Journal of Technology-Integrated Lessons and Teaching
(2026) Volume 5, Issue 1; DOI: 10.13001/jtilt.v5i1.10247

Teaching Prompt Engineering as a Core AI Literacy Skill in Undergraduate Education

Scott Turner, University of Richmond

Overview

This learning representation introduces undergraduate students to prompt engineering as a structured, iterative practice rather than an ad hoc interaction with generative AI tools. Students design, test, and refine prompts within a domain of their choosing, documenting each iteration and evaluating outputs for accuracy, relevance, and ethical considerations. The activity emphasizes transparency, reflection, and intentional AI use, positioning prompt engineering as both a technical and metacognitive skill. By engaging students in guided experimentation and revision, the assignment supports AI literacy while reinforcing critical thinking, communication, and documentation skills applicable across academic and professional contexts.

Topics: Artificial Intelligence, Prompt Engineering, Artificial Intelligence Literacy, Generative AI, Ethical AI Use

Time: One 75-minute class session, plus one take-home assignment

Materials

- Laptop for each learner
- Generative AI tool (e.g., ChatGPT, Claude, or equivalent free-tier access)
- Prompt engineering lab instructions document

Context-at-a-Glance

Setting
The lesson occurred at a public, four-year, accredited university in the northeastern United States.

Modality
Asynchronous online, synchronous online, and in-person.

Class Structure
Upper-division undergraduate course; one 75-minute session per week; mixed technical and non-technical student backgrounds.

Learner Characteristics
Students range from AI novices to working professionals; no programming prerequisites required, making the lesson accessible across business, healthcare, education, and other non-technical disciplines.

Instructor Characteristics
Instructor with experience in artificial intelligence, data analytics, and applied machine learning; adaptable to diverse student backgrounds and professional contexts.

Development Rationale
Generative AI is increasingly present in higher education, yet students often lack guidance on effective, ethical AI interaction (Chen et al., 2020; Zawacki-Richter et al., 2019). This lesson makes AI use explicit, transparent, and assessable by framing prompting as a learned technical skill.

Design Framework
Kolb's (1984) Experiential Learning Theory combined with active learning principles (Bonwell & Eison, 1991; Felder & Brent, 2009), emphasizing hands-on experimentation, reflection, and iterative refinement.

Context and Setting

This lesson was developed and implemented within an undergraduate artificial intelligence and machine learning course at a public four-year institution's School of Professional and Continuing Studies. The course serves a diverse student population with an average age of 37, with 81% enrolled part-time, reflecting a mix of traditional students and working professionals seeking career advancement or professional development (Mew, 2020). The course was designed as part of expanding data analytics offerings, building upon successful program implementations that demonstrated effective pedagogical approaches for accommodating diverse student preparations and balancing theoretical depth with practical applicability.

The prompt engineering lesson was strategically positioned as Unit 2, early in the course sequence, to establish foundational problem-solving skills that would support independent learning throughout subsequent units. This early placement was intentional, as prompt engineering serves as a meta-skill enabling students to effectively utilize AI tools for learning support, code debugging, and concept exploration. By introducing prompt engineering before more complex topics like machine learning algorithms or neural networks, students develop self-directed learning capabilities essential for navigating rapidly evolving technologies.

The course structure accommodates students from complete beginners to experienced professionals, with no formal prerequisites beyond basic computer literacy. This accessibility was achieved through careful tool selection emphasizing free, cloud-based platforms requiring only internet access and free accounts. Technology infrastructure requirements were minimized to reliable internet access and free Google accounts for cloud-based tools, eliminating traditional barriers of software licensing, hardware specifications, and technical support complexity.

Tool selection emphasized entirely free platforms including ChatGPT or equivalent free-tier generative AI tools, GitHub for industry-standard repository management, and cloud-based environments accessible through web browsers. This approach ensured broad institutional accessibility while maintaining professional relevance, as students gained experience with tools used by software developers, data analysts, and cybersecurity professionals in real-world settings. The cloud-based infrastructure provided consistent computing resources for diverse student populations while preparing students for professional collaborative development environments.

