Journal of Inclusive Methodology and Technology in Learning and Teaching
ISSN 2785-5104
Anno 6 n. 3 (2026)

EDIZIONI UNIVERSITARIE ROMANE
editoria scientifica dal 1952

Supporting Instructional Design in Higher Education with Reliable AI-Generated Courseware

Ludovico Boratto [1], Marco Cabras [1,2], Salvatore Mario Carta [1,3], and Mirko Marras [1,*]

[1] University of Cagliari; ludovico.boratto@acm.org, marco.cabras4@unica.it, salvatorem.carta@unica.it, mirko.marras@unica.it
[2] University of Macerata; m.cabras1@unimc.it
[3] VisioScientiae S.r.l.; salvatore.carta@visioscientiae.com
* Correspondence: mirko.marras@unica.it

Abstract: Large language models can substantially accelerate the production of teaching materials, yet generating an entire university course requires more than producing individual pieces of educational text. Learning outcomes, syllabus structure, assessments, slides, and lecture scripts must remain mutually consistent and reflect the intended learners, teaching context, and instructional strategy. In this study, we investigate whether making these instructional decisions explicit before content generation can improve the quality and robustness of AI-generated courseware. We evaluate a structured course-authoring pipeline that combines a typed course specification, course-level instructional planning, schema-constrained lesson generation, and deterministic courseware compilation. The pipeline is compared with prior works using the same university course specifications. Four generator models are evaluated by four shared large-language-model judges using a fixed rubric covering learning objectives, syllabus, assessments, slides, and scripts. The structured pipeline improves the overall judged quality for every generator. The largest improvements concern learning objectives, syllabus organization, and assessment quality. Improvements are also consistently positive across disciplines. These findings provide evidence that explicit instructional representations can strengthen the coherence and practical reliability of AI-supported course authoring, while reinforcing the need for instructor oversight.

Keywords: Generative Artificial Intelligence; Instructional Design; Course Authoring; Large Language Models; Multi-Agent Systems.

Copyright: © 2026 by the authors. Submitted for possible open access publication under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Designing a university course is a complex educational task. Instructors must articulate learning outcomes, sequence concepts, select teaching activities, develop assessment strategies, and create mutually consistent learning resources. These decisions are iterative and interdependent, and university instructors typically move between high-level course structure and detailed learning materials throughout the design process (Bennett et al., 2017). Constructive alignment further emphasizes that intended learning outcomes, teaching activities, and assessment should function as a coherent system rather than as independently produced components (Biggs, 1996). These requirements make course preparation both pedagogically demanding and time-consuming, particularly in academia where Could you please replace

"research, administration, service responsibilities" with "research, administration, and service responsibilities"? (Gavin & McGrath-Champ, 2024).

Recent advances in large language models (LLMs) have created opportunities to support this type of work. Educational applications now span tutoring, feedback, assessment, resource generation, lesson planning, and teacher support (Kasneci et al., 2023; Wang et al., 2024; Zawacki-Richter et al., 2024). Early studies have shown that generative artificial intelligence (AI) can produce complete draft curricula or individual teaching plans, while interactive systems demonstrate the value of embedding model-generated content within a pedagogically structured teacher workflow (Davis & Lee, 2024; Fan et al., 2024). Other work has explored the refinement of teaching plans through simulated classroom interactions or multi-agent educational environments (Hu et al., 2025; Zhang et al., 2024).

Most of these applications, however, address a bounded instructional artifact or a specific stage of teaching preparation. Producing an entire course introduces a different problem: decisions made at one stage should remain visible and influential at later stages. A learning objective should affect the syllabus and lesson sequence; the assessment strategy should remain connected to those objectives; slides and scripts should instantiate rather than silently redefine earlier instructional decisions. The *ADDIE* framework provides one way of organizing these relationships through the phases of analysis, design, development, implementation, evaluation (Branch, 2009), but an AI system can follow these labels without necessarily maintaining a stable representation of the underlying instructional intent. *Instructional Agents* represents an important step toward end-to-end course authoring by organizing multiple LLM-based educational roles around instructional-design phases and producing syllabi, slide decks, lecture scripts, and assessments (Yao et al., 2026). Its contribution shows that multi-agent systems can support substantially broader instructional workflows than isolated prompting. At the same time, end-to-end generation raises practical questions concerning structural reliability, consistency among course artifacts, and the extent to which instructional decisions remain inspectable by educators. Yao et al. also document presentation-layer failures associated with directly generated slide decks, illustrating how technical generation choices can affect the practical availability of otherwise useful instructional content.

