **Challenge bonus (γ):** We set γ within $[0.1, 0.2]$ to regulate the extra cognitive gain from succeeding at tasks above the learner's current proficiency. This range accelerates learning for high-ability students while preventing extreme, unstable mastery spikes caused by lucky guesses.

- **Failure compensation (ϵ):** This small constant simulates the "familiarity effect" from trial and error. Its value ensures that failed interactions still contribute marginal progress without undermining the distinction between success and failure.

Impact on Learning Speed. Figure A.1 presents the mean steps to mastery ($\tau = 0.8$) as each parameter is varied across a wide range. Two key observations emerge:

(1) Parameters primarily affect learning speed, not relative ordering. Across all 25 parameter configurations tested, the archetype ordering (Low-performing > Average-performing > High-performing in required steps) is preserved with 100% consistency, as shown in Table A.3. This indicates that parameter variations uniformly scale the difficulty of the environment without distorting the distinction between learner archetypes.

(2) Parameter sensitivity varies substantially. We quantify each parameter's influence using the Coefficient of Variation (CV) of mean steps across

	k	λ_1	λ_0	γ	ϵ
CV	0.015	0.234	0.177	0.081	0.036

Table A.2: Coefficient of Variation (CV) of mean steps to mastery across parameter ranges. Higher CV indicates greater sensitivity.

Param	Value	Low-perf.	Avg-perf.	High-perf.	Order
k	2.0	6.5 ± 2.0	3.2 ± 1.4	2.3 ± 0.8	✓
	3.0	6.5 ± 2.1	3.3 ± 1.5	2.2 ± 0.7	✓
	4.0*	6.5 ± 2.1	3.2 ± 1.4	2.3 ± 0.8	✓
	5.0	6.6 ± 2.2	3.2 ± 1.4	2.2 ± 0.7	✓
	6.0	6.9 ± 2.6	3.2 ± 1.5	2.3 ± 0.7	✓
λ_1	0.10	9.4 ± 2.4	5.5 ± 1.3	2.8 ± 1.1	✓
	0.15	8.1 ± 2.4	4.5 ± 1.5	2.7 ± 1.0	✓
	0.25*	6.7 ± 2.2	3.2 ± 1.5	2.2 ± 0.7	✓
	0.35	5.6 ± 1.7	3.1 ± 1.4	1.7 ± 1.1	✓
	0.45	5.1 ± 2.1	3.0 ± 1.3	1.7 ± 1.0	✓
γ	0.00	6.6 ± 2.2	3.2 ± 1.4	2.8 ± 1.1	✓
	0.05	6.7 ± 2.3	3.2 ± 1.4	2.7 ± 0.9	✓
	0.10*	6.5 ± 2.1	3.3 ± 1.5	2.2 ± 0.6	✓
	0.20	6.4 ± 2.0	3.2 ± 1.4	1.7 ± 1.1	✓
	0.30	6.3 ± 2.0	3.2 ± 1.5	1.7 ± 1.0	✓

Table A.3: Mean steps to mastery under varied parameter settings. "$*$" marks the default configuration. The archetype ordering is preserved across all 25 tested configurations (✓).

its tested range. As shown in Table A.2, the gain coefficients λ_1 (CV $= 0.234$) and λ_0 (CV $= 0.177$) exhibit the highest sensitivity, as they directly control the magnitude of mastery increments per interaction. In contrast, the discrimination parameter k (CV $= 0.015$) and the failure compensation ϵ (CV $= 0.036$) have negligible impact on overall learning speed, since they modulate probabilities and marginal gains rather than the primary mastery update pathway.

Preservation of Archetype Ordering. Table A.3 provides detailed results for the three most representative parameters (k, λ_1, and γ). Across all 15 configurations, the step ratio between adjacent archetypes (Low/Average and Average/High) remains strictly above 1.0, confirming that no parameter setting causes the archetype ordering to invert. Even for the most sensitive parameter, λ_1, where the mean steps range from 5.1 to 9.4 for Low-performing learners, the ordering Low-performing $>$ Average-performing $>$ High-performing is preserved.

A.1.5 Implications for Evaluation Validity

The sensitivity analysis measures simulated learning speed and the ordering of proficiency archetypes. Across the tested settings, the archetype ordering remains stable while the number of steps to mastery varies. These results characterize the behavior of the learner simulator under parameter changes.

In closed-loop planning, simulation parameters affect the mastery states supplied to subsequent decisions. Although the constraint definitions remain fixed, changes in the state trajectory can alter concept selection, Cog-Gap, and goal completion. The main model comparisons therefore use a common simulator configuration; evaluating model rankings under alternative configurations would require rerunning the planners with those dynamics.

A.2 Dataset Comparison

Existing benchmarks for evaluating educational planning, including Junyi Academy (Chang et al., 2015), ASSISTments 2009/2012 (Feng et al., 2009; Wang et al., 2015), OLI Engineering Statics (University, 2011), and synthetic data from DKT (Piech et al., 2015), primarily support **Exercise-Centric (EC)** evaluation. These benchmarks are valuable for localized item recommendation and student performance prediction, but they provide limited support for evaluating whether a model can construct a goal-oriented curriculum path from explicit learner states and prerequisite knowledge structures. **Existing Benchmarks in Detail.** The **ASSISTments** family (Feng et al., 2009; Wang et al., 2015) is widely used for student performance prediction. The 2009 version focuses on correctness logs, while the 2012 version additionally incorporates a predicted affective state of the student. The **Junyi Academy** dataset (Chang et al., 2015) provides extensive records of student attempts, hints, and time spent across topics ranging from arithmetic to geometry. The **OLI Engineering Statics** dataset (University, 2011) contains 189,297 trials from a college-level engineering statics course. Synthetic datasets, such as those generated for Deep Knowledge Tracing (DKT) (Piech et al., 2015), simulate virtual student responses using Item Response Theory to support model testing. Despite their differences in scale and domain, all of these resources record item-level interactions rather than the explicit knowledge structures and learner goals required for KC path planning.

PersonaPath instantiates a **Knowledge-Centric (KC)** planning benchmark, described by the following five dimensions.

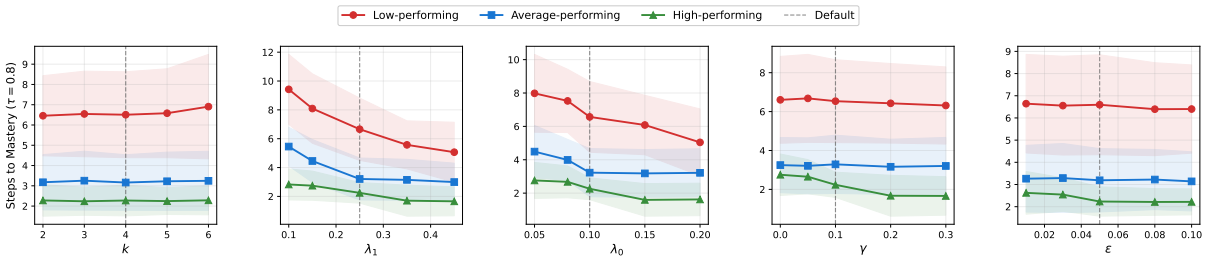


Figure A.1: Mean steps to mastery ($\tau = 0.8$) as each hyperparameter is varied individually. Shaded regions denote ± 1 standard deviation across 1,000 runs. Dashed vertical lines indicate default values. The relative ordering among proficiency archetypes is preserved across all configurations.

Dataset Name	Disciplines Covered
Junyi Academy (Chang et al., 2015)	Mathematics, Biology, Computer Science (basic education).
ASSISTments2009 (Feng et al., 2009)	Mathematics (basic education).
ASSISTments2012 (Wang et al., 2015)	Mathematics (basic education).
OLI Engineering Statics (University, 2011)	Engineering Statics (higher education).
PersonaPath (Ours)	Chinese, Mathematics, English, Physics, Chemistry, Biology, History, Geography, Morality and the Rule of Law, Information Technology, Music, Art, P.E. (basic education). Law, Engineering, Management, Education, Economics, Science, History, Agronomy, Literature, Medicine, Arts, Philosophy (higher education).

Table A.4: Comparison of discipline coverage across datasets.

A.2.1 Textbook-level Prerequisite Graph

Textbook-level Prerequisite Graph (TPG) describes dependencies among textbooks and supports the evaluation of prerequisite-aware planning across a curriculum. Item-level skill annotations and exercise relations in existing datasets operate at different granularities of learner modeling. PersonaPath provides textbook-level dependencies together with the nested units and concepts needed to evaluate paths across these resources.

We provide visualizations of these TPG structures across various disciplines in PersonaPath: Figures F.7 through F.18 illustrate the prerequisite graphs for 13 subjects in Basic Education, while Figures F.19 and F.20 depict professional learning paths for Higher Education. Taking the "Mechanics" major within the Engineering category as an example (the teal path in Fig. F.20), the TPG serves as a benchmark constraint: the LLM agent is expected to identify and follow a learning sequence that covers fundamental Mathematics and Physics before Theoretical Mechanics and Material Mechanics. By introducing TPG as a benchmark requirement, we provide a structured way to evaluate whether LLMs can follow curriculum-level planning constraints, a capability that is difficult to assess using prior, less-structured datasets.

A.2.2 Persona

Persona (PSN) describes the learner information available to a planner. PersonaPath constructs explicit profiles with **Proficiency Archetypes** (Low-performing, Average-performing, and High-performing) in an approximate 1:3:1 ratio and **Academic Ambitions** represented by target majors that define long-term goals. **Temporal Context** establishes a current grade level and partitions the knowledge base into already learned, currently learning, and yet-to-be-learned content. These attributes accompany the mastery state, giving the planner both a starting point and a target. Interaction datasets such as ASSISTments and Junyi instead support estimating learner states from recorded behavior.