The lesson has been delivered in three formats: asynchronous online, synchronous online, and in-person. In the asynchronous format, students complete all content, activities, and lab work independently within a weekly time window. In the synchronous format, students meet via videoconference for the 75-minute class session, with the lab completed asynchronously. In the in-person format, a computer lab with one device per student is required for the hands-on lab component. All three formats use the same instructional framework and assessment structure. Class sections typically enrolled 15-25 students, enabling personalized feedback and collaborative learning while maintaining a manageable assessment workload for the instructor.

Accessibility considerations were built into the lesson design from the outset. The no-prerequisite structure, free tool selection, and industry-agnostic problem framing ensure that students with varying academic backgrounds, technical experience levels, and professional contexts can engage meaningfully with the material. Students with limited prior exposure to AI are supported through structured scaffolding, concrete examples, and a pre-formatted document template that reduces cognitive load. The multiple delivery formats (asynchronous, synchronous, and in-person) further extend access to learners with different scheduling constraints, connectivity limitations, or learning preferences. Instructional materials also follow general web accessibility practices (e.g., WCAG guidelines), including clear text structure, multimodal resources, and alternative formats where applicable. These design choices reflect a deliberate commitment to broad access rather than narrow technical gatekeeping.

Learning Representation

Introduction

The instructor begins the lesson by posing a provocative question to students: "Why do two people get very different answers from the same AI tool?" This opening question immediately engages students' prior experiences and surfaces common frustrations with generative AI systems. Students are invited to share brief examples of their interactions with AI tools, focusing on moments of frustration, inconsistency, or surprising results. Common themes that emerge include receiving generic responses that do not address specific needs, getting different answers when asking the same question multiple times, or struggling to get the AI to understand context or nuance.

This discussion naturally leads to a key insight: AI output quality is heavily dependent on input quality. The instructor guides students to recognize that effective AI interaction requires skill and intentionality, not just luck or intuition. The instructor then introduces prompt engineering as the systematic practice of designing inputs that reliably produce useful, accurate, and appropriate outputs. This definition emphasizes that prompt engineering is not guesswork or trial-and-error, but rather an iterative technical process that can be learned, practiced, and refined.

The introduction establishes prompt engineering as a critical AI literacy skill, positioning it alongside other foundational competencies like information literacy or digital citizenship (Long & Magerko, 2020; Ouyang & Jiao, 2021). Students learn that prompt engineering enables them to move beyond passive consumption of AI outputs to active, intentional interaction with these systems. This framing shifts students' mental model from viewing AI as a magic black box to understanding it as a tool that requires skillful operation, similar to how effective database queries or search strategies require specific techniques and knowledge.

Content Presentation

Prompt Engineering Framework

Students are introduced to a structured prompt engineering framework consisting of five essential components: Context, Instructions, Output Format, Rules, and Examples. This framework provides a repeatable structure that students can apply across diverse applications and domains.

Context establishes the background information the AI needs to understand the task. This includes defining who the AI should act as (role-based prompting), who the user is, and what the broader situation or goal entails. For example, "Act as a senior data scientist reviewing code for a healthcare compliance application" provides context about the AI's role, domain expertise, and the specific application area. Context can also include subject matter background, user experience level ("I am a beginner in machine learning"), or situational context ("This will be used in a professional stakeholder presentation").

Instructions provide clear, specific directives telling the model exactly what to do. Effective instructions are unambiguous and actionable. Weak instructions like "Tell me about transformers" produce generic responses, while strong instructions like "Explain how transformer neural networks work in natural language processing, focusing on the attention mechanism and its advantages over previous architectures" guide the AI toward specific, useful outputs. Instructions should break complex tasks into clear steps.

Output Format specifies exactly how results should be structured and presented. This might include requesting bullet points, numbered lists, tables, JSON format, markdown, or specific document structures. For example, "Format your response as a table with columns for concept, explanation, and example" ensures consistent, usable output. Specifying output format is particularly important when students need structured data for further analysis or when outputs will be integrated into other systems or documents.

Rules establish constraints, dos and do nots, and boundaries for the AI's behavior. These might include tone requirements ("Use professional but accessible language suitable for a general business audience"), length constraints ("Provide a 200-word summary"), or content restrictions ("Do not include personal opinions or unverified claims"). Rules help students control AI behavior and ensure outputs meet specific requirements for their intended use.

Examples (few-shot learning, a technique in which the prompt includes a small number of demonstration input-output pairs to guide the model's response) provide sample inputs and desired outputs that demonstrate the pattern or style students want. Including 2-3 examples helps the AI understand the expected format, depth, and approach. For instance, showing examples of how to explain technical concepts to different audiences helps the AI adapt its explanation style appropriately.

