PathBuilder: A Quality-Controlled LLM System for Personalized Learning Pathways

Jasper Meynard P. Arana[1-4], John Andrew Manacop[3], John Allen Manacop[3], Roy Andrew Garcia[3], Keith Rick Piniera[3], Kristine Ann M. Carandang[1,2], Ethan Robert Casin[1,2], Christian Alis[1,2], Christopher Monterola[1,2]

[1] Asian Institute of Management, Philippines,
[2] ACCeSs@AIM, [3] Makarius Smart Learning, [4] Adamson University

Abstract

Large language models (LLMs) enable scalable content generation for personalized learning, but reliability and pedagogical alignment remain open challenges. We present PathBuilder, a web-based system that integrates expert-validated assessment, retrieval-augmented generation (RAG), and an LLM-as-a-Judge validation loop within a closed instructional pipeline. The system uses a 17,758-item curriculum-aligned question bank, including 1,018 expert-approved LLM-generated items, to construct diagnostic and post-tests for fine-grained learner profiling. In a real-world deployment with 179 registered users (75 matched learners), PathBuilder achieved a mean absolute gain of 37.9 percentage points, Hake's normalized gain of 0.760, and a large effect size (Cohen's d = 0.98). A controlled study of the judge mechanism showed consistent high-quality instructional outputs with a 100% threshold pass rate. These results demonstrate that structured curriculum alignment combined with retrieval grounding and automated validation can support reliable LLM-based personalization in deployed learning systems. A live demonstration of PathBuilder is available at https://demo.pathbuilderedu.com.

1 Introduction

Personalized learning systems aim to adapt instruction to individual learners' knowledge states, cognitive levels, and progression trajectories. Early intelligent tutoring systems achieved personalization through rule-based policies and probabilistic student modeling frameworks such as Bayesian Knowledge Tracing (Corbett and Anderson, 1994; Piech et al., 2015). More recently, large language models (LLMs) have enabled flexible content generation, automated feedback, and adaptive scaffolding in educational settings (Kasneci et al., 2023; Macina et al., 2023; Liermann et al., 2024; Elkins et al., 2024). However, while LLMs substantially increase generative capacity, they also introduce reliability concerns, including hallucinated content, factual inaccuracies, and pedagogical misalignment.

In parallel, research in automatic question generation has demonstrated that LLMs can produce fluent and diverse assessment items (Scaria et al., 2024; Hwang et al., 2024). Yet most prior work evaluates generated questions using automatic metrics or small-scale human annotation, rather than deploying them inside live adaptive systems with measurable learner outcomes (Laban et al., 2022). Similarly, adaptive learning platforms combine diagnostic testing with content sequencing (Scarlatos et al., 2025; Lu and Wang, 2024), but few systems integrate structured curriculum tagging, controlled generative expansion, retrieval grounding, and automated verification into a single production pipeline. Consequently, there remains a gap between generative capability and system-level reliability in real educational deployments.

To address this gap, this paper presents PathBuilder, a web-deployed, quality-controlled LLM system for personalized learning pathways. PathBuilder combines (1) expert-validated, hierarchically tagged diagnostic assessment, (2) controlled LLM-based question expansion, (3) retrieval-augmented generation (RAG) for personalized instructional synthesis, and (4) an LLM-as-a-Judge validation loop that verifies and repairs generated content before delivery to students. Rather than treating generation and evaluation as separate components, PathBuilder integrates them into a closed-loop architecture that enforces curriculum alignment and content reliability at each stage.

PathBuilder models learner knowledge across structured curriculum dimensions and generates personalized instruction using a retrieval-augmented pipeline with automated LLM-based quality control. Rather than relying on one-shot generation, the system integrates expert-aligned

Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations), pages 504–514
July 2-7, 2026 ©2026 Association for Computational Linguistics

assessment, controlled content synthesis, and validation into a closed-loop architecture. We evaluate PathBuilder in a real-world deployment with 179 active users using expert-validated pre- and post-tests to measure learning gains and post-test proficiency. Additionally, we expand the assessment bank through expert-validated LLM-generated questions and structured curriculum classification of existing items, demonstrating both pedagogical impact and system reliability at scale. Our contributions include a quality-controlled personalization framework, a deployed closed-loop LLM architecture, real-world evidence of measurable learning gains, and a validated assessment expansion pipeline for adaptive learning.

2 Related Work

LLMs in Education and Intelligent Tutoring Systems. LLMs have increasingly been explored for tutoring, feedback generation, and adaptive instruction (Schmucker et al., 2024; Reddig et al., 2025; Liermann et al., 2024; Shi et al., 2026). Prior systems demonstrate that LLMs can provide step-by-step explanations and scaffolded hints, but they also highlight risks of factual inaccuracies and pedagogical misalignment. Earlier intelligent tutoring systems (ITS) relied on rule-based or Bayesian student modeling approaches (Koedinger et al., 2013; Piech et al., 2015). Compared to traditional ITS, PathBuilder leverages foundation models for flexible content generation while retaining structured assessment and level estimation mechanisms grounded in expert-validated question banks.