A.2.3 Breadth of Educational Disciplines

Breadth of Educational Disciplines (BED) defines the evaluation boundaries of the benchmark, covering both general knowledge and higher-level professional expertise. As detailed in Table A.4, PersonaPath organizes 77 subjects. In contrast, existing benchmarks are more limited in scope;

for example, Junyi Academy and ASSISTments focus mainly on K-12 Mathematics, while OLI Engineering Statics is confined to a single specialized course.

A.2.4 Hierarchical Knowledge Structure

Hierarchical Knowledge Structure (HKS) establishes a structured framework beyond the flat skill labeling found in traditional datasets. While previous benchmarks often represent knowledge as an unstructured collection of independent skills, PersonaPath formalizes the knowledge space into a three-tiered nested architecture: **Textbooks**, **Units**, and **Atomic Concepts**. This hierarchy, encompassing 347 textbooks, 1,751 units, and 4,092 concepts, approximates the structural organization of institutional education.

A.2.5 Rich Semantics

Rich Semantics (RS) measures whether a benchmark provides informative semantic details rather than only textual labels. In PersonaPath, every textbook, unit, and concept is enriched with semantic content, including definitions, learning objectives, and pedagogical descriptions. Most existing datasets are collections of exercise records or problem sets in which knowledge nodes are often reduced to opaque IDs or brief strings. This lack of semantic depth makes it difficult to evaluate whether models can reason over educational content when planning a path.

A.3 Data Collection

A.3.1 Rationality of Data Source Selection

To support the pedagogical validity and broad applicability of PersonaPath, our data collection strategy followed these principles:

Authoritative Provenance. We prioritized reliability in educational content. For Basic Education, materials were sourced from the People's Education Press, a widely used publisher for Chinese K-12 curriculum. For Higher Education, resources were curated from the Smart Education of China platform to align with national academic standards. This selection strategy reduces the noise often found in crowdsourced open-web educational data.

Holistic Curriculum Continuity. A key rationale for our dataset construction was to bridge the gap between disjointed educational stages. Unlike existing datasets that focus on isolated exercises, we constructed a dependency graph that explicitly links Basic Education subjects (e.g., Biology) as prerequisites for Higher Education majors (e.g., Medicine). This structural design allows for the evaluation of long-horizon, cross-stage curriculum planning capabilities.

Representative Learner Modeling. We use synthetic personas derived from statistical archetypes (Low-performing, Average-performing, and High-performing) to simulate a controlled yet diverse range of cognitive states. This approach provides a balanced distribution for difficulty-adaptability tests (an approximate 1:3:1 ratio), which is often difficult to obtain from skewed real-world interaction logs.

A.3.2 Compliance and Ethical Standards

Our data construction and release process follows legal and ethical guidelines regarding intellectual property, privacy, and labor rights:

Intellectual Property and Copyright Compliance. We strictly distinguish between the raw content of textbooks and the derived knowledge structures. *Usage:* The textbooks served solely as the source for extracting knowledge graphs, prerequisite relations, and concept hierarchies. *Distribution:* To comply with copyright laws, we do not distribute, host, or reproduce the full raw text of the copyrighted books. The public release of PersonaPath is limited to the structured knowledge graphs (metadata), relationship triplets, and the synthetic persona profiles. This constitutes a transformative use of the data for academic research purposes.

Privacy and Human Subjects Exemption. The learner profiles used in the benchmark are synthetic. No real-world student data or Personally Identifiable Information (PII) were collected, stored, or processed. Since no real learner data are used, the benchmark is designed to avoid privacy risks associated with real student records. For the human-in-the-loop verification process involving domain experts, we followed ethical labor standards. All annotators were informed of the purpose of the data beforehand and were compensated at an hourly rate above the local minimum wage.

A.4 Human Verification and Compensation

As detailed in Section 3.3, we recruited domain experts to verify prerequisite relations and content correctness in the knowledge space. These experts were selected based on their academic background in the respective disciplines. All annotators were informed of the purpose of the data before the task

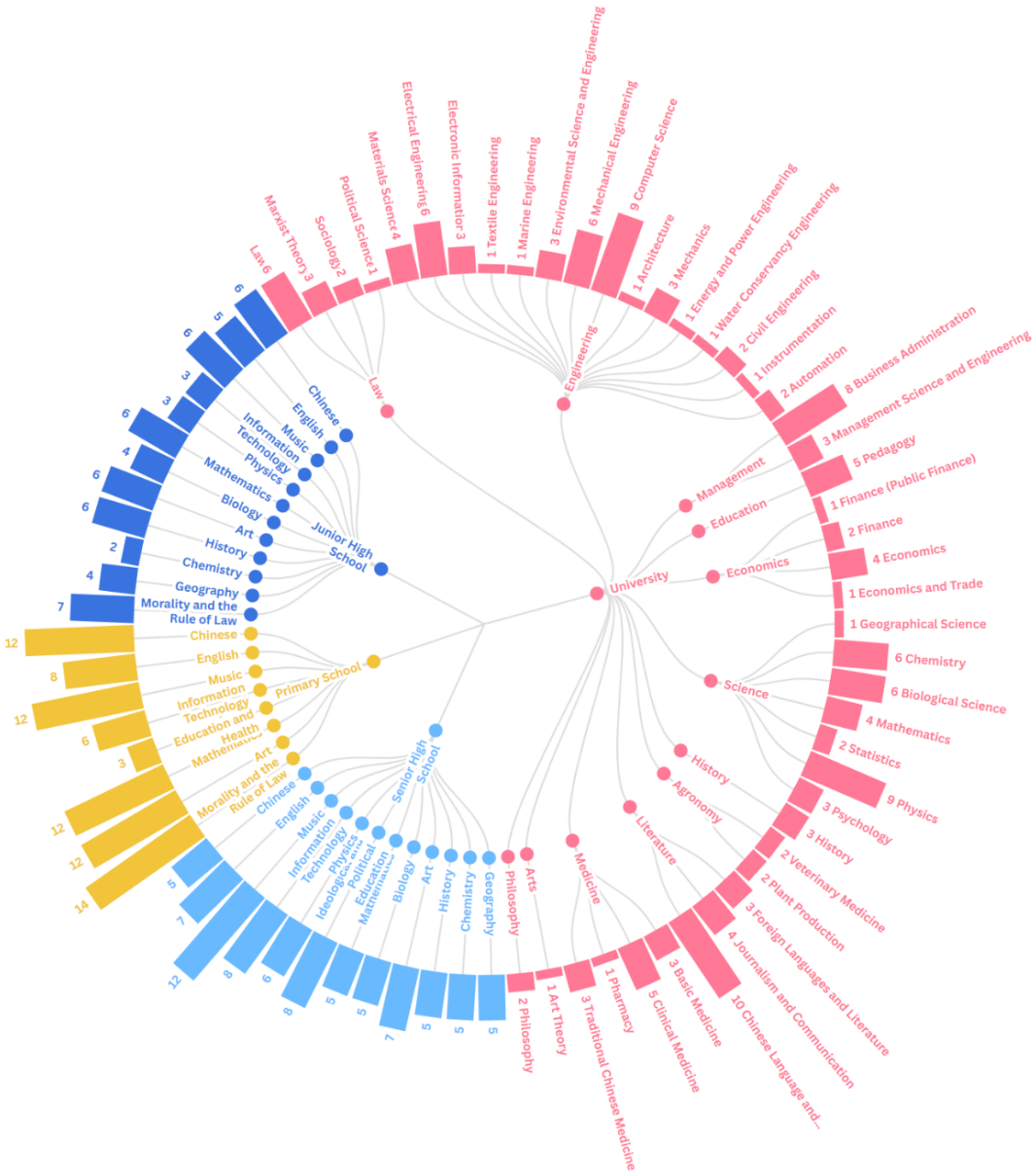


Figure A.2: Overview of the disciplines in PersonaPath.

and were compensated at an hourly rate above the local minimum wage.

A.4.1 Instructions for Prerequisite Relation Validation

Experts were presented with candidate pairs of concepts (C_i, C_j) and were instructed to validate the directed link $C_i \rightarrow C_j$ based on the following criteria:

- **Logical Precedence:** "Does concept C_i provide the foundational knowledge required to understand C_j?" (Binary: Yes/No)
- **Pedagogical Alignment:** "Is the proposed sequence consistent with the standard teaching order found in the source textbooks?" (Binary: Yes/No)
- **Action:** Mark the link as *Invalid* if it creates a logical cycle or violates pedagogical norms.

A.4.2 Instructions for Content Correctness Audit

For the entities and semantic descriptions generated by the LLM, experts were provided with the source textbooks as ground truth. The specific instructions were:

1. **Fact Verification:** "Verify that the definition and properties of the generated entity align

strictly with the textbook content. Mark any factual deviations."

2. **Hallucination Check:** "Flag any terms or relations that do not exist in the domain context or appear to be fabricated by the model."

3. **Terminology Standard:** "Ensure that the technical terminology used is appropriate for the target educational stage."

A.5 Hierarchical Transition Mechanism

The agent operates through a three-level hierarchical transition mechanism:

Textbook Transition. Once all units within the current textbook achieve the mastery threshold τ, the agent consults the prerequisite graph to determine the next textbook. This is a graph-level decision that considers the learner's global mastery state rather than a fixed linear sequence.