The instructor demonstrates how small changes in any component can significantly alter outputs. For example, the same question ("What should I do with $10,000?") produces noticeably different responses depending on whether role-based context is included. Without context, the AI might respond: "You could invest in stocks, bonds, a savings account, or pay down debt, depending on your goals." With the context "Act as a financial advisor speaking to a first-generation college graduate with no investment experience," the response shifts to a structured, audience-aware recommendation that addresses emergency funds, risk tolerance, and long-term planning in accessible language. The role-based context changed not just the tone but the depth, structure, and relevance of the output. Similarly, modifying output format from paragraph text to a structured table transforms how information is presented, and changing rules about length or tone produces different levels of detail and formality. These demonstrations help students understand why undocumented AI use leads to inconsistent results and why systematic prompt engineering produces more reliable outcomes.

Theoretical Grounding

The prompt engineering framework is grounded in Kolb's (1984) Experiential Learning Theory, which provides a four-stage learning cycle particularly well-suited to iterative prompt development (see Figure 1; McCarthy, 2010). In the concrete experience stage, students engage directly with AI tools, writing initial prompts and observing outputs. During reflective observation, students analyze what worked and what did not, comparing different prompt variations and their resulting outputs. Abstract conceptualization occurs as students learn the framework components and understand why certain prompt structures produce better results. Finally, active experimentation happens when students apply their understanding to refine prompts and test new approaches.

This cyclical learning process mirrors professional AI development workflows where practitioners iteratively test, observe, conceptualize, and experiment with different approaches. The framework provides structure for this iterative process, preventing random trial-and-error while encouraging systematic exploration and refinement.

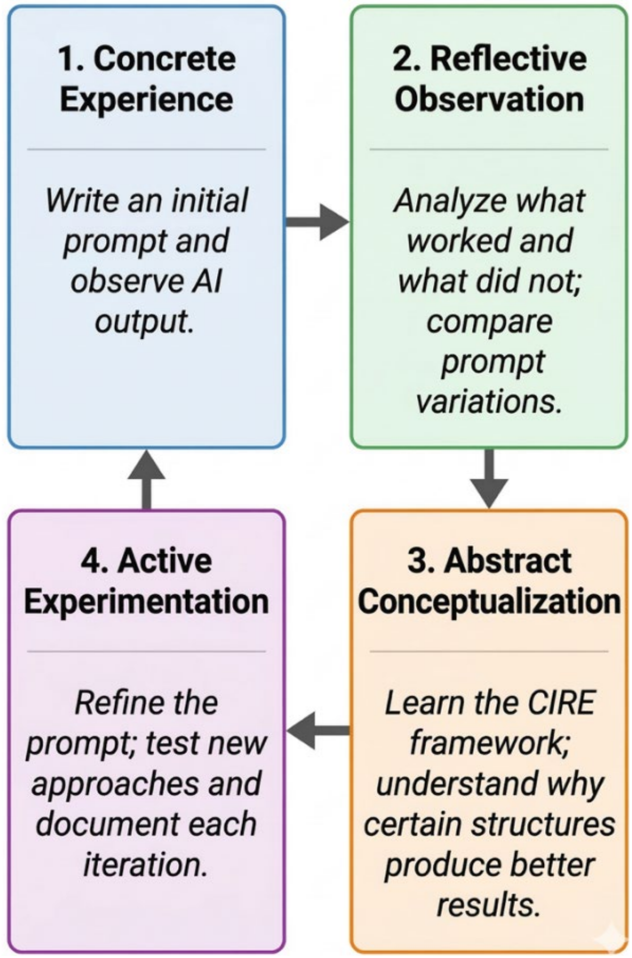

Figure 1. Kolb's (1984) Experiential Learning Cycle applied to prompt engineering

Active learning principles (Felder & Brent, 2009; Prince, 2004) are embedded throughout the lesson design. Rather than passively receiving information about prompt engineering, students actively engage in hands-on experimentation, collaborative discussion, and reflective analysis. Bonwell and Eison (1991) identify key active learning characteristics particularly relevant here: student involvement in activities beyond listening, emphasis on developing skills rather than transmitting information, and engagement in higher-order thinking processes like analysis, synthesis, and evaluation. These characteristics align closely with the competencies required for effective prompt engineering practice.