Automatic Question Generation and Assessment. Building on advances in educational NLP, automatic question generation has been widely studied in NLP (Alberti et al., 2019; Yuan et al., 2017; Du et al., 2017), with recent LLM-based approaches improving fluency and diversity. However, prior work often evaluates generated questions using automatic metrics or small-scale human judgments, rather than integrating them into live learning systems (Uto et al., 2023; Ashok Kumar et al., 2023; Dugan et al., 2022; Doostmohammadi et al., 2024). In contrast, PathBuilder incorporates expert-validated generated questions and automatically classified existing items into a deployed assessment pipeline, enabling large-scale empirical evaluation of learning gains.

Personalized Learning Systems. More broadly, adaptive learning platforms combine diagnostic testing with personalized content sequencing (Zhou et al., 2024; Cheng et al., 2024). Recent data-driven systems incorporate reinforcement learning and student modeling for adaptive sequencing (Shen et al., 2024). We extend this line of work by integrating expert-aligned assessment, RAG-based content synthesis, and automated quality control into a unified, web-deployed system with demonstrated real-user learning gains.

3 Methodology

PathBuilder is a quality-controlled LLM system for personalized learning that integrates diagnostic assessment, structured question generation, retrieval-grounded instructional synthesis, and automated verification into a unified pipeline. Rather than treating generation and evaluation as separate components, PathBuilder employs a closed-loop architecture to ensure pedagogical alignment and content reliability. The system operates in three stages: student profiling through structured diagnostic testing, controlled assessment expansion via expert-aligned question generation, and personalized content generation with automated quality control. Each component is detailed in the following subsections.

3.1 Student Profiling via Structured Diagnostic Assessment

Personalization in PathBuilder begins with a diagnostic assessment. Unlike conventional adaptive systems that rely solely on aggregate scores, PathBuilder models student knowledge along three explicit axes: Topic hierarchy T (Topic-Subtopic-Microtopic), Bloom's Taxonomy level B, and Difficulty level D. Each diagnostic item is tagged as:

$$q_i = (t_i, b_i, d_i) \tag{1}$$

where $t_i \in T$ denotes a curriculum-aligned microtopic, b_i denotes cognitive level, and d_i denotes calibrated difficulty. Given student responses $r_i \in 0, 1$, topic-level mastery is computed as:

$$\hat{M}_t = \frac{1}{n_t} \sum_{i \in t} r_i \tag{2}$$

This formulation allows PathBuilder to detect where and at what cognitive depth a student struggles. For example, a student may demonstrate high remembering-level performance but low application-level performance within the same topic. This distinction is critical for instructional targeting.

Instead of merely identifying low-performing students, PathBuilder identifies cognitive progression gaps. The next instructional target level is defined as:

$$B^* = \text{next mastered Bloom Level} \quad (3)$$

Students progress to the next level of Bloom's taxonomy upon successfully passing the post-test. This structured profiling step forms the foundation for both question selection and content generation, ensuring that personalization remains curriculum-grounded rather than heuristic-driven.

3.2 Expert-Aligned Question Generation Pipeline

Assessment quality directly affects personalization reliability. For this reason, PathBuilder does not rely solely on generative models for question creation. Instead, it begins with an expert-developed seed bank of validated multiple-choice questions derived from high-stakes examinations and institutional test banks.

3.2.1 Hierarchical Curriculum Tagging

Each seed question is annotated using a four-level topic taxonomy (Subject-Topic-Subtopic-Microtopic) derived from the program's official Table of Specifications. This hierarchical structure enables fine-grained topic targeting and balanced test construction. It also allows Bloom's-aligned question selection. The tagging schema ensures that the system's adaptive behavior remains aligned with institutional curriculum requirements.

3.2.2 Controlled LLM-Based Expansion

To increase coverage while preserving pedagogical alignment, PathBuilder uses LLMs to generate question variants. However, generation is strictly conditioned on structured parameters. For each seed question q, the model generates a set:

$$G(q) = \{q'_1, ..., q'_k\} \quad (4)$$

Each generated variant must preserve: Topic t, Target Bloom's level b, and Difficulty level d. This conditioning prevents uncontrolled drift in content scope. The LLM is therefore used as a structured transformer of expert knowledge, rather than as an unconstrained content creator. This design choice is central to PathBuilder: generation operates within expert-defined boundaries, enabling scalable yet controlled assessment expansion.

3.2.3 Human Validation and Reliability Control

Because generated questions form the basis of diagnostic and post-tests, each new item undergoes independent expert review. Questions are evaluated along three dimensions: clarity, factual accuracy, and design quality. Each dimension is rated using a 5-point Likert scale. A question is accepted only if it meets a predefined quality threshold across dimensions. To ensure annotation reliability, PathBuilder computes: within-group agreement r_{wg} for Likert consistency and Krippendorff's α for topic classification agreement. This validation layer ensures that the expanded question bank remains aligned with domain standards before being deployed in adaptive testing.

3.3 Personalized Instructional Generation

Once a student's knowledge profile is computed, the system generates personalized instructional content for the weakest identified topic.

3.3.1 Knowledge-Grounded Generation

LLM-generated instructional content can suffer from hallucinations or conceptual drift. To mitigate these risks, PathBuilder employs a Retrieval-Augmented Generation (RAG) architecture grounded in a curated knowledge base comprising textbooks, lecture notes, and validated instructional materials. As of this writing, the knowledge base contains more than 500 instructional resources. During retrieval, only the top-K results (K = 5), ranked using OpenAI's embedding model, are incorporated into the generation pipeline to ensure relevance and reduce noise.