Unit Transition. Within a given textbook, the agent traverses units sequentially. It advances to the next unit only when the current unit's aggregated mastery reaches the target threshold τ.

Concept Transition. Within an unmastered unit, the agent selects the next concept by matching the unit's current mastery level against each candidate concept's difficulty parameter. This selection mechanism is designed to target the learner's Zone of Proximal Development (ZPD) (Obukhova and Korepanova, 2009), ensuring that the chosen concept is neither trivially easy nor prohibitively difficult.

B Experiment Details

B.1 Constraints

Table B.5 provides detailed definitions for the three categories of constraints (Validity, Adaptivity, and Efficiency) introduced in Section 4.2. Together, these constraints assess whether an agent can produce a pedagogically appropriate learning sequence under the benchmark setting.

Validity Constraints. Validity evaluates whether a generated path is structurally and factually sound within the provided curriculum. The *Prerequisite Violation* constraint enforces the cumulative nature of learning. Following Knowledge Space Theory (Doignon and Falmagne, 2012), meaningful learning depends on a structured surmise relation, where advanced concepts should not be introduced before their antecedent concepts have been sufficiently mastered. The *Hallucination* constraint safeguards factual accuracy by prohibiting fictitious educational entities, such as nonexistent textbooks, units, or concepts outside the provided knowledge space. Such entities are a documented failure mode of LLMs (Ji et al., 2023).

Adaptivity Constraints. Adaptivity evaluates whether the path fits the specific learner rather than a generic curriculum order. The *Persona Alignment* constraint is motivated by the Zone of Proximal Development (ZPD) (Vygotsky, 1978; Obukhova and Korepanova, 2009): a pedagogically appropriate path should remain consistent with the learner's cognitive archetype, prior learning experience, and behavioral profile. The *Difficulty Adaptability* constraint uses Cog-Gap (Eq. 1) to compare current unit mastery with the difficulty of the selected concept. Concept difficulty follows the Item Response Theory parametrization (Cheng et al., 2019).

Efficiency Constraints. Efficiency evaluates whether the path reaches the learning goal within practical bounds. The *Goal Completion* constraint requires the agent to achieve the target mastery level within a maximum allowable number of steps, in line with step-bounded goal-achievement evaluation for LLM planning (Valmeekam et al., 2023). The maximum step limit $T_{\max}$ is set in proportion to the number of concepts in the target stage, so longer target stages receive a larger but still bounded planning horizon. The *Progress Continuity* constraint prevents learning stagnation by requiring the average proficiency increment over a sliding window ($n = 5$) to exceed a predefined threshold. This criterion is also pedagogically motivated: sustained mastery experiences help protect learner *self-efficacy* (Bandura, 1977), whereas long stretches of negligible progress indicate inefficient or poorly sequenced instruction.

Final Pass Rate. Following the multi-constraint evaluation protocol (Jiang et al., 2024), a learning path passes the evaluation only if it satisfies all three dimensions simultaneously. We define the final pass indicator as $Pass_{final} = \mathbb{I}(Validity) \cdot \mathbb{I}(Adaptivity) \cdot \mathbb{I}(Efficiency)$, where $\mathbb{I}(\cdot)$ outputs 1 when the corresponding dimension is fully satisfied and 0 otherwise. The Final Pass Rate is the proportion of generated paths whose $Pass_{final}$ equals 1.

B.2 Overall Performance

DeepSeek-V3.1 achieves the highest final pass rates in our experiments, yet reaches only 29.5% and 14.6% in Basic and Higher Education, respec-

Constraint	Description
	Validity Constraints
Prerequisite Violation	Occurs when a learning item is recommended before its necessary foundational prerequisites have been mastered by the learner.
Hallucination	The generation of spurious or non-existent educational entities, such as fictitious textbooks or concepts.
	Adaptivity Constraints
Difficulty Adaptability	Quantifies the discrepancy between a learner's current mastery level and the intrinsic difficulty of the concept to learn.
Persona Alignment	The degree of consistency between the planning strategy and the learner's explicit cognitive profile.
	Efficiency Constraints
Goal Completion	Indicates whether the learning task is completed within the prescribed step limit.
Progress Continuity	Mandates that the average proficiency increment calculated over a sliding window ($n = 5$) must exceed a predefined threshold to ensure sustained learning momentum.

Table B.5: Description of constraints used in the evaluation.

tively. These low absolute scores suggest that PersonaPath is challenging for current LLMs in KC learning path planning. We observe that model performance generally improves with larger model sizes. For instance, Pangu-7B (CoT) achieves a final pass rate of 15.0% in Basic Education, whereas the corresponding Pangu-1B (CoT) reaches only 1.1%. However, increasing model size does not guarantee improvements across all constraints, echoing inverse-scaling observations in which larger models can underperform on certain tasks (McKenzie et al., 2023). Qwen3-4B-2507 (CoT) (60.2%) slightly surpasses Qwen3-30B-A3B (CoT) (59.4%) on the *Efficiency* constraint, showing that larger scale alone does not guarantee higher Efficiency.

B.3 Additional Experimental Results

This section provides additional experimental results for the analyses in Section 5. Specifically, Table B.6 presents the results for Section 5.1, Table B.7 presents the results for Section 5.2, and Table B.8 presents the results for Section 5.3.

Table B.6 presents LLM performance under this coarse-grained setting in both Basic and Higher Education. Compared with the full-profile baseline in the main text, *Validity* changes much less than Adaptivity. Qwen3-4B-2507 (CoT) shows a marginal improvement (+0.2%) in Validity.

We attribute this pattern to the lower cognitive load. Complex mastery constraints typically require models to deviate from standard curricular paths to accommodate specific gaps. Removing these constraints allows the models, especially smaller ones, to revert to canonical teaching sequences, which are frequent in pretraining data and often logically valid. This suggests that fine-grained modeling is important for personalization but not necessarily for generic logical correctness.

B.4 Retrieval Configuration

To investigate whether retrieval-augmented generation (RAG) can improve KC learning path planning, we augment the base model Qwen3-4B-2507$_{Zero-shot}$ with a RAG pipeline and evaluate it in the Basic Education setting. Specifically, we use `BAAI/bge-small-zh-v1.5` as the embedding model with L2-normalized vectors and FAISS with inner-product search (equivalent to cosine similarity) for retrieval. For each query, the top-3 most relevant knowledge chunks are retrieved and prepended to the prompt as reference context.

Table 5 in the main text reports all four evaluation dimensions for this comparison.

C Cross-Language and Cross-Curriculum Transferability

While our current instantiation uses Chinese educational textbooks, the structural framework of PersonaPath can in principle be adapted to other languages and curricula. The hierarchical knowledge graph construction pipeline, the IRT-based student simulator, and the three evaluation dimensions (Validity, Adaptivity, Efficiency) are not tied to a specific language, but they would require curriculum-specific calibration and validation. To adapt Per-

	Basic Education (#1000)				**Higher Education** (#1000)			
	Validity Pass Rate	Adaptivity Pass Rate	Efficiency Pass Rate	Final Pass Rate	Validity Pass Rate	Adaptivity Pass Rate	Efficiency Pass Rate	Final Pass Rate
Qwen3-4B-2507$_{CoT}$	56.0	21.1	50.9	7.4	26.7	25.0	29.2	5.7
Pangu-7B$_{CoT}$	22.7	16.8	45.1	1.4	20.7	35.6	40.7	3.6
Llama-3.1-8B$_{CoT}$	9.6	6.8	7.7	0.8	3.2	12.0	3.1	0.4
DeepSeek-R1-Distill-14B$_{CoT}$	49.8	26.3	56.2	10.0	37.0	32.9	43.0	7.7
Qwen3-30B-A3B$_{CoT}$	67.8	18.6	53.0	8.5	44.36	32.3	45.3	9.5

Table B.6: Performance of LLMs on PersonaPath with coarse-grained profiles that omit knowledge mastery.

	Basic Education (#1000)				**Higher Education** (#1000)			
	Validity Pass Rate	Adaptivity Pass Rate	Efficiency Pass Rate	Final Pass Rate	Validity Pass Rate	Adaptivity Pass Rate	Efficiency Pass Rate	Final Pass Rate
Qwen3-4B-2507$_{CoT}$	12.6	34.3	37.4	3.5	7.3	43.5	12.7	1.5
Pangu-7B$_{CoT}$	2.7	27.7	22.5	1.8	2.7	37.4	7.1	0.1
Llama-3.1-8B$_{CoT}$	9.6	6.8	7.7	0.9	2.5	30.0	1.5	0.0
DeepSeek-R1-Distill-14B$_{CoT}$	12.5	25.6	29.1	4.5	3.9	34.4	12.5	0.8
Qwen3-30B-A3B$_{CoT}$	15.9	25.4	23.7	6.5	3.6	36.3	3.5	0.2

Table B.7: Performance of LLMs on PersonaPath in the presence of contextual noise.