The lesson employs backward design methodology (Wiggins & McTighe, 2005), starting with desired workplace outcomes (specifically, students who can effectively use AI tools for professional problem-solving), then systematically developing assessments and learning activities to achieve these goals. This approach ensures that prompt engineering instruction directly connects to real-world applications and professional readiness.

Ethical Considerations

Ethical considerations are introduced alongside technical instruction, integrated throughout rather than treated as a separate topic. The instructor weaves ethical discussion into each phase of the lesson: during the opening discussion, students are asked to share examples of AI interactions that surprised or concerned them; during the framework presentation, the instructor explicitly names over-reliance, hallucinations, and bias as risks alongside each component; and during the lab, the instructor pauses the class at mid-point to briefly share a real example of an AI hallucination or biased output for group analysis. Students learn to recognize and address several key ethical concerns when working with generative AI.

Over-reliance on AI-generated content represents a significant risk, particularly in academic settings. Students learn to use AI as a tool for exploration, ideation, and assistance rather than as a replacement for their own thinking and learning. The portfolio assessment structure, requiring students to document their iterative refinement process, helps prevent simple copy-paste behavior and encourages critical engagement with AI outputs.

Hallucinations and false confidence pose risks when students accept AI outputs uncritically. The lesson emphasizes the importance of fact-checking, verification, and critical evaluation of all AI-generated content. Students practice identifying when AI responses contain errors, inconsistencies, or unsubstantiated claims. They learn to cross-reference AI outputs with authoritative sources and to recognize the limitations of AI knowledge, particularly regarding recent events, specialized domains, or factual accuracy.

Bias and hidden assumptions in AI outputs require careful attention. Students learn that AI systems reflect biases present in their training data and may perpetuate stereotypes, exclude perspectives, or make unwarranted assumptions. The lesson includes examples of how prompts can inadvertently reinforce biases or how AI outputs may contain problematic assumptions. Students practice identifying bias in outputs and learn strategies for crafting prompts that mitigate bias or explicitly request diverse perspectives.

Academic integrity concerns are addressed directly, with clear guidelines about appropriate AI use in coursework. Students learn that using AI for brainstorming, clarification, or learning support is acceptable, while submitting AI-generated work as their own is not. The portfolio structure, requiring documentation of the prompt development process and reflection on limitations, helps ensure academic integrity while allowing productive AI use.

Privacy and data security considerations are introduced, particularly regarding what information students should and should not share with AI tools. Students learn about data retention policies, the importance of not sharing sensitive personal or institutional information, and considerations for using AI tools in professional contexts.

Practice: Introductory Lab

The introductory lab provides hands-on practice applying the prompt engineering framework to authentic problems. The lab is structured to guide students through the iterative refinement process while allowing flexibility for individual interests and professional contexts.

Lab Setup and Preparation

The instructor begins by distributing lab instructions and ensuring all students have access to a generative AI tool. Students are reminded to create accounts if needed and to verify they can access the tool. The instructor provides a brief demonstration of basic tool navigation, showing how to start a new conversation, how to reference previous messages, and how to copy outputs for documentation. Students are introduced to a structured document template they will use to document their work.

Problem Selection

Students are instructed to select a problem relevant to their academic or professional interests. This industry-agnostic approach ensures authentic engagement while preventing academic dishonesty through diverse problem selection. Examples might include a business student wanting to analyze market trends, a healthcare student seeking to understand patient communication strategies, an education student developing lesson plan ideas, or a data analytics student exploring data interpretation approaches. The instructor circulates to help students refine overly broad or vague problem statements into specific, actionable prompts.

Initial Prompt Development

Students write their first prompt using the framework components. The instructor provides a template structure:

- Context: [Define the role, background, and situation]
- Instructions: [Specify what the AI should do]
- Output Format: [Describe desired structure]
- Rules: [List constraints and requirements]
- Examples: [Optional - provide sample patterns]

Students capture their initial prompt and the AI's response, saving both in their working document. The instructor emphasizes that this first attempt is expected to be imperfect; the goal is to establish a baseline for comparison rather than to produce a polished result.