Given a target topic t, relevant documents are retrieved using embedding similarity:

$$D_t = \text{TopK}_{d \in KB} cos(\phi(t), \phi(d)) \quad (5)$$

The instructional content is then generated as:

$$C = f_{LLM}(D_t, B^*, \text{LearnerType}) \quad (6)$$

Here, B^* controls cognitive depth, while learner type modulates explanation style (e.g., analytical, reflective, active). Importantly, style adaptation occurs only after factual grounding is established through retrieval. This separation between content correctness and pedagogical adaptation is deliberate. It ensures that personalization does not compromise factual integrity. Each generated lesson

includes explicit learning objectives, conceptual explanation, worked examples, and practice exercises. Thus, personalization operates at both the cognitive and stylistic levels.

3.4 LLM-as-a-Judge Quality Control Loop

Even with retrieval grounding, generative models may produce inaccuracies. To prevent unverified content from reaching students, PathBuilder introduces an automated quality-control stage using an LLM-as-a-Judge. The Judge evaluates generated content along two primary dimensions: Correctness (conceptual and mathematical validity) and Factual grounding (alignment with retrieved sources). Scores are assigned on a structured rubric. The final acceptance score is computed as:

$$S = \frac{s_{\text{correctness}} + s_{\text{grounding}}}{2} \tag{7}$$

If the score falls below a predefined threshold, the content is rejected and accompanied by structured feedback. The generator then produces a revised version conditioned on the Judge's feedback:

$$C_{new} = f_{LLM}(D_t, \text{feedback}) \tag{8}$$

This iterative repair mechanism forms a closed-loop system. Content is not displayed to the student until it satisfies quality constraints. By integrating validation directly into the generation pipeline, PathBuilder transforms LLM usage from a one-shot generation paradigm into a self-correcting system.

3.5 System Integration

The whole pipeline is implemented and integrated in the frontend as an API service. This modularity enables real-time personalization while preserving separation of concerns between assessment, generation, and validation. Crucially, personalization decisions are always grounded in structured curriculum tags and verified instructional content. The system therefore combines expert-aligned assessment, controlled generative expansion, retrieval-based grounding, and automated validation into a unified architecture for reliable personalized learning.

4 Results and Discussion

This section evaluates PathBuilder along three dimensions: (1) reliability of the expert-validated assessment backbone, (2) measurable learning gains under real deployment, and (3) evaluation of the LLM-based quality control mechanism. Together, these analyses aim to determine whether the system produces pedagogically aligned content, supports meaningful student improvement, and maintains quality assurance at scale.

4.1 Expert Validation: Question Bank Reliability

To ensure the integrity of the diagnostic and generated question bank, we conducted a structured expert validation study. Six licensed professionals currently teaching in the academe independently evaluated the generated questions for clarity, design quality, factual correctness, topic alignment, difficulty level, and Bloom's taxonomy classification.

The original evaluation set contained 294 questions. However, items with critical rendering failures (e.g., missing answer choices or corrupted formatting) were removed upon inspection, resulting in 261 evaluated questions. These questions were then distributed among the six professionals, divided into seven overlapping subsets, with each subset evaluated by at least three raters to reduce fatigue while maintaining multiple independent evaluations per item.

To assess consistency among raters, we computed within-group agreement (r_{wg}) for Likert-scale metrics (clarity, design quality, and factual correctness) and Krippendorff's α for categorical classifications (topic alignment, difficulty level, and Bloom's taxonomy classification).

Metric	Mean r_{wg}	Interpretation
Clarity	0.859	High Agreement
Design	0.850	High Agreement
Factualness	0.817	Acceptable to High Agreement

Table 1: **Inter-Rater Agreement via Mean r_{wg}.** Values above 0.70 indicate strong within-group agreement.

Table 1 shows that all Likert-scale metrics exceed the accepted threshold of 0.70, indicating strong agreement among evaluators. The slightly lower agreement for factualness likely reflects variation in interpreting domain-specific technical nuance rather than systemic inconsistency, which we plan to investigate further in future work.

Metric	Avg. % Agreement	α
Correct Topic Classification	85.99%	0.812
Difficulty	88.23%	0.811
Bloom's Classification	93.39%	0.810

Table 2: **Inter-Rater Reliability for Classification Metrics.** Krippendorff's α values above 0.80 indicate strong reliability.

For categorical classifications, all Krippendorff's α values exceeded 0.81, demonstrating strong inter-rater reliability across metrics. Percent agreement was similarly high, indicating consistent interpretation of pedagogical intent despite minor formatting artifacts in raw LLM outputs.

Beyond the evaluated diagnostic subset, PathBuilder integrates 1,018 professionally reviewed LLM-generated questions and 16,740 automatically classified existing questions (curated from books and other reference materials) within its structured taxonomy. In total, the deployed system contains 17,758 assessment items (as of this writing) aligned to the topic hierarchy and cognitive level. These results demonstrate that PathBuilder's assessment pipeline combines expert validation with scalable taxonomy integration, supporting both rigorous evaluation and large-scale deployment.