These concerns motivate a different design principle: instructional meaning and presentation should be represented separately. Rather than asking an LLM to simultaneously decide what should be taught and how the final artifact should be rendered, instructional decisions can first be represented in explicit, validated structures. Models can then generate educational content within these structures, while deterministic software handles layout, fit, and export. Such an approach is compatible with established instructional-design perspectives because it provides a persistent representation of course-level decisions before lesson-level artifacts are produced. It may also improve inspectability: instructors can review learning objectives, lesson dependencies, assessment intentions, and individual content units without reconstructing those decisions from generated presentation code.

In this study, we investigate this approach through a comparative evaluation of a structured course-authoring pipeline. The proposed approach receives an explicit course specification containing instructional constraints and pedagogical context, builds a global representation of the course, generates typed lesson content, and compiles that content into usable courseware. The study evaluates university courses

EDIZIONI UNIVERSITARIE ROMANE
editoria scientifica dal 1952

using four generator models and four shared LLM-based judges under a paired experimental design. The principal findings are that the strongest improvements occur in learning objectives, syllabus quality, and assessments - the components most closely related to instructional planning and alignment - and that the overall advantage remains positive across the three disciplinary domains. The results therefore suggest that explicit instructional representations may contribute to greater coherence across the educational components of a course.

2. Materials and Methods

2.1. Design Study

The study uses a controlled comparative design in which two course-generation methods are applied to the same course specifications under matched generator and evaluation conditions. The unit of analysis is the generated course. The comparison focuses on the proposed structured course-authoring pipeline and *Instructional Agents* (Yao et al., 2026), hereafter referred to as the baseline. The baseline implementation was used without manual correction. To preserve comparability, the experiment retained the original course specifications and evaluation rubric while changing how the instructional material was represented and generated. The original source explicitly adopts this matched design so that the comparison concerns how the task is solved rather than a change in task definition or measurement.

2.2. Considered Approaches

2.2.1. Approach 1: The Proposed Structured Course-Authoring Workflow

The proposed approach treats course generation as an instructional-design process in which pedagogical decisions are represented explicitly before the final teaching artifacts are produced (Figure 1). The followed workflow is based on an ADDIE-inspired progression from analysis and design to development and delivery, while separating instructional content generation from presentation rendering.

Figure 1. Structured workflow for AI-supported university course authoring. The process starts from a typed course specification and proceeds through analysis, instructional design, lesson development, deterministic compilation, and delivery of a complete course package. The resulting syllabus, slides, lecture scripts, and assessments are subsequently evaluated using a shared rubric covering learning objectives, syllabus quality, assessment quality, slide quality, and script quality. The workflow separates instructional content generation from presentation, keeping pedagogical decisions explicit while leaving layout and formatting deterministic.



The input stage is based on a typed course specification containing both enforceable parameters and pedagogical information. Structural parameters include lesson count, lesson duration, maximum number of slides, and delivery mode,

whereas pedagogical fields describe learning outcomes, learner characteristics, assessment strategy, instructor preferences, institutional requirements, and previous feedback. These constraints are validated when the specification is ingested and remain explicitly available throughout the subsequent stages of the workflow.

During analysis and design, the system first represents the course goals, target learners, and relevant resource requirements. It then constructs a global course map before individual lessons are developed. This representation specifies the lesson sequence, prerequisite relationships, connections between lessons, objectives, topic coverage, and associated sources. Validation mechanisms ensure that the resulting structure remains consistent with the original course specification. Consequently, course-level relationships are established before lesson-level content is generated.