First Iteration and Reflection

Students review their initial output and identify areas for improvement. Common issues include outputs that are too generic, missing specific details, formatted incorrectly, or not addressing the intended use case. Students revise their prompt, making specific changes to address identified issues. They capture the revised prompt and new output, documenting what changed and why. This process helps students understand the cause-and-effect relationship between prompt structure and output quality.

Second Iteration and Comparison

Students make a second round of refinements, further improving their prompts. They then create a comparative analysis documenting:

- What changed between iterations
- How outputs improved (or did not improve)
- What they learned about effective prompt structure
- Remaining limitations or areas for future improvement

Reflection and Documentation

Students write a brief reflection on the process, considering:

- What surprised them about prompt engineering
- How the framework helped structure their thinking
- Ethical concerns or limitations they noticed
- How they might apply this skill in their field

The instructor circulates throughout the lab to provide feedback, focusing on prompt clarity and reasoning rather than "right answers." In synchronous online sections, the instructor monitors student progress by requesting that students share their working document links in the chat, reviews submissions in real time, and provides feedback via direct message or brief breakout room check-ins. In asynchronous sections, the instructor reviews a mid-activity draft post before the final submission to provide timely guidance. This approach reinforces that prompt engineering is a skill requiring practice and iteration, not a set of correct formulas to memorize.

Assessment: Graded Homework

The homework assignment extends the lab work into a comprehensive prompt engineering portfolio, submitted as a written document. This assessment evaluates both technical competency and critical thinking while ensuring academic integrity through individualized problem selection and documented process.

Portfolio Requirements

Students independently expand their lab work into a professional-quality portfolio demonstrating progressive skill development. Required components include:

Problem Statement and Context (approximately 200 words)

Students provide a clear description of the problem they are addressing, why it matters in their field, and what they hope to achieve. This establishes authenticity and helps students connect prompt engineering to real-world applications.

Prompt Iterations (minimum of three)

Students document at least three distinct prompt versions, showing progressive refinement. For each iteration, students must include:

- The complete prompt text
- The AI's output (or representative excerpt for long outputs)
- Explanation of what changed from the previous version and why
- Analysis of how the output improved (or did not improve) and what limitations remain

This requirement ensures students engage in genuine iterative refinement rather than simply trying three different prompts without learning from the process.

Comparative Analysis (approximately 400 words)

Students write a detailed comparison of their prompt iterations, analyzing:

- Which prompt components (context, instructions, format, rules) had the greatest impact on output quality
- Patterns they noticed about effective prompt structure
- Trade-offs between specificity and flexibility
- How different output formats affected usability

Final Optimized Prompt

Students present their best prompt version with justification for why this version represents optimal balance between specificity, clarity, and effectiveness for their use case.

Reflection on Bias, Limitations, and Appropriate Use (approximately 300 words)

Students critically evaluate their AI interactions, considering:

- Any bias or problematic assumptions they noticed in outputs
- Limitations of AI for their specific problem domain
- When AI use would be appropriate versus inappropriate in their field
- How they would verify or validate AI outputs for professional use

Professional Documentation

The portfolio must be professionally formatted as a single document with clear section headings, labeled prompt iterations, and a brief overview explaining the structure of the submission. This requirement develops professional skills while ensuring work is accessible for assessment.

Assessment Criteria

The portfolio is evaluated using a rubric, assessing:

- Technical Competency (40%): Quality of prompt structure, effective use of framework components, demonstration of iterative refinement
- Critical Analysis (30%): Depth of comparative analysis, identification of patterns and trade-offs, recognition of limitations
- Professional Documentation (20%): Organization, clarity, and completeness of the written submission, code/documentation quality, professional presentation
- Ethical Awareness (10%): Recognition of bias, understanding of limitations, appropriate use considerations

This assessment approach evaluates process over product, emphasizing reasoning, refinement, and transparency. Because each student selects their own domain and problem, responses are inherently original, reducing opportunities for academic dishonesty while encouraging authentic exploration of personally relevant applications.

Submission and Feedback

Students submit their completed document through the learning management system. The instructor reviews portfolios, providing detailed feedback on prompt engineering technique, analytical depth, and professional presentation. Research supports the value of instructor feedback in AI-integrated learning environments as a mechanism for improving student outcomes and reinforcing critical engagement (Hooda, 2022). Common feedback themes include suggestions for more specific context, clearer instructions, better output formatting, or deeper analysis of limitations. This feedback supports continued skill development while recognizing effective practices.