Having established the reliability of the assessment backbone—designed to accurately determine students' capabilities and therefore requiring rigorous validation—we next examine whether learners demonstrate measurable gains when engaging with the system.

4.2 Learning Gains: Pre and Post Assessment

At evaluation time, PathBuilder has 179 registered users. For evaluation, we considered only students who consistently used the platform for learning activities. Consistently here means that students were able to complete at least one microtopic with pre and post-test scores. Some of the students started the pre-test but did not take the post-test. Of the 179 active users, 75 students met the inclusion criteria of completing both pre- and post-assessments. We evaluated PathBuilder using matched pre–post assessments from these 75 unique students, yielding 2,192 matched test pairs across 169 unique curriculum-aligned microtopics. Because PathBuilder is personalized, students may complete different numbers of microtopics; therefore, the number of matched assessments varies per learner. All assessments were drawn from the expert-validated question bank described in Section 4.1.

Metric	Pre-Test	Post-Test
Mean Score (%)	53.24	91.14
Proficiency $\geq$75%	42.2%	97.9%
Mean Absolute Gain	37.90 pp	
Median Gain	30.00 pp	
Hake's g	0.760 (High)	
Cohen's d	0.9845 (Large)	

Table 3: **Pre and Post Test Analysis.** Pre–post learning outcomes for 75 matched students (2,192 matched pairs).

Table 3 summarizes aggregate performance statistics. Learning gains were measured using matched pre–post diagnostic assessments drawn from the expert-validated question bank. Each student completed a structured pre-test prior to instructional exposure and a post-test aligned to the same topic taxonomy and Bloom level.

We report both absolute and normalized gains. The mean absolute gain (37.90 percentage points) reflects raw performance improvement, while Hake's normalized gain ($g = 0.760$) measures improvement relative to the maximum possible gain. Under established educational benchmarks, $g > 0.70$ is considered high, indicating substantial conceptual advancement rather than marginal score inflation (Hake, 1998).

Effect size was computed using Cohen's d for paired samples ($d = 0.9845$), indicating a large practical effect beyond statistical significance. Unlike p-values, which reflect sample size sensitivity, effect size quantifies the magnitude of improvement in standardized units. The large d value suggests that performance gains are not only statistically reliable but educationally meaningful.

Proficiency was defined as achieving at least 75% accuracy. The increase from 42.2% to 97.9% indicates that improvement was not limited to high-performing students but shifted the majority of learners above the mastery threshold. This suggests that the system supports broad competency development rather than isolated performance gains.

Normality testing (Shapiro–Wilk, $p < 10^{-35}$) indicated non-normal score differences. Both parametric (paired t-test, $t = 46.09$, $p < 0.001$) and non-parametric (Wilcoxon signed-rank, $p <$

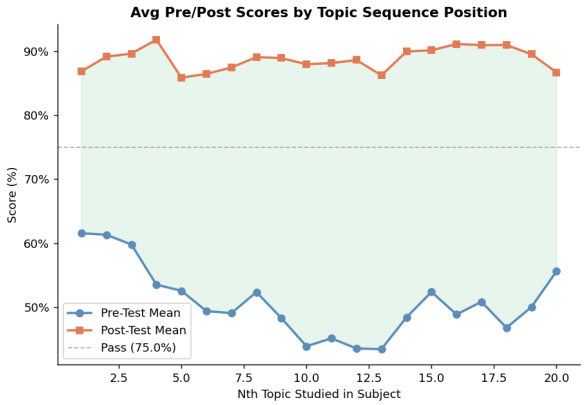


Figure 1: **Average pre- and post-test scores by topic sequence position.** Post-test performance remains consistently high across later topics, indicating sustained learning gains without degradation. The dashed line marks the 75% proficiency threshold.

10^{-200}) tests confirmed the statistically significant improvement.

Together, these metrics suggest that the integrated pipeline—comprising diagnostic profiling, retrieval-grounded generation, and quality-controlled instructional synthesis—is associated with large, meaningful improvements under real deployment conditions.

To further investigate whether gains persist across curriculum progression, we analyze cohort-average performance by topic sequence position. Figure 1 plots the mean pre- and post-test scores for the nth topic studied by each student. Post-test performance remains consistently high (85–93%) across topic sequence positions, with stable improvement magnitude. This suggests sustained effectiveness across sequential learning stages.

This analysis suggests that PathBuilder's personalization mechanism maintains effectiveness across sequential learning stages rather than exhibiting diminishing returns.

4.3 LLM-Based Quality Control

We evaluated the behavior of the LLM-as-a-Judge using a controlled set of 31 generated instructional outputs spanning multiple topics and cognitive levels. Each output was evaluated using the rubric with threshold $\tau = 9$ and up to two regeneration attempts. All 31 generations achieved a score ≥ 9 on the first evaluation pass, yielding a 100% pass rate without requiring retries. The mean correctness score was 10/10, while the mean factuality score was 9.52/10, resulting in a mean overall score of 9.58/10. These results indicate that

Metric	Value
Total Generations	31
Pass Rate (≥ 9)	100%
Retries Required	0
Mean Correctness	10.00 / 10
Mean Factuality	9.52 ± 0.51
Mean Overall Score	9.58 / 10
Perfect Score (10/10)	51.6%
Minor Factual Gaps (9/10)	48.4%

Table 4: **Rubric-based quality control results** using threshold $\tau = 9$ (chosen to enforce $\geq$90% rubric compliance) with up to two regeneration attempts.

