The Assessment Gap: Why Every Undergraduate Networking Course Needs to Change in the Age of Generative AI

Ranysha Ware
Swarthmore College

Vasanta Chaganti
Swarthmore College

Abstract

Generative AI (GenAI) tools now match or beat students on programming assessments, and the lab assignments at the core of undergraduate computer networking courses are unlikely to be an exception. However, the dominant institutional response remains a ban on using AI. We argue that bans are neither enforceable nor pedagogically sound in this context, and that the networking education community must treat GenAI's arrival as a call to revisit *what* we assess and *why*. We document the current landscape of lab-weight distributions and AI policies and then propose interventions that instructors can adopt incrementally, ranging in scope from shifting the weight of proctored assessments to redesigning labs around process, reasoning, and AI-scaffold use. Networking's position as a largely elective upper-level course, with no canonical downstream dependency, makes it an ideal laboratory for the curriculum-reform experiments our community urgently needs.

1 Introduction

Every undergraduate computer networking course needs to change. LLMs already match or beat students on introductory programming exams [9, 10, 17], and they handle networking exam questions too [3]. It is highly likely they can already solve the canonical socket-programming, routing, and congestion-control labs that form the backbone of most networking syllabi. Yet the field has barely begun to grapple with what that means. The most common response—a blanket AI ban—is neither enforceable nor equitable, and treating this moment as business as usual is a mistake we will regret as a community.

We structure our argument around three observations:

Bans without enforcement are aspirational, not functional: AI-detection tools carry well-documented false-positive rates [16], and OpenAI itself states that no detector has proven able to reliably distinguish AI-generated from human-written text [15]. As models improve, the claim that instructors can "tell when something is written by AI" will only grow less credible.

"We've always had cheating" understates the change: Prior work on academic integrity shows that student decisions to misuse AI are driven primarily by assessment design, workload pressure, and unclear policies—not by individual dishonesty alone [7]. LLMs do not change the *motivation* to cheat; they reduce the *effort* required to produce a plausible-looking correct solution to near zero, qualitatively changing the enforcement burden on instructors [7]. A recent survey found that 85% of college students report using AI in their coursework, and roughly 30% believe institutions should develop more AI-resistant assessments [18].

Bans exacerbate inequity: Premium GenAI access carries a significant monthly cost, widening the gap between students who can afford the most capable models and those who cannot. Inequity is not only about access: Hou et al. report preliminary evidence that effective help-seeking with GenAI is itself a skill, with disproportionate benefits for students better able to harness these tools [13]. Furthermore, unenforceable bans disproportionately burden students who want to engage honestly while watching peers benefit from non-compliance. Being accused of AI-enabled cheating carries documented psychological costs [11].

2 Background and Related Work

Despite the explosion of GenAI-in-education research since 2023, the attention has been concentrated in introductory programming (CS0/CS1/CS2) courses. An ITiCSE 2025 working group report on GenAI in *post-introductory* computing courses found the majority of literature on upper-level courses focuses on software engineering, with only 2 papers on networking [5]. As Bouvier et al. put it: "there remains a notable gap in research addressing [GenAI's] application in 'upper-level' computing courses."

Computer networking courses share key characteristics with the upper-level courses that have begun to receive attention — heavy reliance on implementation artifacts, labs that are individually assigned, and assessment concentrated in remote take-home environments — but add a distinctive profile. Networking labs (socket programming, protocol parsing, topology emulation in Mininet or Containerlab) are short, precisely specified, and I/O-testable: exactly the kind of task for which current frontier LLMs produce correct or near-correct solutions with minimal prompting [9, 10, 17]. The networking education community has rich venues for sharing practice (SIGCOMM Education Workshop, SIGCSE and ITiCSE) yet systematic evidence on how courses are actually adapting to GenAI is absent.

Networking also occupies a structurally unique position in the curriculum. As a largely upper-level elective, it typically sits downstream of the introductory sequence and data structures, but is not itself a hard prerequisite for many other required courses. This gives instructors an unusual degree of freedom to experiment with assessment redesign that would be impractical in high-enrollment gateway courses. The question is whether the community will use that freedom.