During development, each lesson is generated as structured educational content rather than directly as presentation code. The representation consists of typed content units that include paragraphs, bullet lists, tables, mathematical expressions, code, charts, explanatory notes, difficulty indicators, and common misconceptions. This intermediate representation preserves instructional meaning.

The presentation stage is handled deterministically. A compiler transforms the lesson material into slide decks, controls content fit and timing, introduces the required structural elements, and exports editable teaching artifacts. The language model is therefore primarily responsible for instructional semantics, whereas layout and formatting are handled by deterministic components. The resulting course package includes the syllabus, slides, lecture scripts, and assessments, which can subsequently be evaluated using the rubric adopted in the experimental comparison.

2.2.2. Approach 2: The Baseline Course-Authoring Workflow

The baseline is the released implementation of *Instructional Agents* (Yao et al., 2026). Like the proposed approach, it organizes course generation around ADDIE-inspired phases and produces a complete set of instructional artifacts, including a syllabus, slide decks, lecture scripts, and assessments. The main differences concern how course requirements are represented, how the course structure is planned, and how final presentation artifacts are generated.

The input stage relies predominantly on free-form course information rather than on a typed specification with explicitly enforced structural constraints. Course requirements are provided to the agents as textual configuration and are interpreted during generation rather than validated and propagated as structured parameters.

During analysis and design, instructional artifacts are progressively developed through role-based agent deliberations. Rather than constructing a single validated representation of the course before lesson development, the baseline derives its lesson structure from the generated syllabus and develops subsequent components from the artifacts produced in earlier stages. Course-level organization emerges from the generation process rather than being encoded in a global course map.

During development, the baseline generates instructional content together with its presentation representation. In particular, slide material is produced directly as LaTeX code by the language model. This means that instructional semantics and presentation formatting are generated within the same model-driven process, rather than being represented separately through typed content units.

The presentation stage remains model dependent. Layout, formatting, and slide-source generation are part of the LLM output, and the resulting artifacts must be

successfully parsed and compiled before they can form part of the final course package. For the experiment, the baseline was executed using its original generation procedure, without manual correction or post-generation editing.

2.3. Pedagogical Scope and Experimental Conditions

The evaluation was designed to examine whether the proposed approach can support course authoring across different forms of university teaching rather than within a single disciplinary setting. The experimental set therefore included 15 course specifications distributed equally across *Computer Science*, *Economics*, and *Political Science*, with five courses per area. These domains were selected because they place different demands on instructional design. The three-domain setting therefore provides a useful first test of whether the benefits of a structured authoring workflow extend beyond a particular type of course organization or subject matter.

Each course specification was generated using both the proposed workflow and the baseline under the same instructional requirements. Four generator models were considered: *GPT-4o-mini*, *GPT-5.4-mini*, *Llama-3.1-8B*, and *Qwen3.5-35B*. The choice was intended to cover different model families and deployment conditions rather than to identify a single best-performing model. In particular, the inclusion of both externally hosted models and open-weight models allows the comparison to reflect educational settings with different constraints. Smaller locally deployable models are especially relevant for institutions that may need to limit reliance on proprietary services because of cost, privacy, data-governance, or infrastructure requirements. The open-weight models were executed in quantized configurations.

The resulting experimental design crossed the 15 course specifications with the two authoring approaches and four generator models, producing 120 generation runs. Both approaches were operated in fully autonomous mode, without instructor feedback, manual correction, or post-generation editing. This choice does not represent the intended final use of AI-generated courseware, which would normally involve instructor review and adaptation. Rather, autonomous generation was used as a controlled experimental condition: by excluding human intervention from both approaches, differences in the resulting course packages can be attributed more directly to the authoring workflows themselves. This is particularly important in the present study as we are interested in studying whether explicit instructional structure improves the quality and consistency of the generated materials.