PathBuilder's retrieval-grounded generation and structured prompting pipeline produced outputs that met predefined evaluator thresholds in controlled testing. In deployment settings (2,192 generations), all outputs similarly satisfied the threshold on the first pass under the predefined rubric threshold, indicating stable real-world performance of the quality control mechanism. Taken together, these results show that PathBuilder combines expert-validated assessment, retrieval-grounded personalization, and LLM-based quality control into a stable and scalable learning pipeline with measurable real-world impact.

5 Conclusion

We presented PathBuilder, a quality-controlled LLM system for personalized learning that integrates expert-validated assessment, retrieval-grounded instructional generation, and automated judge-based validation within a unified deployment pipeline. Unlike systems that rely solely on generative capability, PathBuilder embeds structured curriculum modeling and quality control directly into the personalization process.

Real-world deployment with 179 registered users (75 matched learners) demonstrated substantial learning gains, including a high normalized gain (g = 0.760) and large effect size (d = 0.98). The integration of over 1,000 expert-validated LLM-generated questions into a 17,758-item assessment bank further shows that controlled generative expansion can be reliably incorporated into operational systems.

These findings suggest that closed-loop, quality-controlled LLM architectures can enable scalable personalization without sacrificing reliability. Future work will evaluate the system with controlled comparison groups and longitudinal tracking.

6 Limitations

Several limitations should be acknowledged.

First, the evaluation relies on a pre–post design without a randomized control group. While statistically significant gains and large effect sizes were observed, causal attribution to specific components (e.g., retrieval grounding or judge-based validation) cannot be fully isolated. Future work will include controlled comparisons and ablation studies to disentangle the contribution of individual modules.

Second, the deployment was conducted within a specific curriculum domain and institutional context. Although the system architecture is domain-agnostic, generalization to other subject areas or learner populations requires further validation.

Third, the LLM-as-a-Judge mechanism was evaluated using rubric-based scoring rather than independent human evaluation of instructional outputs. While the rubric enforces structured quality thresholds, future work will incorporate external expert audits of generated instructional content to further assess reliability.

Finally, learner modeling is currently based on structured diagnostic profiling without longitudinal cognitive modeling (e.g., Bayesian knowledge tracing across sessions). Incorporating temporal student modeling may further enhance personalization accuracy over extended learning trajectories.

7 Ethical Considerations

PathBuilder is designed for educational use and incorporates multiple safeguards to mitigate risks associated with LLM-generated content. All instructional materials are grounded in curated knowledge sources through retrieval-augmented generation and pass an automated rubric-based validation stage before delivery to learners. These safeguards aim to reduce the risk of hallucinated or factually incorrect content.

Assessment expansion includes expert review of generated questions prior to deployment, ensuring alignment with curriculum standards and reducing the risk of misleading or low-quality evaluation items.

Regarding learner data, the system operates within institutional deployment guidelines. Only performance-related data (e.g., diagnostic scores and progression metrics) are stored for personalization purposes. No sensitive personal data are used for model training. Future work will further formalize data governance and anonymization protocols for larger-scale deployments.

We also acknowledge the broader ethical concern that over-reliance on automated systems may reduce instructor oversight. PathBuilder is intended as a supplemental instructional tool rather than a replacement for human educators. Human supervision remains essential in high-stakes educational contexts.

8 Demo and Availability

A live demonstration of PathBuilder is available at `https://demo.pathbuilderedu.com`. The system provides role-based access for students, teachers, and administrators, allowing reviewers to explore the full end-to-end pipeline described in this paper.

The system runs entirely through a web interface and requires no local installation. The landing page provides role-based access via a "Log in as" selection (student, teacher, or administrator), enabling reviewers to explore the corresponding system functionalities.

Acknowledgments

This work was supported by Makarius Smart Learning and the Department of Science and Technology – Science Education Institute (DOST-SEI) under the ASTHRDP Graduate Scholarship Program, and the Asian Institute of Management. Special thanks to Adamson University for its institutional support, and to all human evaluators and professionals who contributed their time and insights to validate the LLM-generated questions and enrich this research.

References

Chris Alberti, Daniel Andor, Emily Pitler, Jacob Devlin, and Michael Collins. 2019. Synthetic QA corpora generation with roundtrip consistency. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6168–6173, Florence, Italy. Association for Computational Linguistics.

Nischal Ashok Kumar, Nigel Fernandez, Zichao Wang, and Andrew Lan. 2023. Improving reading comprehension question generation with data augmentation and overgenerate-and-rank. In *Proceedings of the 18th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2023)*, pages 247–259, Toronto, Canada. Association for Computational Linguistics.

Cheng Cheng, GuanHao Zhao, Zhenya Huang, Yan Zhuang, Zhaoyuan Pan, Qi Liu, Xin Li, and Enhong Chen. 2024. Towards explainable computerized adaptive testing with large language model. In

Findings of the Association for Computational Linguistics: EMNLP 2024, pages 2655–2672, Miami, Florida, USA. Association for Computational Linguistics.

Albert T. Corbett and John R. Anderson. 1994. Knowledge tracing: Modeling the acquisition of procedural knowledge. *User Modeling and User-Adapted Interaction*, 4(4):253–278.