Critical Reflection

This learning representation has been implemented three times in undergraduate courses, with class sizes ranging from 10 to 15 students. Implementations have included asynchronous online, synchronous online, and in-person formats; the instructional framework and assessment structure remain consistent across all three. Across implementations, students demonstrated increased intentionality in prompt construction, greater awareness of ethical considerations, and improved ability to articulate how iterative refinement influenced AI outputs.

This lesson has proven highly effective in reframing how students view generative AI, transforming their understanding from seeing AI as a shortcut or magic solution to recognizing it as a tool requiring skillful, intentional interaction. The structured framework provides students with a systematic approach they can apply across diverse contexts, building confidence and competence in AI interaction.

Students consistently report increased confidence using AI responsibly and demonstrate improved ability to troubleshoot unclear outputs, ask more precise questions, and critically evaluate AI-generated information. The hands-on lab component generates high engagement, with students actively experimenting and sharing discoveries with peers. The portfolio assessment structure, requiring documented iteration and reflection, ensures students engage deeply with the material rather than superficially completing assignments.

The industry-agnostic approach, allowing students to select problems relevant to their fields, produces authentic engagement and meaningful learning. Business students develop prompts for market analysis, healthcare students explore patient communication strategies, education students create lesson planning assistants, and data analytics students build data interpretation tools. This diversity enriches class discussions as students share insights across domains while ensuring all students develop core competencies regardless of their specific applications.

Several implementation challenges emerged during initial iterations, each addressed through course refinement. Some students initially struggled with the abstract nature of prompt engineering, finding it difficult to understand how small prompt changes could significantly impact outputs. This challenge was addressed by providing more concrete examples and demonstrations, showing side-by-side comparisons of prompt variations and their resulting outputs. The lab structure, with guided iteration steps, also helped students experience the cause-and-effect relationships directly.

Time management proved challenging for some students, particularly those less familiar with structured written documentation and iterative revision practices. The solution involved scaffolding the assignment through a pre-formatted document template and staged submission expectations, reducing cognitive load while reinforcing the iterative nature of prompt engineering. Some students also needed guidance on selecting appropriately scoped problems: those that were too broad produced overwhelming outputs, while those too narrow limited learning opportunities. The instructor now provides clearer problem selection guidelines and examples of well-scoped problems.

Academic integrity concerns required careful attention, particularly ensuring students were not simply copying AI outputs without engagement. The portfolio structure, requiring documented iteration and reflection, addresses this by making the learning process visible and assessable. The requirement for students to explain changes between iterations and analyze limitations ensures genuine engagement with the material.

A key instructional insight is that teaching prompt engineering early in the course significantly reduces downstream confusion in later AI, ML, and analytics topics. Students who master prompt engineering can independently troubleshoot AI outputs, seek clarification on complex concepts, and use AI tools effectively for learning support throughout the course. This foundational skill enables more advanced learning by providing students with self-directed problem-solving capabilities.

The integration of ethical considerations throughout technical instruction, rather than as a separate unit, proves more effective. Students naturally encounter ethical questions during hands-on practice (noticing bias in outputs, recognizing limitations, or questioning appropriate use), making ethical reasoning feel relevant and practical rather than abstract or disconnected from technical skills.

The backward design approach, starting with professional readiness outcomes, ensures the lesson remains focused on transferable skills applicable beyond the classroom. Students consistently report that prompt engineering skills transfer to professional contexts, with many using these techniques in internships, work projects, or other courses.

Several potential modifications could enhance the lesson further. A future extension could introduce AI-generated study aids (e.g., summaries or podcasts) using the same prompting framework, demonstrating additional applications while reinforcing framework principles. This extension would be optional, allowing instructors to adapt based on course goals and time constraints.

The lesson could be expanded to include more advanced techniques like chain-of-thought prompting, few-shot learning with multiple examples, or prompt chaining for complex multi-step tasks. However, the current scope appropriately focuses on foundational skills accessible to diverse student backgrounds.

For institutions with different student populations or course structures, the lesson can be adapted while maintaining core principles. The framework remains applicable whether delivered in-person, online, or in hybrid formats. The industry-agnostic problem selection approach accommodates any discipline or professional context. The assessment structure can be modified to fit different grading schemes or course requirements while preserving emphasis on process, iteration, and reflection.