2.4. Evaluation Framework and Rationale

The evaluation was designed to capture the quality of a course package as an instructional whole. For this reason, we retained the fixed rubric used in the baseline study, which is derived from the Quality Matters Higher Education framework (Quality Matters, 2023). Using the same rubric also preserves comparability between the two approaches and avoids introducing a tailored evaluation framework.

Five dimensions were considered: *Learning Objectives* (LO), *Syllabus Quality* (SY), *Assessment Quality* (AS), *Slide Quality* (SL), and *Script Quality* (SC). These dimensions were selected because they correspond to complementary stages of course design and delivery. Learning objectives capture whether the intended educational outcomes are clearly articulated; syllabus quality reflects the organization and progression of the course; assessment quality concerns the extent to which evaluation activities are appropriately designed; and slide and script quality address

EDIZIONI UNIVERSITARIE ROMANE
editoria scientifica dal 1952

the materials through which the planned instruction is ultimately communicated. Considering the five dimensions together therefore makes it possible to examine both course-level instructional coherence and the quality of the resulting teaching artifacts. Each dimension was scored on a five-point scale.

Each generated package was independently assessed by four LLM judges, corresponding to the same model families considered in the generation experiment. The use of multiple judges was motivated by the known variability of LLM-based evaluation: individual models may differ in scoring tendencies, preferences, or sensitivity to particular output styles (Zheng et al., 2023). Relying on a shared panel reduces the extent to which the comparison depends on the evaluator's behavior.

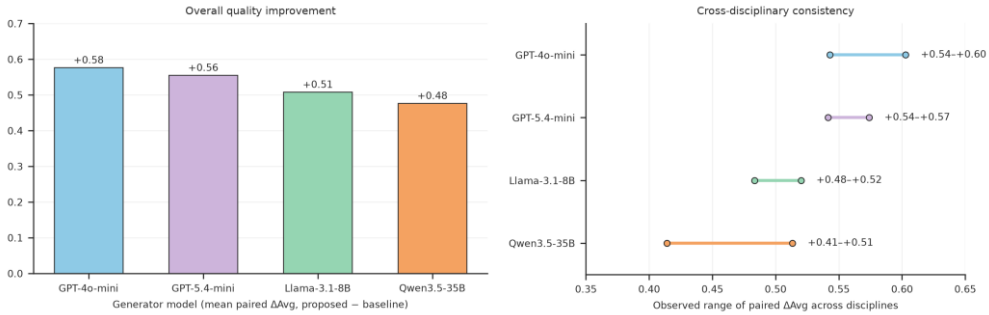
The evaluation protocol was also structured to make the comparison between the two authoring approaches as symmetric as possible. Each judge received the generated course package together with the same rubric, but was not informed which workflow had produced it and was not shown the corresponding course generated by the alternative method. The same four judges evaluated both approaches for every generator and course specification. Consequently, any model-specific scoring tendency was applied under matched conditions to both sides of the comparison. Although stylistic cues in the artifacts may prevent complete blinding, this design reduces systematic differences attributable to the evaluation procedure.

For the statistical analysis, the four judge scores were first averaged for each combination of course specification, authoring approach, generator model, and rubric dimension. The resulting judge-averaged values were then paired by course specification when comparing the two approaches. This course-level pairing is important from an educational perspective: rather than asking whether a workflow obtains higher scores on average across unrelated outputs, it asks whether the same course, generated under the same requirements and the same model, is evaluated more positively when decisions are represented with the structured workflow.

3. Experimental Results

The comparative evaluation indicates a consistent advantage for the structured course-authoring pipeline across generator models (Figure 2).

Figure 2. Quality improvements of the generated course packages. (a) Mean paired differences in overall rubric score between the proposed pipeline and the baseline across the four generator models. Positive values indicate higher judged quality for the proposed pipeline; values above the bars report the mean paired improvement. (b) Range of paired differences in overall rubric score across Computer Science, Economics, and Political Science for each generator model, indicating the consistency of the observed gains across disciplinary contexts. For Llama-3.1-8B, the overall comparison is based on complete cases, and the disciplinary range covers only Computer Science and Political Science because the baseline completed no Economics course.