Ehsan Doostmohammadi, Oskar Holmström, and Marco Kuhlmann. 2024. How reliable are automatic evaluation methods for instruction-tuned LLMs? In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 6321–6336, Miami, Florida, USA. Association for Computational Linguistics.

Xinya Du, Junru Shao, and Claire Cardie. 2017. Learning to ask: Neural question generation for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1342–1352, Vancouver, Canada. Association for Computational Linguistics.

Liam Dugan, Eleni Miltsakaki, Shriyash Upadhyay, Etan Ginsberg, Hannah Gonzalez, DaHyeon Choi, Chuning Yuan, and Chris Callison-Burch. 2022. A feasibility study of answer-agnostic question generation for education. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 1919–1926, Dublin, Ireland. Association for Computational Linguistics.

Sabina Elkins, Ekaterina Kochmar, Jackie C.K. Cheung, and Iulian Serban. 2024. How teachers can use large language models and bloom's taxonomy to create educational quizzes. In *Proceedings of the Thirty-Eighth AAAI Conference on Artificial Intelligence and Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence and Fourteenth Symposium on Educational Advances in Artificial Intelligence*, AAAI'24/IAAI'24/EAAI'24. AAAI Press.

Richard Hake. 1998. Interactive-engagement versus traditional methods: A six-thousand-student survey of mechanics test data for introductory physics courses. *American Journal of Physics - AMER J PHYS*, 66.

Seonjeong Hwang, Yunsu Kim, and Gary Lee. 2024. Cross-lingual transfer for automatic question generation by learning interrogative structures in target languages. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 3194–3208, Miami, Florida, USA. Association for Computational Linguistics.

Enkelejda Kasneci, Kathrin Sessler, Stefan Küchemann, Maria Bannert, Daryna Dementieva, Frank Fischer, Urs Gasser, Georg Groh, Stephan Günnemann, Eyke Hüllermeier, Stephan Krusche, Gitta Kutyniok, Tilman Michaeli, Claudia Nerdel, Jürgen Pfeffer, Oleksandra Poquet, Michael Sailer, Albrecht Schmidt, Tina Seidel, Matthias Stadler, Jochen Weller, Jochen Kuhn, and Gjergji Kasneci. 2023. Chatgpt for good? on opportunities and challenges of large language models for education. *Learning and Individual Differences*, 103:102274.

Kenneth R. Koedinger, Emma Brunskill, Ryan S.J.d. Baker, Elizabeth A. McLaughlin, and John Stamper. 2013. New potentials for data-driven intelligent tutoring system development and optimization. *AI Magazine*, 34(3):27–41.

Philippe Laban, Chien-Sheng Wu, Lidiya Murakhovs'ka, Wenhao Liu, and Caiming Xiong. 2022. Quiz design task: Helping teachers create quizzes with automated question generation. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 102–111, Seattle, United States. Association for Computational Linguistics.

Wencke Liermann, Jin-Xia Huang, Yohan Lee, and Kong Joo Lee. 2024. More insightful feedback for tutoring: Enhancing generation mechanisms and automatic evaluation. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 10838–10851, Miami, Florida, USA. Association for Computational Linguistics.

Xinyi Lu and Xu Wang. 2024. Generative students: Using llm-simulated student profiles to support question item evaluation. In *Proceedings of the Eleventh ACM Conference on Learning @ Scale*, L@S '24, page 16–27, New York, NY, USA. Association for Computing Machinery.

Jakub Macina, Nico Daheim, Sankalan Chowdhury, Tanmay Sinha, Manu Kapur, Iryna Gurevych, and Mrinmaya Sachan. 2023. MathDial: A dialogue tutoring dataset with rich pedagogical properties grounded in math reasoning problems. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 5602–5621, Singapore. Association for Computational Linguistics.

Chris Piech, Jonathan Bassen, Jonathan Huang, Surya Ganguli, Mehran Sahami, Leonidas J Guibas, and Jascha Sohl-Dickstein. 2015. Deep knowledge tracing. In *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc.

Jennifer M. Reddig, Arav Arora, and Christopher J. MacLellan. 2025. Generating in-context, personalized feedback for intelligent tutors with large language models. *International Journal of Artificial Intelligence in Education*, 35(6):3459–3500.

Nicy Scaria, Suma Dharani Chenna, and Deepak Subramani. 2024. How good are Modern LLMs in generating relevant and high-quality questions at different bloom's skill levels for Indian high school social science curriculum? In *Proceedings of the 19th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2024)*, pages 1–10, Mexico City, Mexico. Association for Computational Linguistics.

Alexander Scarlatos, Ryan S. Baker, and Andrew Lan. 2025. Exploring knowledge tracing in tutor-student dialogues using llms. In *Proceedings of the 15th International Learning Analytics and Knowledge Conference*, LAK '25, page 249–259, New York, NY, USA. Association for Computing Machinery.

Robin Schmucker, Meng Xia, Amos Azaria, and Tom Mitchell. 2024. Ruffle&riley: Insights from designing and evaluating a large language model-based conversational tutoring system. In *Artificial Intelligence in Education*, pages 75–90, Cham. Springer Nature Switzerland.