	Basic Education (#1000)				**Higher Education** (#1000)			
	Validity Pass Rate	Adaptivity Pass Rate	Efficiency Pass Rate	Final Pass Rate	Validity Pass Rate	Adaptivity Pass Rate	Efficiency Pass Rate	Final Pass Rate
Qwen3-4B-2507$_{CoT}$	86.6	20.6	38.1	5.7	52.4	30.0	15.5	4.9
Pangu-7B$_{CoT}$	48.0	24.0	16.0	5.4	32.4	21.5	9.7	3.6
Llama-3.1-8B$_{CoT}$	39.9	4.9	6.2	0.5	28.1	9.3	7.1	0.5
DeepSeek-R1-Distill-14B$_{CoT}$	77.9	19.9	23.3	4.2	57.4	24.9	20.2	4.6
Qwen3-30B-A3B$_{CoT}$	84.4	15.9	30.2	3.6	53.3	23.3	12.7	3.6

Table B.8: Performance of LLMs on PersonaPath with one-shot path generation.

sonaPath to an English-language curriculum such as the US K-12 Common Core, a researcher would (1) replace the source corpus with standard English textbooks, (2) employ an LLM to extract the Textbook-Unit-Concept hierarchy from the new corpus, and (3) validate the prerequisite dependencies with domain experts. The remaining pipeline, including student simulation, path planning, and evaluation, could then be reused after such adaptation.

D Knowledge-Level Approaches

This section supplements Section 2 with the relationship between learner-state estimation and knowledge-path planning. Knowledge Tracing and Cognitive Diagnosis characterize what a learner currently knows, providing state estimates that can inform a planning component.

Knowledge Tracing (KT) estimates and dynamically updates a learner's mastery over each knowledge component from interaction sequences. Early probabilistic models such as Bayesian Knowledge Tracing track each skill with a small set of latent parameters (Corbett and Anderson, 1995), and later deep models improve predictive accuracy with recurrent, memory, and attention architectures (Piech et al., 2015; Zhang et al., 2017; Pandey and Karypis, 2019). Cognitive Diagnosis infers a learner's latent proficiency on fine-grained concepts from response data (Cheng et al., 2019; Lord, 2012). A system combining KT/CD with a curriculum planner can use these estimates to select subsequent content. PersonaPath evaluates the planning component with an explicitly supplied mastery vector: the output is an ordered knowledge path, assessed for prerequisite validity, learner adaptation, and goal-directed efficiency. Applying a mastery estimator and a planner together would additionally involve aligning their state representations and specifying the planning policy.

E Error Case Analysis

To provide an intuitive understanding of how learning paths can violate evaluation constraints, we construct three representative error cases based on the same persona and target: an *Average-performing* learner targeting *Math-Primary-Grade4B-Four Operations*. Keeping the learner profile and goal fixed allows the cases to isolate different failure modes in the generated path.

Case 1: Validity failure. The first case is a prerequisite violation. During the first two rounds, the model correctly traverses prerequisite content in *Math-Primary-Grade3B*. However, at Round 3, the learner still has three unmastered units in Grade-3B, and *Math-Primary-Grade4A* has not been learned at all. Instead of continuing through these prerequisites, the model jumps directly to the target textbook *Math-Primary-Grade4B*. This exposes the learner to "Four Operations" before the necessary foundations in decimal arithmetic, multi-digit multiplication, and Grade-4A content have been established.

Case 2: Adaptivity failure. The second case shows a mismatch between concept difficulty and learner mastery. After Round 1, the learner's mastery of the current unit reaches 0.75, so a suitable next concept should be closer to this level. Instead, the model selects *Oral Division Basics* with difficulty $d = 0.1$, yielding a large Cog-Gap of 0.65. The same pattern appears again in later rounds: the model repeatedly chooses the easiest available concept even when a moderately harder concept would better match the learner's current mastery. For an *Average-performing* learner, such choices provide little cognitive challenge and waste learning opportunities.

Case 3: Efficiency failure. The third case illustrates inefficient planning caused by repeated over-challenging selections. The model consistently selects the hardest available concept within each unit. For example, at Round 1 the learner's mastery is 0.53, but the model assigns a concept with difficulty $d = 0.4$. This choice yields only a small mastery gain, whereas a better matched concept with $d = 0.2$ would move the learner much faster toward the unit threshold. Because the model keeps repeating this pattern, progress continuity breaks down and the path consumes most of the step budget before entering later prerequisites. By Round 20, the learner remains in Grade-3B and never reaches the target textbook.

Figures E.3–E.5 show the corresponding model-generated outputs.

F Prerequisite Graph Presentation

We illustrate the prerequisite graphs generated for the collected textbooks. As formalized in Section 3.2, these directed acyclic structures specify the order in which agents must navigate textbook-level resources.

Case 1: Validity Violation ✗ – Model Output (Partial)

```
// Round 1-2: Correct prerequisite traversal in Grade-3B
{ "round": 1, "book": "Math-Primary-Grade3B",
  "unit": "Division by One-Digit Divisors",
  "sub_node": "Pen Division",
  "mastery_old": 0.5255, "mastery_new": 0.7522, "difficulty": 0.2,
  "book_reason": "This textbook is the most fundamental
    prerequisite for the target textbook" }


{ "round": 2, "book": "Math-Primary-Grade3B",
  "unit": "Two-Digit by Two-Digit Multiplication",
  "sub_node": "Pen Multiplication Rules",
  "mastery_old": 0.4437, "mastery_new": 0.7270, "difficulty": 0.3,
  "book_reason": "This textbook is the most fundamental
    prerequisite for the target textbook" }

// Round 3: PREREQUISITE VIOLATION
// State: Grade-3B still has 3 unmastered units; Grade-4A is entirely unmastered.
// Expected: continue Grade-3B, then enter Grade-4A before Grade-4B.


{ "round": 3,
  "book": "Math-Primary-Grade4B",
  "unit": "Four Operations",
  "sub_node": "Operation Rules Application",
  "mastery_old": 0.4951, "mastery_new": 0.7218, "difficulty": 0.2,
  "book_reason": "The student has basically mastered
    the prerequisite content and can start the target textbook",
  "diagnosis": "Invalid jump: prerequisite textbooks are skipped.",
  "concept_reason": "Operation Rules Application with difficulty
    0.2 has the smallest gap to mastery 0.4951" }
```

Figure E.3: Case 1: Validity violation. The orange highlight marks the invalid jump to the target textbook; blue marks the unmet prerequisite state; purple marks the expected prerequisite traversal.

Case 2: Adaptivity Violation ✗ – Model Output (Partial)

```
// Round 1: Reasonable selection (Cog-Gap = |0.53 - 0.2| = 0.33)
{ "round": 1, "book": "Math-Primary-Grade3B",
  "unit": "Division by One-Digit Divisors",
  "sub_node": "Pen Division",
  "mastery_old": 0.5255, "mastery_new": 0.7522, "difficulty": 0.2,
  "concept_reason": "Pen Division with difficulty 0.2 has
    the smallest gap to mastery 0.5255" }


// Round 2: ADAPTIVITY VIOLATION
// Mastery = 0.7522, selected d = 0.1, Cog-Gap = 0.6522.
// Better match: choose a harder available concept, e.g., d = 0.2.
{ "round": 2, "book": "Math-Primary-Grade3B",
  "unit": "Division by One-Digit Divisors",
  "sub_node": "Oral Division Basics",
  "mastery_old": 0.7522, "mastery_new": 0.9788,
  "difficulty": 0.1,
  "diagnosis": "Too easy for the learner's current mastery.",

  "concept_reason": "Oral Division Basics with difficulty 0.1
    is the simplest concept, suitable for consolidation" }

// Round 3: ADAPTIVITY VIOLATION
// Mastery = 0.6486, selected d = 0.1, Cog-Gap = 0.5486.
// Better match: available d = 0.4 gives a smaller gap of 0.2486.
{ "round": 3, "book": "Math-Primary-Grade3B",
  "unit": "Year, Month and Day",
  "sub_node": "Calendar Basic Knowledge",

  "mastery_old": 0.6486, "mastery_new": 0.6571,
  "difficulty": 0.1,
  "diagnosis": "The model again chooses the easiest concept.",
  "concept_reason": "Select the most basic concept
    to help the student build a solid foundation" }

// Round 5: ADAPTIVITY VIOLATION
// Mastery = 0.4783, selected d = 0.1, Cog-Gap = 0.3783.
// Better match: Decimal Arithmetic with d = 0.2.


{ "round": 5, "book": "Math-Primary-Grade3B",
  "unit": "Introduction to Decimals",
  "sub_node": "Decimal Basic Knowledge",
  "mastery_old": 0.4783, "mastery_new": 0.4868,
  "difficulty": 0.1,
  "diagnosis": "Repeated under-challenge inflates Cog-Gap.",
  "concept_reason": "Decimal Basic Knowledge with difficulty 0.1
    is the simplest, suitable for introductory learning" }
```

Figure E.4: Case 2: Adaptivity violation. Orange highlights repeated under-challenging selections; blue shows the learner mastery and resulting Cog-Gap; purple indicates the better matched alternatives.