The lesson's effectiveness depends on students having reliable access to generative AI tools, which may present challenges in regions with restricted access or for students with limited internet connectivity. Instructors should have backup plans and alternative tools available. The free-tier limitations of some AI tools may restrict usage volume, requiring students to be strategic about their interactions.

The rapid evolution of AI technology means that specific tool interfaces and capabilities change frequently. The lesson focuses on transferable principles rather than tool-specific instructions, but instructors must stay current with tool updates and be prepared to adapt examples and demonstrations accordingly.

Assessment workload can be substantial, given the detailed portfolio review required. Instructors should plan adequate time for providing meaningful feedback on student work. The portfolio structure, while ensuring academic integrity and deep engagement, requires more assessment time than traditional assignments.

This lesson successfully addresses the critical need for structured AI literacy education in undergraduate programs. By treating prompt engineering as a foundational, teachable skill grounded in established learning theories, the lesson prepares students for effective, ethical AI interaction across diverse contexts. The combination of structured framework, hands-on practice, and reflective assessment creates learning experiences that transfer beyond the classroom to professional applications. The lesson demonstrates that generative AI can be integrated into undergraduate instruction in ways that promote transparency, ethical use, and transferable problem-solving skills without requiring extensive technical prerequisites or additional instructional overhead.

References

Bonwell, C. C., & Eison, J. A. (1991). Active learning: Creating excitement in the classroom. ASHE-ERIC Higher Education Report No. 1. George Washington University, School of Education and Human Development. https://files.eric.ed.gov/fulltext/ED336049.pdf

Chen, L., Chen, P., & Lin, Z. (2020). Artificial intelligence in education: A review. IEEE Access, 8, 75264-75278. https://doi.org/10.1109/ACCESS.2020.2988510

Felder, R. M., & Brent, R. (2009). Active learning: An introduction. *ASQ Higher Education Brief, 2*(4), 1-5. https://engr.ncsu.edu/wp-content/uploads/drive/1XaOo9WCKcMq6-fTcQGidOT2SDGqg70l5/2009-ALpaper%28ASQ%29.pdf

Hooda, M. (2022). Artificial intelligence for assessment and feedback to enhance student success in higher education. Mathematical Problems in Engineering. https://doi.org/10.1155/2022/5215722

Kolb, D. A. (1984). Experiential learning: Experience as the source of learning and development. Prentice-Hall. https://search.worldcat.org/title/Experiential-learning-:-experience-as-the-source-of-learning-and-development/oclc/976979814

Long, D., & Magerko, B. (2020). What is AI literacy? Competencies and design considerations. Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems, 1-16. https://doi.org/10.1145/3313831.3376727

McCarthy, M. (2010). Experiential learning theory: From theory to practice. *Journal of Business & Economics Research, 8*(5), 131-139. https://doi.org/10.19030/jber.v8i5.725

Mew, L. (2020). Designing and implementing an undergraduate data analytics program for non-traditional students. *Information Systems Education Journal, 18*(3), 18-27. https://isedj.org/2020-18/n3/ISEDJv18n3p18.pdf

Ouyang, F., & Jiao, P. (2021). Artificial intelligence in education: The three paradigms. *Computers and Education: Artificial Intelligence*, 2, 100020. https://doi.org/10.1016/j.caeai.2021.100020

Prince, M. (2004). Does active learning work? A review of the research. *Journal of Engineering Education, 93*(3), 223-231. https://doi.org/10.1002/j.2168-9830.2004.tb00809.x

Wiggins, G., & McTighe, J. (2005). Understanding by design (2nd ed.). Association for Supervision and Curriculum Development. https://www.ascd.org/books/understanding-by-design-expanded-2nd-edition?variant=103055

Zawacki-Richter, O., Marín, V. I., Bond, M., & Gouverneur, F. (2019). Systematic review of research on artificial intelligence applications in higher education. International Journal of *Educational Technology in Higher Education, 16*(1), 1-27. https://doi.org/10.1186/s41239-019-0171-0

About the Author

Scott Turner is an adjunct faculty member in the School of Professional and Continuing Studies at the University of Richmond, where he teaches undergraduate courses in artificial intelligence and data analytics. He is an active AI practitioner, working professionally in applied artificial intelligence and machine learning.

Sharing & Modification Permissions