Shuanghong Shen, Qi Liu, Zhenya Huang, Yonghe Zheng, Minghao Yin, Minjuan Wang, and Enhong Chen. 2024. A survey of knowledge tracing: Models, variants, and applications. *IEEE Trans. Learn. Technol.*, 17:1898–1919.

Yuhong Shi, Kun Yu, Yifei Dong, and Fang Chen. 2026. Large language models in education: a systematic review of empirical applications, benefits, and challenges. *Computers and Education: Artificial Intelligence*, 10:100529.

Masaki Uto, Yuto Tomikawa, and Ayaka Suzuki. 2023. Difficulty-controllable neural question generation for reading comprehension using item response theory. In *Proceedings of the 18th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2023)*, pages 119–129, Toronto, Canada. Association for Computational Linguistics.

Xingdi Yuan, Tong Wang, Caglar Gulcehre, Alessandro Sordoni, Philip Bachman, Saizheng Zhang, Sandeep Subramanian, and Adam Trischler. 2017. Machine comprehension by text-to-text neural question generation. In *Proceedings of the 2nd Workshop on Representation Learning for NLP*, pages 15–25, Vancouver, Canada. Association for Computational Linguistics.

Hanqi Zhou, Robert Bamler, Charley M Wu, and Álvaro Tejero-Cantero. 2024. Predictive, scalable and interpretable knowledge tracing on structured domains. In *The Twelfth International Conference on Learning Representations*.

A Detailed Statistical Analysis of Learning Gains

This appendix provides detailed statistical results supporting the pre–post learning gains reported in Section 4.2. The matched evaluation includes $N = 75$ learners and 2,192 matched pre–post assessment pairs extracted from 7,809 total platform records.

A.1 Aggregate Results

Metric	Value
Total Platform Records	7,809
Unique Students (Matched)	75
Matched Pre–Post Pairs	2,192
Mean Pre-Test Score	53.24%
Mean Post-Test Score	91.14%
Mean Absolute Gain	37.90 percentage points
Median Absolute Gain	30.00 percentage points
Mean Hake's g	0.760 (High Gain)
Pre-Test Proficiency ($\geq 75\%$)	42.2%
Post-Test Proficiency ($\geq 75\%$)	97.9%
Students Who Improved	94.7% (71/75)

Table 5: Aggregate learning outcomes across matched learners.

A.2 Student-Level Summary Statistics

The following table reports learner-level descriptive statistics computed across matched microtopics. Hake's g was defined for 73 learners (undefined in cases of ceiling effects).

Statistic	Pre Mean	Post Mean	Abs Gain	Hake's g
Count	75.00	75.00	75.00	73.00
Mean	55.46	86.59	31.13	0.63
Std	24.99	7.59	24.12	0.30
Min	0.43	60.00	-20.00	-1.00
25%	40.56	82.40	13.38	0.57
Median	59.00	86.67	28.33	0.67
75%	72.25	90.84	42.69	0.82
Max	100.00	100.00	93.24	1.00

Table 6: Student-level summary statistics across matched microtopics.

B System Transparency and Example Outputs

This appendix provides visual evidence of the deployed PathBuilder pipeline, and an example rendered personalized lesson.

B.1 End-to-End System Architecture

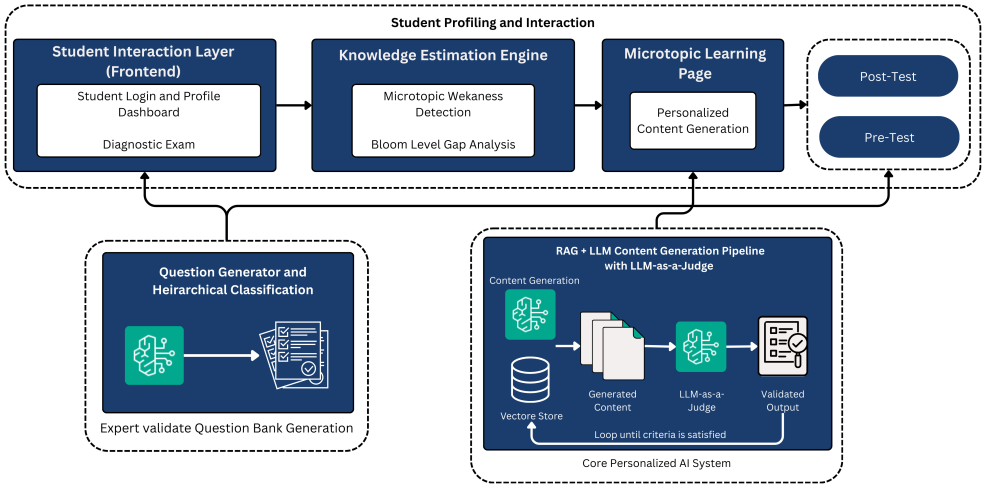
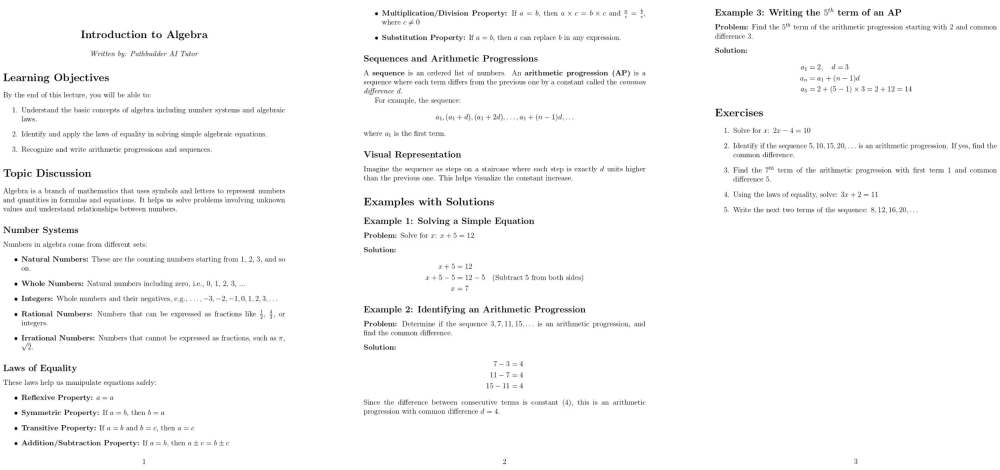