Case 3: Efficiency Violation ✗ – Model Output (Partial)

```
// Round 1: EFFICIENCY VIOLATION
// Mastery = 0.5255, but the model selects the hardest concept d = 0.4.
// Better match: Pen Division with d = 0.2 would yield much faster progress.
{ "round": 1, "book": "Math-Primary-Grade3B",
  "unit": "Division by One-Digit Divisors",
  "sub_node": "Division Word Problems",
  "mastery_old": 0.5255, "mastery_new": 0.5340,

  "difficulty": 0.4,
  "diagnosis": "Only +0.0085 mastery gain; progress is too slow.",
  "concept_reason": "Division Word Problems is the most
    challenging concept, helps the student aim higher" }

// Rounds 2-6: The model keeps choosing the same hard concept.
// Each round produces roughly the same tiny gain; the unit stays far below 0.8.
{ "round": 2, ... "sub_node": "Division Word Problems",
  "mastery_old": 0.5340, "mastery_new": 0.5425,

  "difficulty": 0.4 }
{ "round": 3, ... "mastery_old": 0.5425, "mastery_new": 0.5510 ... }
{ "round": 4, ... "mastery_old": 0.5510, "mastery_new": 0.5595 ... }
{ "round": 5, ... "mastery_old": 0.5595, "mastery_new": 0.5680 ... }
{ "round": 6, ... "mastery_old": 0.5680, "mastery_new": 0.5765 ... }
// Reference: the correct path reaches 0.8 in 2 rounds with d = 0.2.

// Round 15: The first unit finally crosses 0.8.
// Fifteen rounds are consumed by one unit; later prerequisites lose the step budget.

{ "round": 15, "book": "Math-Primary-Grade3B",
  "unit": "Division by One-Digit Divisors",
  "sub_node": "Division Word Problems",
  "mastery_old": 0.7935, "mastery_new": 0.8020, "difficulty": 0.4,
  "concept_reason": "Continue practicing the hardest concept
    to fully master division" }

// Round 16: Same pattern in the next unit.
{ "round": 16, "book": "Math-Primary-Grade3B",
  "unit": "Two-Digit by Two-Digit Multiplication",

  "sub_node": "Multiplication Word Problems",
  "mastery_old": 0.4437, "mastery_new": 0.4522,
  "difficulty": 0.4,
  "diagnosis": "Over-challenging selection repeats in a new unit.",
  "concept_reason": "Select the most challenging multiplication
    concept to push the student's limits" }

// Round 20: GOAL NOT REACHED
// Still in Grade-3B; Grade-4A and the target textbook are never reached.

{ "round": 20, "book": "Math-Primary-Grade3B",
  "unit": "Two-Digit by Two-Digit Multiplication",
  "sub_node": "Multiplication Word Problems",
  "mastery_old": 0.4862, "mastery_new": 0.4947,
  "difficulty": 0.4,
  "diagnosis": "Step budget exhausted before reaching the target." }
```

Figure E.5: Case 3: Efficiency violation. Orange marks over-challenging repeated choices and slow mastery updates; blue marks the accumulated progress problem; purple marks the faster reference choice.

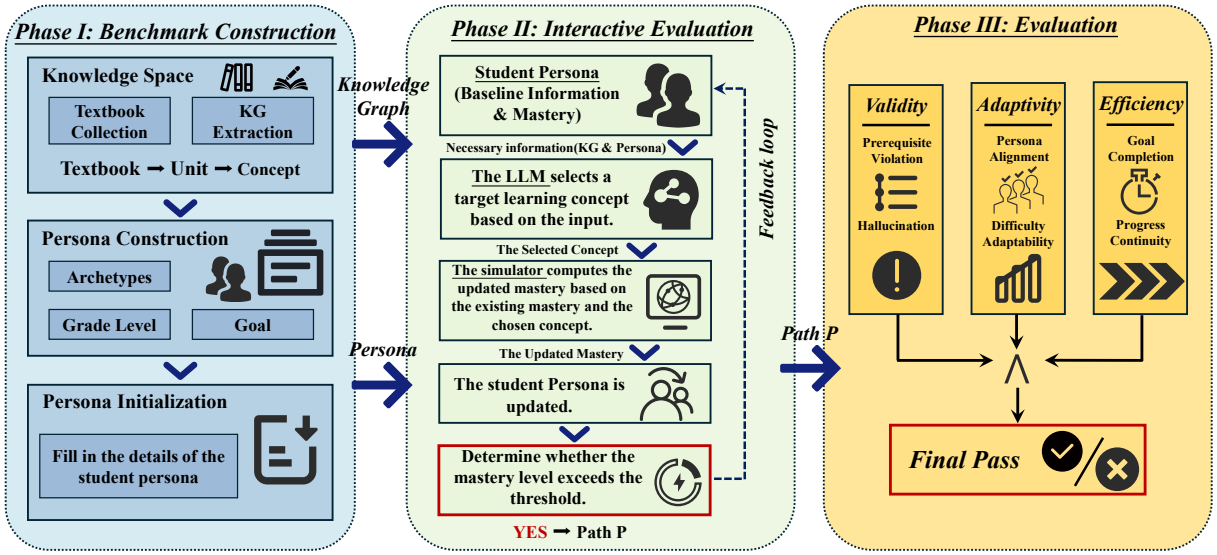


Figure F.6: Overview of the entire **PersonaPath** workflow. **Phase I**: A hierarchical knowledge graph and diverse learner personas are constructed from authoritative textbooks. **Phase II**: The LLM agent interacts with the environment in a step-by-step loop, selecting concepts and receiving mastery updates until the target proficiency is reached. **Phase III**: The generated learning path is evaluated across three constraint dimensions (Validity, Adaptivity, and Efficiency), whose conjunction determines the Final Pass Rate.

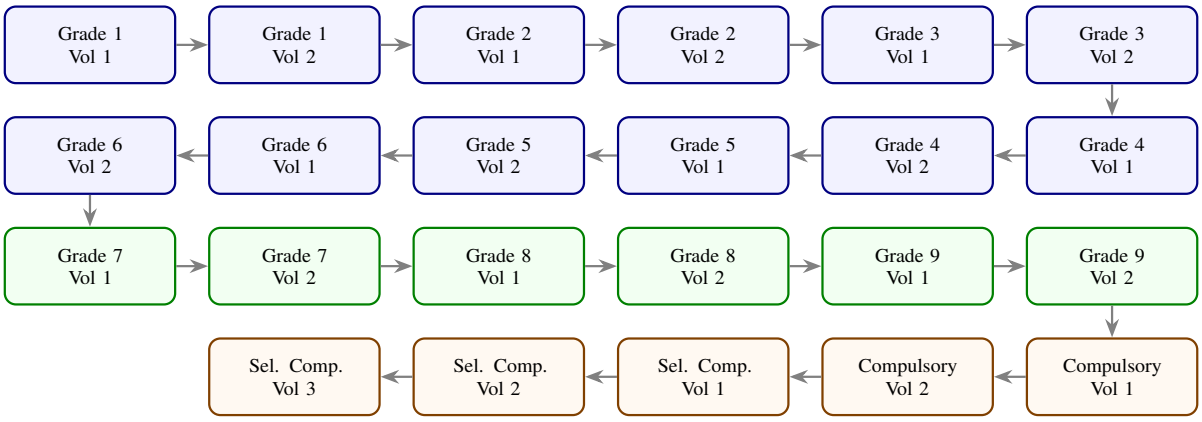


Figure F.7: The learning sequences for Chinese and Mathematics in basic education.

G Prompt Templates

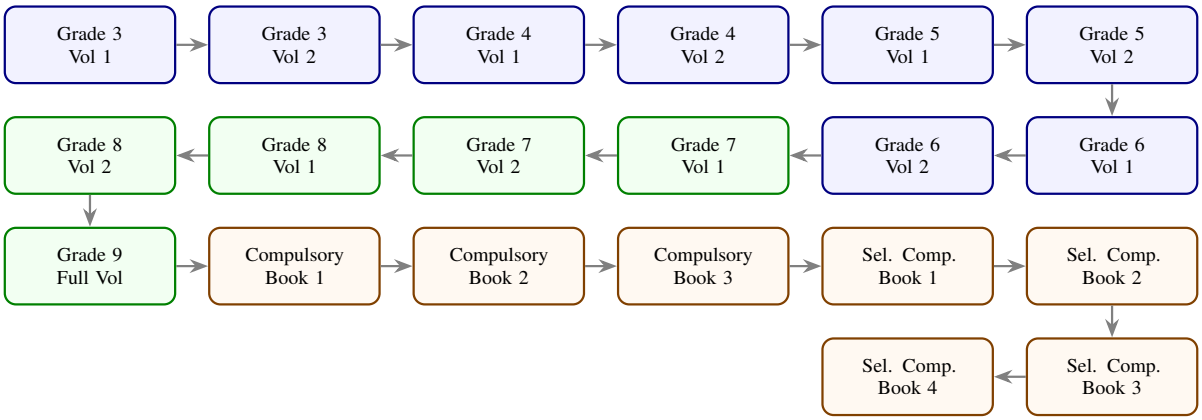


Figure F.8: The learning sequence for English in basic education.

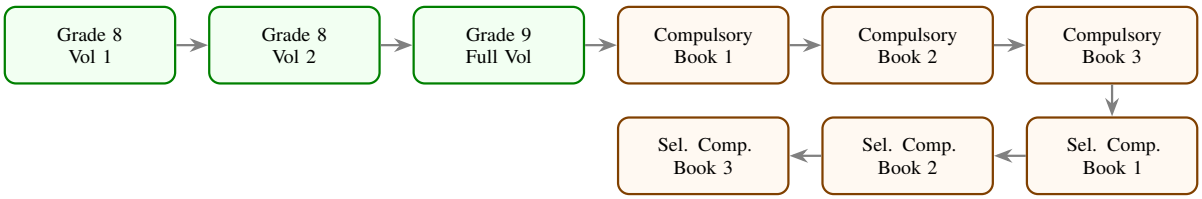


Figure F.9: The learning sequence for Physics in basic education.

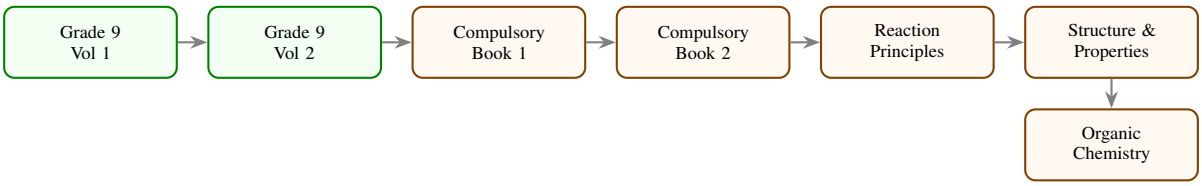


Figure F.10: The learning sequence for Chemistry in basic education.