(**a**) (**b**)

As shown in Figure 2a, when rubric scores are averaged across the four judges and paired by course specification, the overall quality improvement ranges from +0.48 to +0.58 points on the five-point scale. The largest mean gain is observed for *GPT-4o-mini* (+0.58), followed by *GPT-5.4-mini* (+0.56), *Llama-3.1-8B* (+0.51), and *Qwen3.5-35B* (+0.48). Considering the individual rubric dimensions, the strongest improvements concern *Learning Objectives*, *Syllabus*, and *Assessments*, whereas the gains for *Slides* and *Scripts* are generally smaller. This pattern is relevant from an instructional-design perspective since the former dimensions depend more strongly on consistency across the course as a whole: learning objectives need to inform course sequencing, and assessments need to remain aligned with the intended outcomes. The results therefore suggest that the main advantage of the structured pipeline lies in strengthening course-level coherence rather than merely improving the presentation of individual teaching artifacts.

Generation reliability further qualifies this comparison. With *Llama-3.1-8B*, the baseline completes only 3 of the 15 requested courses, whereas the proposed pipeline completes all 15 with the planned course structure. The *Llama-3.1-8B* quality difference reported in Figure 2a is therefore based only on complete cases and should be interpreted descriptively. This completion gap is particularly important in an educational context, because a course package that lacks substantial portions of its planned lessons, assessments, or teaching materials cannot be considered equivalent to a complete resource simply because its available components receive acceptable quality scores. The proposed pipeline thus shows an advantage not only in judged quality but also in the practical availability of complete course materials for subsequent instructor review and adaptation.

Figure 2b examines whether these improvements depend on the disciplinary context. For the three generators supporting complete paired comparisons across all domains, the gains remain positive and relatively stable. *GPT-4o-mini* shows improvements ranging from +0.54 to +0.60 across *Computer Science*, *Economics*, and *Political Science*; *GPT-5.4-mini* ranges from +0.54 to +0.57; and *Qwen3.5-35B* ranges from +0.41 to +0.51. These narrow intervals indicate that the overall advantage is not driven by a single disciplinary area. This consistency is notable because the three domains impose different instructional demands: *Computer Science* commonly combines conceptual and procedural learning, *Economics* links formal models with interpretation, and *Political Science* relies heavily on comparison and argumentation. The observed pattern therefore suggests that the structured pipeline supports general instructional-design operations, such as sequencing content, maintaining relationships among objectives and assessments, and preserving coherence across artifacts, rather than exploiting characteristics specific to one type of subject matter.

Llama-3.1-8B again requires a cautious interpretation. The baseline completes no *Economics* course and only three courses overall. Where paired disciplinary comparisons are available, the overall-score gain is +0.48 in *Computer Science* and +0.52 in *Political Science*. Given the limited number of complete baseline runs, these differences should not be interpreted as evidence of a discipline-specific effect. The more robust finding is that the proposed pipeline produces complete course packages in all three disciplinary areas, whereas the baseline fails to do so.

Overall, the structured pipeline improves the judged quality of the generated courseware in dimensions related to instructional planning and alignment, and this advantage remains consistently positive across the disciplinary contexts examined.

4. Discussion

The present study examined whether making instructional decisions explicit before content generation can improve the quality and robustness of AI-generated university courseware. Taken together, the results suggest that the main value of the proposed approach lies not in generating individual artifacts in isolation, but in preserving instructional coherence across the course-design process. Three main implications emerge from the findings, concerning instructional alignment, cross-disciplinary robustness, and the role of human oversight in course authoring.