3 Syllabus Analysis

To characterize the current state of networking course policies, we used Claude Opus 4.8 to extract the top 100 institutions by aggregate SIGCOMM/NSDI/IMC publication count from `csrankings.org` [1]. While Claude was effective at gathering the institutions and even course numbers, it was less good at finding the relevant course websites: it would often generate links to outdated pages. With the course numbers in hand, however, we were able to manually search Google for the course websites, fiddling with URLs to find a current (2025 or 2026) offering. We included only institutions whose course websites are in English, and we excluded courses that did not post a public syllabus with a grade breakdown. We focused on the introductory networking course geared toward undergraduates, as graduate courses are often oriented around paper reading and research rather than completing programming assignments. One of the authors manually reviewed 15 course websites for a grade breakdown and classified any explicit statement about AI or GenAI policy on programming assignments. Three institutions were excluded because they did not publicly post grade breakdowns.

Table 1 summarizes the remaining 12 courses. The order of the rows in the table is scrambled to help maintain anonymity.

We classify each course's AI policy as a *ban*, a *partial ban*, or *no policy*. A ban is a very clear statement that AI is not allowed to be used for programming assignments at all. For example: "the use of any unauthorized assistance, including but not limited to programs such as ChatGPT and GitHub Copilot, is strictly prohibited in this course," or "unless otherwise specified, use of LLMs, chatbots, AI agents, etc. to complete homeworks and projects is prohibited in this course." Partial bans often permit something like using LLMs to figure out C syntax or debugging while forbidding submitting AI-written code as one's own. For example: "You may use LLMs (e.g., ChatGPT, GitHub Copilot) as a tool for brainstorming, clarifying concepts, or debugging" or LLM use is allowed "as a substitute for searching the internet to fix issues." We also saw one course that allowed AI on some assignments and not others. No policy means there was no policy on the public website, which could mean there is no explicit policy, it is on a syllabus posted privately elsewhere, or that a policy is discussed in class.

Key observations from the full dataset:

- Programming and lab assignments account for between 0% and 60% of the course grade, with a median near 45%. Over half place 50% or more of the grade on proctored exams.
- Some "partial ban" policies are confusingly worded, neither entirely permitting nor prohibiting AI use.
- Several policies, including partial bans, explicitly forbid copy-pasting code or submitting work generated entirely by LLMs; at least one course instead requires students to document any AI use.

4 What Should Instructors Do Next?

The rise of GenAI does not require a redesign of every assignment. We do, however, as instructors, need to ask a harder question: if a student can offload the majority of the implementation to a capable AI model — with or without permission — are our current assessments still achieving their stated learning objectives?

Audit your labs: Use a state-of-the-art model with only the materials you give students—starter code, handout, specification—and observe what it produces. Anecdotally, instructors systematically underestimate how capable these tools are, particularly if they have not used them recently. What was infeasible six months ago is routine today. If an ideological objection to using AI tools makes this uncomfortable, a safe default is to assume that all well-specified take-home labs are fully solvable.

Reassess what proctored and unproctored assessments are measuring: The ITiCSE 2025 working group on upper-level courses identifies the barrier to change not as ideological opposition to AI but as logistical: lab redesign is time-intensive, and course development rarely has protected time [5]. Even without full redesign, simple rebalancing—increasing the weight of proctored assessments relative to take-home labs—can immediately reduce the reward-to-effort ratio for AI offloading.

Consider low-cost complementary assessments: Recent work on oral assessments in the AI era documents implementations that attach comprehension checks to every assignment for under $0.50 per student per exam using AI-mediated interviewing [14], as well as empirical evidence that oral exams effectively surface AI offloading and build communication skills students genuinely need [12, 18]. For networking specifically, asking a student to explain the control flow of their TCP implementation or justify their congestion window update logic takes fewer than ten minutes and is essentially AI-proof in live form.