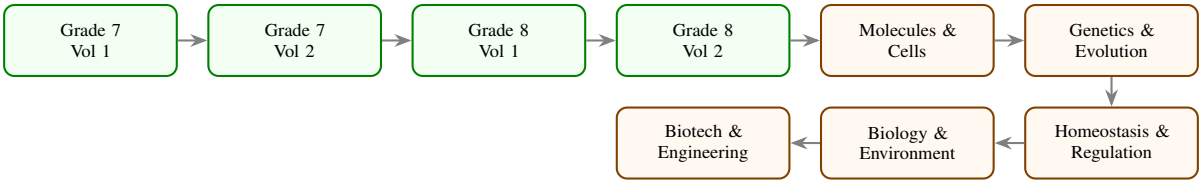


Figure F.11: The learning sequence for Biology in basic education.

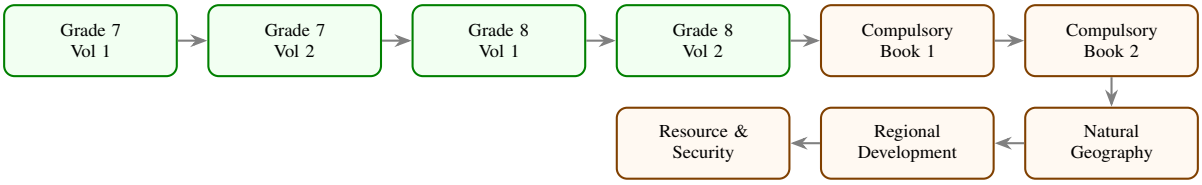


Figure F.12: The learning sequence for Geography in basic education.

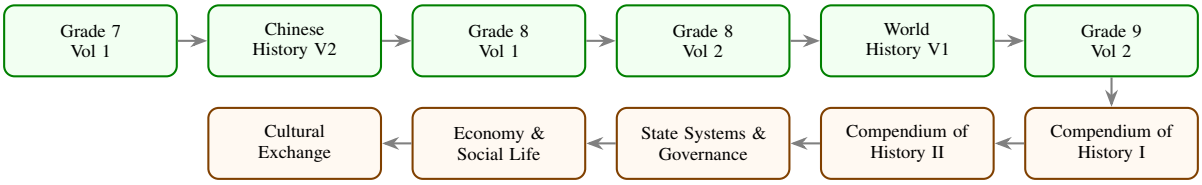


Figure F.13: The learning sequence for History in basic education.

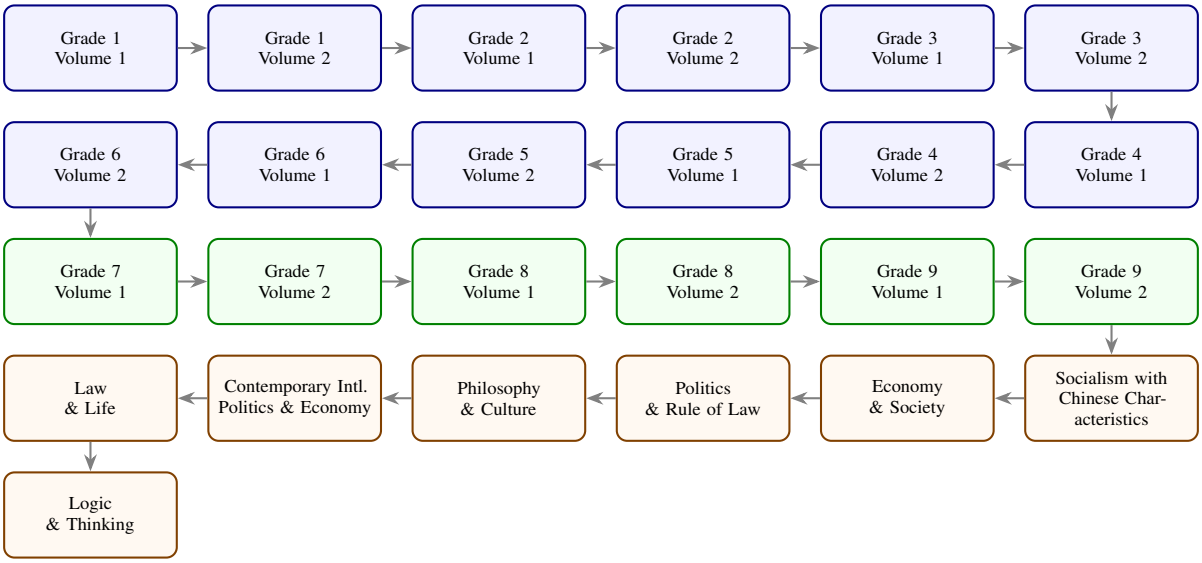


Figure F.14: The learning sequence for Morality and Rule of Law / Ideological and Political Education in basic education.

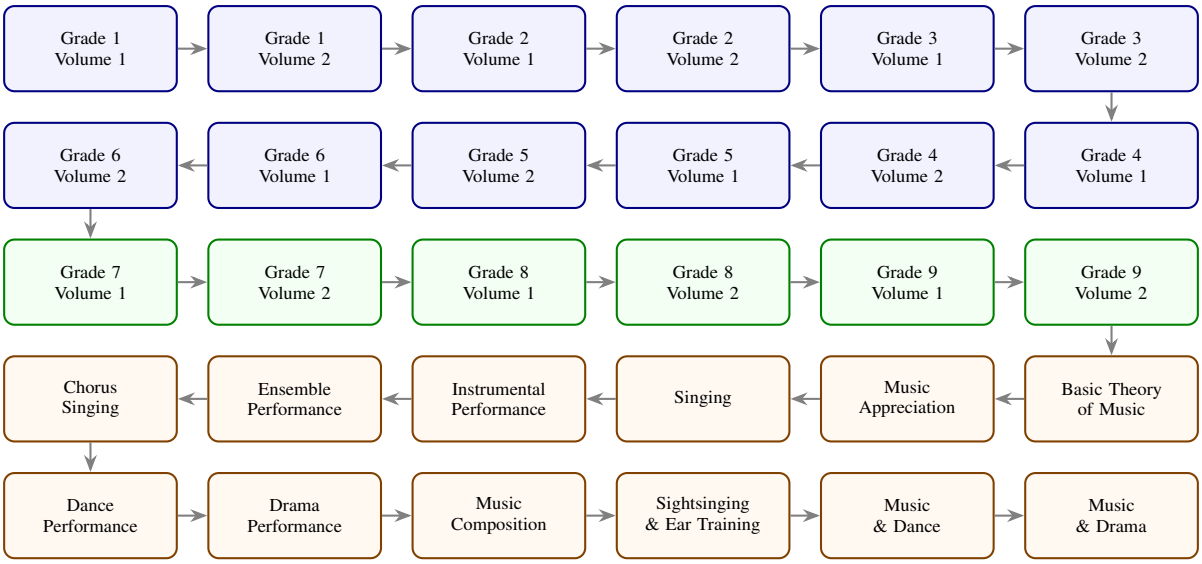


Figure F.15: The learning sequence for Music in basic education.

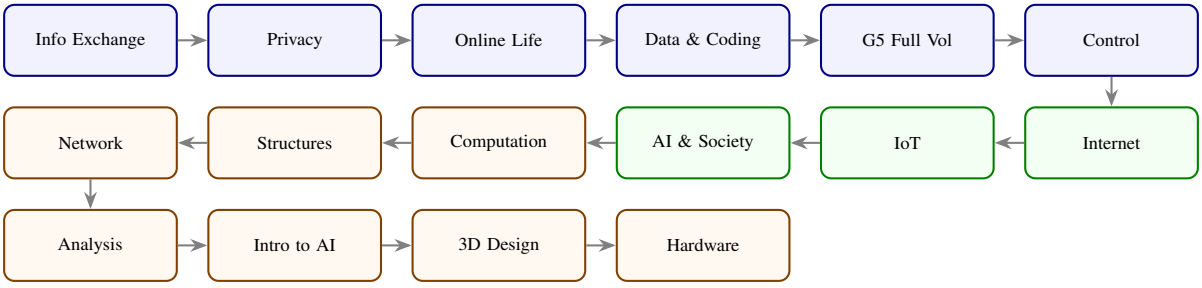


Figure F.16: The learning sequence for Information Science / Information Technology in basic education.