Figure 2: PathBuilder end-to-end system architecture. The pipeline integrates diagnostic profiling, microtopic detection, retrieval-augmented generation, LLM-based quality control, and post-test validation within a closed instructional loop.

B.2 Example Rendered Personalized Instruction



Introduction to Algebra

Written by: Pathbuilder AI Tutor

Learning Objectives

By the end of this lecture, you will be able to:

1. Understand the basic concepts of algebra including number systems and algebraic laws.
2. Identify and apply the laws of equality in solving simple algebraic equations.
3. Recognize and write arithmetic progressions and sequences.

Topic Discussion

Algebra is a branch of mathematics that uses symbols and letters to represent numbers and quantities in formulas and equations. It helps us solve problems involving unknown values and understand relationships between numbers.

Number Systems

Numbers in algebra come from different sets:

- **Natural Numbers:** These are the counting numbers starting from 1, 2, 3, and so on.
- **Whole Numbers:** Natural numbers including zero, i.e., 0, 1, 2, 3, ...
- **Integers:** Whole numbers and their negatives, e.g., ..., −3, −2, −1, 0, 1, 2, 3, ...
- **Rational Numbers:** Numbers that can be expressed as fractions like $\frac{1}{2}$, $\frac{3}{4}$, or integers.
- **Irrational Numbers:** Numbers that cannot be expressed as fractions, such as π, $\sqrt{2}$.

Laws of Equality

These laws help us manipulate equations safely:

- **Reflexive Property:** $a = a$
- **Symmetric Property:** If $a = b$, then $b = a$
- **Transitive Property:** If $a = b$ and $b = c$, then $a = c$
- **Addition/Subtraction Property:** If $a = b$, then $a \pm c = b \pm c$

1

- **Multiplication/Division Property:** If $a = b$, then $a \times c = b \times c$ and $\frac{a}{c} = \frac{b}{c}$, where $c \neq 0$
- **Substitution Property:** If $a = b$, then a can replace b in any expression.

Sequences and Arithmetic Progressions

A **sequence** is an ordered list of numbers. An **arithmetic progression (AP)** is a sequence where each term differs from the previous one by a constant called the *common difference* d.

For example, the sequence:

$$a_1, (a_1 + d), (a_1 + 2d), \ldots, a_1 + (n-1)d, \ldots$$

where a_1 is the first term.

Visual Representation

Imagine the sequence as steps on a staircase where each step is exactly d units higher than the previous one. This helps visualize the constant increase.

Examples with Solutions

Example 1: Solving a Simple Equation

Problem: Solve for x: $x + 5 = 12$

Solution:

$$\begin{aligned} x + 5 &= 12 \\ x + 5 - 5 &= 12 - 5 \quad \text{(Subtract 5 from both sides)} \\ x &= 7 \end{aligned}$$

Example 2: Identifying an Arithmetic Progression

Problem: Determine if the sequence $3, 7, 11, 15, \ldots$ is an arithmetic progression, and find the common difference.

Solution:

$$\begin{aligned} 7 - 3 &= 4 \\ 11 - 7 &= 4 \\ 15 - 11 &= 4 \end{aligned}$$

Since the difference between consecutive terms is constant (4), this is an arithmetic progression with common difference $d = 4$.

2

Example 3: Writing the 5^{th} term of an AP

Problem: Find the 5^{th} term of the arithmetic progression starting with 2 and common difference 3.

Solution:

$$\begin{aligned} a_1 &= 2, \quad d = 3 \\ a_n &= a_1 + (n-1)d \\ a_5 &= 2 + (5-1) \times 3 = 2 + 12 = 14 \end{aligned}$$

Exercises

1. Solve for x: $2x - 4 = 10$
2. Identify if the sequence $5, 10, 15, 20, \ldots$ is an arithmetic progression. If yes, find the common difference.
3. Find the 7^{th} term of the arithmetic progression with first term 1 and common difference 5.
4. Using the laws of equality, solve: $3x + 2 = 11$
5. Write the next two terms of the sequence: $8, 12, 16, 20, \ldots$

3

Figure 3: Excerpt of a rendered personalized lesson generated via retrieval-augmented generation and validated by the LLM-as-a-Judge prior to student delivery.