The first takeaway is that explicit instructional structure appears to benefit precisely those course components that depend most strongly on alignment across the educational design. The largest improvements occur in learning objectives, syllabus organization, and assessments rather than in scripts. This result is consistent with constructive alignment, where Could you please replace "quality depends on maintaining relationships among intended outcomes, learning activities, and assessment than optimizing each artifact independently" with "quality depends more on maintaining relationships among intended outcomes, learning activities, and assessment than on optimizing each artifact independently"? (Biggs, 1996). It also reflects empirical descriptions of university course design as an iterative process that begins with a broad structure and is progressively elaborated (Bennett et al., 2017). In this sense, the contribution of the proposed workflow is not primarily that an LLM produces better prose; it is that instructional decisions are represented before and across artifact generation. This interpretation complements prior AI-assisted lesson-planning research, which similarly shows that introducing pedagogical structure around generation can improve the usefulness of produced material (Fan et al., 2024; Hu et al., 2025).

A second takeaway is that the observed advantage is not restricted to one disciplinary context. The gains are consistently positive across Computer Science, Economics, and Political Science for the three generators allowing complete comparisons. The study therefore provides initial evidence that the relevant mechanism operates at the level of general instructional-design processes—organizing learning goals, sequencing material, and preserving alignment—rather than relying primarily on discipline-specific generation patterns. The Llama-3.1-8B result adds a practical dimension: a structured pipeline can make complete course generation possible with a smaller open-weight model where a less constrained workflow frequently fails. This does not by itself demonstrate greater educational equity, accessibility, or learning effectiveness. It does, however, matter for institutions whose technological choices are constrained by local deployment, privacy, infrastructure, or access to proprietary services, concerns increasingly recognized in the broader literature on responsible AI adoption in higher education (Kasneci et al., 2023; Zawacki-Richter et al., 2024).

A third takeaway concerns how the generated material should be positioned in teaching practice. The present results measure rubric-based proxy quality, not student learning and not instructor acceptance. LLM judges permit systematic comparison at a scale that would be expensive to reproduce with expert panels, but

they can exhibit self-preference and disagree on absolute rankings (Zheng et al., 2023). Accordingly, the results support the structured pipeline as a stronger authoring substrate, not as an autonomous replacement for instructional expertise. This distinction is consistent with mixed-initiative educational systems such as LessonPlanner and with the co-pilot modes proposed by Instructional Agents, where generated materials remain objects for educator inspection and revision (Fan et al., 2024; Yao et al., 2026). The explicit course map and typed content units may be particularly useful in such a type of workflow because they expose intermediate instructional decisions to review rather than hiding them inside final artifacts.

5. Conclusions

In this study, we examined whether making instructional decisions explicit before content generation can support the creation of more coherent AI-generated university courseware. Across the examined conditions, the structured workflow received higher overall evaluations than the baseline, with the clearest improvements in learning objectives, syllabus organization, and assessment design. These are precisely the components through which instructors articulate what students are expected to learn, how learning activities are sequenced, and how achievement is assessed. The findings therefore suggest that the educational value of AI-supported course authoring may depend less on the ability to generate individual teaching materials and more on the capacity to preserve alignment across the course as a whole. The positive results across disciplinary areas further indicate that this principle may be relevant to different forms of university teaching. The greater completion reliability observed with the smaller Llama-3.1-8B model also has practical implications for institutions that may need to rely on locally deployable models. Overall, the results support a view of generative AI as a tool for assisting structured instructional design, while leaving educators responsible for interpreting, adapting, and contextualizing the resulting materials.

The limitations of the study also identify the directions for further educational evaluation. First, course quality was assessed through LLM-based judges; future studies should include instructors and instructional-design experts to examine whether the observed differences correspond to improvements that educators consider meaningful in practice. Second, the present evaluation concerns the quality of generated materials rather than their effects on teaching and learning. Classroom studies are therefore needed to investigate how instructors actually revise and integrate these materials, how students experience them, and whether they contribute to learning, engagement, or reduced preparation workload. Third, the study evaluates one generated course per experimental condition, and repeated generations would be useful for characterizing the stability of the approach. Fourth, because the complete workflows were compared as integrated systems, the contribution of individual design elements (such as the typed specification, global course map, structured lesson representation, and deterministic compilation) cannot yet be isolated. Finally, the three disciplinary areas provide an initial test of transfer across teaching contexts but do not represent the full diversity of higher education. Future work should therefore involve a broader range of disciplines and institutions and, importantly, investigate human-in-the-loop authoring workflows in which instructors review, revise, and selectively regenerate course components. Such studies would shift the evaluation from asking whether AI can autonomously

EDIZIONI UNIVERSITARIE ROMANE
editoria scientifica dal 1952

produce courseware to the more educationally relevant question of how AI can support instructors in designing, adapting, and improving learning experiences.