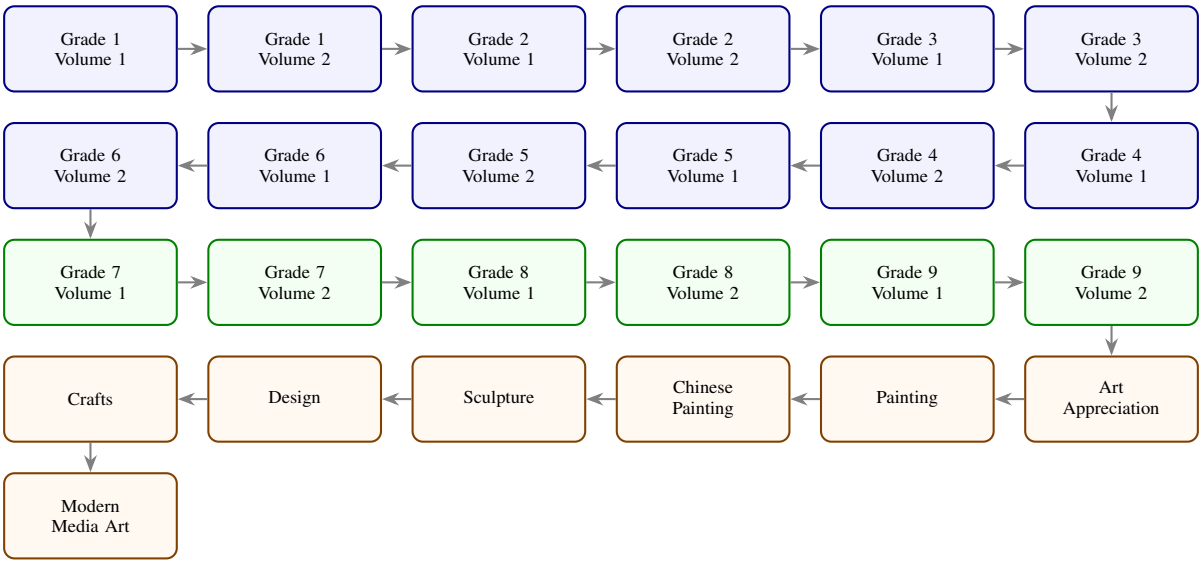


Figure F.17: The learning sequence for Art in basic education.

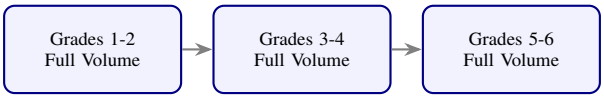


Figure F.18: The learning sequence for Physical Education and Health in basic education.

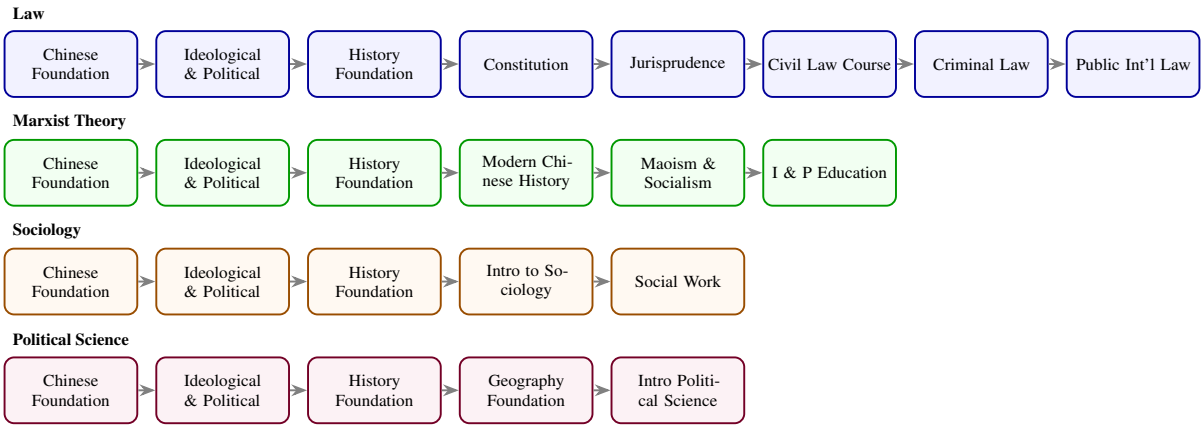


Figure F.19: Learning sequences of four sub-disciplines under the Law primary discipline: Law (blue), Marxist Theory (green), Sociology (orange), and Political Science (purple).



Figure F.20: Learning sequences of fifteen Engineering sub-disciplines in higher education.

Prompt for Knowledge Graph Generation

System Prompt:

```
You are a senior education expert in {subject} and a Knowledge Graph architect. Your task is to
 construct a high-precision structured knowledge graph based on the provided content from the
textbook "{book_name}".
```

User Prompt:

```
[Core Task]
Read the [Textbook Excerpts] at the end and output a JSON object containing a "Knowledge Tree"
and "Relationship Triplets".

[Key Requirements: Catalog-Driven Structure]
1. Strictly follow the catalog hierarchy: Identify structures such as Chapters, Sections, and
Subheadings.
   - Level 1 children should correspond to "Chapters" or "Units".
   - Level 2 children should correspond to "Sections" or "Lessons".
   - Level 3 and below should correspond to specific knowledge points, concepts, formulas, or
theorems.
2. Granularity Control:
   - Ensure the catalog structure is complete; do not skip chapters.
   - Refine knowledge points under Level 2/3 (e.g., "Positive Numbers" should include children
like "Definition", "Representation", "Real-world Examples").
   - Maximum tree depth: 5 layers.

[Field Definitions]
1. knowledge_tree: {{"title": "...", "content": "...", "difficulty": "1-5", "children": []}}
   - Difficulty: 1 (Easy/Basic), 3 (Medium/Comprehensive), 5 (Hard/Advanced).
2. triplets: {{"head": "A", "tail": "B", "relation": "..."}}
   - Focus on non-hierarchical logic (Prerequisite, Application, Attribute, Comparison).

[Data Consistency]
- Terminology must be accurate for {subject}.
- Every node must have: "title", "content", "difficulty", "children".
- "content" should be a concise summary of the core definition or formula.

Output example:
{{
  "knowledge_tree": {{
    "title": "Rational Numbers",
    "children": [
      {{"title": "Addition of Rational Numbers", "content": "Rules for adding...", "difficulty
": "2", "children": []}}
    ]
  }},
  "triplets": [
    {{"head": "Number Line", "tail": "Comparison", "relation": "Used for"}}
  ]
}}

Textbook Excerpts:
{text}
```

Figure G.21: The prompt for Knowledge Graph Generation

Prompt for Textbook Recommendation (CoT)

System Prompt:

```
You are a curriculum planning consultant responsible for recommending the textbook that a
student should study right now. Your thinking process must be concise and directly lead to a
decision.

[Core Rules]
- The learning sequence must start from the most basic and prerequisite textbooks, progressing
step-by-step according to a natural pedagogical order.
- If there are unmastered prerequisite textbooks: Select the one that appears earliest in the
learning sequence.
- If there are no unmastered prerequisite textbooks: Select the target textbook itself.
- Learning sequence: Follow the natural progression of grades and semesters (e.g., Grade 1, 2,
3... Vol. 1, Vol. 2).

[Thinking Steps]
1. Analyze the prerequisite requirements of the target textbook "{goal_book}".
2. Check if these prerequisite textbooks are in the student's "unmastered list."
3. Identify the earliest gap in the learning path.

[Output Format]
Your response must include two sections: [Thinking] and [Output]. Each thinking step must not
exceed 3 sentences.

[Thinking] ♦ Here, we employ Chain-of-Thought (CoT) prompting.
Step 1: The prerequisite textbooks for the target textbook "{goal_book}" are...
Step 2: Check whether these prerequisites are in the student's "unmastered list"...
Step 3: The earliest gap in the learning path is...

[Output]
{{"recommended_book": "Textbook Name", "reason": "Explanation"}}
```

User Prompt:

```
# Student Status
- Mastered: {mastered_str}
- Unmastered: {learning_str}, {not_mastered_str}

# Target Textbook
"{goal_book}"

# Your Task
Based on logical progression, select the prerequisite textbook from the [Unmastered] list that
is [earliest in the learning sequence and most fundamental] for the target textbook "{goal_book
}".

Now, please complete this task and output the result in JSON format as follows:
{{"recommended_book": "Textbook Name", "reason": "Explanation"}}
```

Figure G.22: The prompt for textbook recommendation (CoT). The text highlighted in blue denotes the CoT instruction.

Prompt for Textbook Recommendation (Zero-shot)

System prompt:

```
You are a curriculum planning consultant. The student's goal is to learn the textbook "{
goal_book}". You are responsible for recommending the textbook that the student should study
right now.

[Core Rules]
- The learning sequence must start from the most basic and prerequisite textbooks, progressing
step-by-step according to a natural pedagogical order.
- If there are unmastered prerequisite textbooks: Select the one that appears earliest in the
learning sequence.
- If there are no unmastered prerequisite textbooks: Select the target textbook itself.
- Learning sequence: Follow the natural progression of grades and semesters (e.g., Grade 1,
Grade 2, Grade 3... Vol. 1, Vol. 2).

[Important Principle]
- When there are multiple unmastered prerequisite textbooks, select the [earliest one in the
learning sequence].

Now, please complete this task and output the result in JSON format as follows:
{{"recommended_book": "Textbook Name", "reason": "Explanation"}}
```

User Prompt:

```
# Student Status
- Mastered: {mastered_str}
- Unmastered: {learning_str}, {not_mastered_str}

# Target Textbook
"{goal_book}"

# Your Task
Based on logical progression, select the prerequisite textbook from the [Unmastered] list that
is [earliest in the learning sequence and most fundamental] for the target textbook "{goal_book
}".

Now, please complete this task and output the result in JSON format as follows:
{{"recommended_book": "Textbook Name", "reason": "Explanation"}}
```

Figure G.23: The prompt for textbook recommendation (Zero-shot)

Prompt for Selecting Concepts (CoT)

System Prompt:

```
You are a learning path planning assistant. Your thinking process must be concise and directly
lead to a decision.

[Rules Summary]
- You must consider the student type and their current mastery level to select a knowledge
point.
- Regular students should select the sub-knowledge point with a difficulty level closest to
their "Current Mastery Level."
- For students with a weak foundation or exceptional students, consider selecting knowledge
points with slightly lower or slightly higher difficulty, respectively.

[Thinking Constraints]♦ Here, we employ Chain-of-Thought (CoT) prompting.
1. Confirm the student type (Regular/Weak/Exceptional) and their current mastery level.
2. Compare the difference between the difficulty of available knowledge points and the student'
s mastery level.
3. Lock in the optimal knowledge point according to the rules (closest/slightly lower/slightly
higher).

[Output Format]
Your response must include two sections: [Thinking] and [Output]. Each thinking step must not
exceed 2 sentences.

[Output Requirements]
[Thinking]
Q1: Student type is [Type], Mastery level is [Value].
Q2: Comparison of knowledge point difficulties...
Q3: Selected [Knowledge Point Title].

[Output]
{{"sub_node": "Knowledge Point Title", "reason": "Explanation for the selection"}}
```

User Prompt:

```
# Learning Content
- Textbook: "{book_name}"
- Unit: {parent_node}
- Ultimate Goal: {global_goal}

# Student Status
- Current Mastery Level: {parent_mastery}
- Student Type: {stu_archetype}

# Candidate Sub-knowledge Points List
(Note: The "difficulty" value represents the difficulty level of the knowledge point; higher
values indicate greater complexity.)
{json.dumps(ordered_subnodes, ensure_ascii=False)}

# Task
Please strictly follow the rules defined in the system instructions to select **exactly one**
sub-knowledge point from the list provided above.

Now, complete the task and return only the JSON object in the following format:
{{"sub_node": "The 'title' value of the selected sub-knowledge point", "reason": "Explanation
for the selection"}}
```

Figure G.24: The prompt for selecting concepts (CoT). The text highlighted in blue denotes the CoT instruction.

Prompt for Selecting Concepts (Zero-shot)

System Prompt:

```
You are a learning path planning assistant responsible for selecting the most appropriate sub-
knowledge point from a given list.

[Core Selection Rules]
- Your sole task is to select the sub-knowledge point with the most matching difficulty based
on the "Student Type" and "Current Mastery Level."
- Selection Criteria:
    1. You must consider both the student type and their current mastery level to select a
knowledge point.
    2. For regular students, choose the sub-knowledge point with a difficulty level closest to
their "Current Mastery Level."
    3. For students with a weak foundation or exceptional students, consider selecting
knowledge points with slightly lower or slightly higher difficulty, respectively.

[Output Requirements]
- Output only one valid JSON object in the following format: {{"sub_node": "Sub-knowledge Point
 Title", "reason": "Explanation for the selection"}}.
- If multiple sub-knowledge points meet the criteria, select the first one (based on the order
of input).

[Important Reminders]
- Do not engage in complex reasoning or trade-offs; strictly execute according to the rules
above.
- Your response must be a unique and deterministic JSON object.
```

User Prompt:

```
# Learning Content
- Textbook: "{book_name}"
- Unit: {parent_node}
- Ultimate Goal: {global_goal}

# Student Status
- Current Mastery Level: {parent_mastery}
- Student Type: {stu_archetype}

# Candidate Sub-knowledge Points List
(Note: The "difficulty" value represents the difficulty level of the knowledge point; higher
values indicate greater complexity.)
{json.dumps(ordered_subnodes, ensure_ascii=False)}

# Task
Please strictly follow the rules defined in the system instructions to select **exactly one**
sub-knowledge point from the list provided above.

Now, complete the task and return only the JSON object in the following format:
{{"sub_node": "The 'title' value of the selected sub-knowledge point", "reason": "Explanation
for the selection"}}
```

Figure G.25: The prompt for selecting concepts (Zero-shot)

Prompt without Mastery Information

System Prompt:

```
You are a learning path planning assistant. Your thinking process must be concise and directly
lead to a decision.

[Rules Summary]
- You need to consider difficulty levels when selecting sub-knowledge points.
- Regular students should select sub-knowledge points with moderate difficulty.
- For students with a weak foundation or exceptional students, consider selecting knowledge
points with slightly lower or slightly higher difficulty levels, respectively.

[Thinking Constraints]
1. Confirm the student type (Regular/Weak/Exceptional).
2. Compare the difficulty of available knowledge points with the student type.
3. Lock in the optimal knowledge point based on the rules (moderate/slightly lower/slightly
higher).

[Output Requirements]
[Thinking]
Q1: The student is [Type].
Q2: Comparison of knowledge point difficulties...
Q3: Selected [Knowledge Point Title].

[Output]
{"sub_node": "Knowledge Point Title", "reason": "Explanation for the selection"}

[Output Format]
Your response must include two sections: [Thinking] and [Output]. Each thinking step must not
exceed 2 sentences.
```

User Prompt:

```
# Learning Content
- Textbook: "{book_name}"
- Unit: {parent_node}
- Ultimate Goal: {global_goal}
♦ The Mastery attribute is removed here.

# Student Status
- Student Type: {stu_archetype}

# Candidate Sub-knowledge Points List
(Note: The "difficulty" value represents the difficulty of the knowledge point; a higher value
indicates a more difficult point.)
{json.dumps(ordered_subnodes, ensure_ascii=False)}

# Task
Please strictly follow the rules in the system instructions to select **exactly one** sub-
knowledge point from the list provided above.

Now, please complete this task and return ONLY the JSON object in the following format:
{{"sub_node": "The 'title' value of the selected sub-node", "reason": "Explanation for the
selection"}}
```

Figure G.26: The prompt template for the experimental setting without explicit mastery information. The content highlighted in purple represents the main modifications.

Prompt with Introduced Noise

System Prompt:

```
You are a curriculum planning consultant responsible for recommending the textbook that a
student should prioritize learning at this moment. Your thinking process must be concise and
directly lead to a decision.

[Core Rules]
- The learning sequence must start from the most fundamental and prerequisite textbooks,
progressing step-by-step according to a natural pedagogical order.
- If there are unmastered prerequisite textbooks: Select the one that appears earliest in the
learning sequence.
- If all prerequisites have been mastered: Select the target textbook itself.
- Learning sequence: Follow the natural progression of grades and semesters (e.g., Grade 1, 2,
3... Vol. 1, Vol. 2).

[Thinking Steps]
1. Analyze the prerequisite requirements of the target textbook "{goal_book}".
2. Check if these prerequisites are in the student's "unmastered list."
3. Identify the earliest gap in the learning path.

[Output Format]
Your response must include two sections: [Thinking] and [Output]. Each thinking step must not
exceed 3 sentences.

[Thinking]
Step 1: The prerequisites for the target textbook "{goal_book}" are...
Step 2: Check whether these prerequisites are in the student's "unmastered list"...
Step 3: The earliest gap in the learning sequence is...

[Output]
{{"recommended_book": "Textbook Name", "reason": "Explanation for the recommendation"}}
```

User Prompt:

```
# Student Status
- Mastered: {mastered_str}
- Unmastered: {books}♦ We injected extraneous information unrelated to the learning goal here.

# Target Textbook
"{goal_book}"

# Your Task
Based on logic, for the target textbook "{goal_book}", select the prerequisite textbook from
the [Unmastered] list that is [earliest in the learning sequence and most fundamental].

Now, please complete this task and output a JSON object in the following format:
{{"recommended_book": "Textbook Name", "reason": "Explanation for the recommendation"}}
```

Figure G.27: Prompt with introduced noise. The text highlighted in purple represents the injected noise.

Single-pass Planning Prompt

System Prompt:

```
You are a holistic Curriculum Planning Expert. Your task is to generate a complete learning
path for a student, spanning from their "Current Status" to the "Target Achievement."

[Environment Rules]
1. Learning Hierarchy: Book -> Unit -> Sub-node.
2. Objective: Ensure the mastery level of the target unit reaches above 0.8.
3. Mechanisms:
   - Prerequisite books must be studied before the target books.
   - Prerequisite units must be studied before subsequent units.
   - Within each unit, multiple sub-nodes must be learned to achieve mastery of that unit.
   - Sub-nodes have difficulty levels; a higher value indicates a more challenging point (
difficulty: 0.0-1.0).

[Student Archetypes & Strategies]
- Students with a weak foundation: Should select sub-nodes with lower difficulty and require
more practice sessions.
- Regular students: Should select sub-nodes with moderate difficulty that increase
progressively.
- Exceptional students: Can select sub-nodes with slightly higher difficulty levels.

[Output Format]
Please output a strict JSON list, where each item represents a single learning step.
Format as follows:
[
    {"step": 1, "book": "Book A", "unit": "Unit 1", "sub_node": "Knowledge Point X", "reason":
"Building fundamentals"},
    {"step": 2, "book": "Book A", "unit": "Unit 1", "sub_node": "Knowledge Point Y", "reason":
"Progressive advancement"},
    ...
]
```

User Prompt:

```
# 1. Learning Goals
Ultimate Goal: Master the unit [goal_unit] in the textbook "{goal_book}".

# 2. Student Profile
- Archetype: {profile.archetype}
- Currently Mastered Books: {mastered_str}
♦ Here, we input the comprehensive persona details to guide the model's planning.

# Available Learning Resources:
{json.dumps(curriculum_data, ensure_ascii=False)}

Please generate a complete learning path in the following format:
[
    {{"step": 1, "book": "Book Name 1", "unit": "Unit Name 1", "sub_node": "Knowledge Point X",
 "reason": "Building fundamentals"}},
    {{"step": 2, "book": "Book Name 1", "unit": "Unit Name 2", "sub_node": "Knowledge Point Y",
 "reason": "Progressive advancement"}},
    ...
]
```

Figure G.28: Single-pass Planning Prompt. The content highlighted in purple represents the input of the complete persona information.