References

Bennett, S., Agostinho, S., & Lockyer, L. (2017). The process of designing for learning: Understanding university teachers' design work. Educational Technology Research and Development, 65(1), 125–145. DOI 10.1007/s11423-016-9469-y.

Biggs, J. (1996). Enhancing teaching through constructive alignment. Higher Education, 32, 347–364. DOI 10.1007/BF00138871.

Branch, R. M. (2009). Instructional design: The ADDIE approach. Springer. DOI 10.1007/978-0-387-09506-6.

Davis, R. O., & Lee, Y. J. (2024). Prompt: ChatGPT, create my course, please! Education Sciences, 14(1), 24. DOI 10.3390/educsci14010024.

Fan, H., Chen, G., Wang, X., & Peng, Z. (2024). LessonPlanner: Assisting novice teachers to prepare pedagogy-driven lesson plans with large language models. Proceedings of the 37th Annual ACM Symposium on User Interface Software and Technology, Article 146, 1–20. DOI 10.1145/3654777.3676390.

Gavin, M., & McGrath-Champ, S. (2024). Teacher workload and the organisation of work: A research agenda for a post-pandemic future. Labour & Industry, 34(1), 88–99. DOI 10.1080/10301763.2024.2357891.

Hu, B., Zhu, J., Pei, Y., & Gu, X. (2025). Exploring the potential of LLM to enhance teaching plans through teaching simulation. npj Science of Learning, 10, Article 7. DOI 10.1038/s41539-025-00300-x.

Kasneci, E., Seßler, K., Küchemann, S., Bannert, M., Dementieva, D., Fischer, F., Gasser, U., Groh, G., Günnemann, S., Hüllermeier, E., et al. (2023). ChatGPT for good? On opportunities and challenges of large language models for education. Learning and Individual Differences, 103, Article 102274. DOI 10.1016/j.lindif.2023.102274.

Quality Matters. (2023). QM Higher Education Rubric, Seventh Edition. Quality Matters.

Wang, S., Xu, T., Li, H., Zhang, C., Liang, J., Tang, J., Yu, P. S., & Wen, Q. (2024). Large language models for education: A survey and outlook. arXiv.

Yao, H., Xu, W., Turnau, J., Kellam, N., & Wei, H. (2026). Instructional Agents: Reducing teaching faculty workload through multi-agent instructional design. In Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers) (pp. 4087–4109). Association for Computational Linguistics. DOI 10.18653/v1/2026.eacl-long.191.

Zawacki-Richter, O., Bai, J. Y. H., Lee, K., Slagter van Tryon, P. J., & Prinsloo, P. (2024). New advances in artificial intelligence applications in higher education? International Journal of Educational Technology in Higher Education, 21, Article 32. DOI 10.1186/s41239-024-00464-3.

Zhang, Z., Zhang-Li, D., Yu, J., Gong, L., Zhou, J., Hao, Z., Jiang, J., Cao, J., Liu, H., Liu, Z., Hou, L., & Li, J. (2024). Simulating classroom education with LLM-empowered agents. arXiv.

Zheng, L., Chiang, W.-L., Sheng, Y., Zhuang, S., Wu, Z., Zhuang, Y., Lin, Z., Li, Z., Li, D., Xing, E. P., Zhang, H., Gonzalez, J. E., & Stoica, I. (2023). Judging LLM-as-a-judge with MT-Bench and Chatbot Arena. Advances in Neural Information Processing Systems, 36.