Consider low-cost complementary assessments: Changes can be as incremental as shifting grading weights or adding a post-submission reflection prompt—"What did you try first? What broke and why?"—as a graded artifact alongside code submission. More substantial redesigns might include standards-based grading schemes that reduce the absolute point value of any single lab, peer-teaching assessments where small groups explain a protocol to each other (evaluated on clarity of explanation, not correctness of implementation), and fully AI-collaborative labs where the goal is to document, critique, and extend AI-generated code rather than to produce code from scratch [4].

5 Rethinking Networking Education

The recommendations in the preceding section address changes an instructor can make within an existing course structure. This section takes a broader view. The arrival of capable GenAI tools creates an opening for a more fundamental question: not just how we assess networking, but what we believe a networking course is for. The remainder of this section outlines three directions for instructors who wish to move beyond assessment adjustments toward a more substantial reconception of the course itself.

Process over product: Reorienting lab assignments around the process of problem decomposition, debugging intuition, and design decisions—rather than whether submitted code passes a test suite—is perhaps the most durable change instructors can make. This means requiring students to annotate AI-generated code and shifting rubrics to weight explanation, iteration logs, and design rationale alongside functional correctness. As an example, consider asking students to critique an AI-generated BGP configuration rather than writing one from scratch. Route filtering errors, missing community attributes, and policy mismatches are rich material for analysis regardless of the original configuration. This framing—GenAI as a collaborator to interrogate rather than a shortcut to exploit—is directly analogous to how practitioners work with AI-assisted tools in industry, making it both pedagogically sound and professionally authentic.

AI as infrastructure scaffold.: A persistent friction point in networking education is environment setup: configuring VMs, debugging

Table 1: Grade composition and AI policy for the 12 sampled courses with public syllabi. Rows may not sum to 100% (homework and participation omitted). Institutions are anonymized.

ID	Type	Year	Prog. assign.	Proctored exams	GenAI Policy
Inst-1	Public	2026	45%	45%	No Policy
Inst-2	Public	2026	45%	40%	No Policy
Inst-3	Public	2025	50%	50%	No Policy
Inst-4	Public	2025	45%	50%	Ban
Inst-5	Public	2026	40%	50%	Partial Ban
Inst-6	Private	2026	40%	25%	Ban
Inst-7	Private	2025	60%	40%	Partial Ban
Inst-8	Public	2025	40%	60%	Partial Ban
Inst-9	Public	2026	30%	60%	Partial Ban
Inst-10	Private	2025	0%	95%	No Policy
Inst-11	Public	2025	60%	30%	Ban
Inst-12	Public	2026	50%	50%	Ban

Docker networking, understanding Mininet topologies, or navigating NSF testbeds such as FABRIC [8], Chameleon [6], or CloudLab. Setup overhead takes up time meant for understanding core networking concepts. GenAI can substantially lower this barrier: students can describe what they are trying to accomplish in natural language and receive working environment configurations, interpret error messages, and orient themselves in a new environment before engaging with documentation in depth. Repositioning AI as an infrastructure assistant frees cognitive load for the networking concepts that are the actual course objectives.

Humanistic assessment within networking courses.: Computer networking sits at the intersection of technical, social, political, and economic forces. Assessment modalities borrowed from the humanities, such as structured debates on traffic throttling and zero-rating, position papers on internet shutdowns and routing sovereignty, or oral defenses of architectural trade-offs in protocol design, are inherently resistant to AI: a live exchange of reasoning that cannot be delegated to a language model [12]. The networking curriculum is rich in policy-relevant questions: net neutrality [20], state-level censorship and surveillance infrastructure [19], the governance of critical internet resources, and the equity implications of last-mile connectivity all sit at the intersection of technical and societal reasoning that humanistic assessment modalities are designed to surface. Incorporating these modalities responds to a shift that AI itself has accelerated: as routine technical optimization becomes increasingly automatable, the ability to reason critically about the social, political, and economic dimensions of network design is no longer supplementary—it is central.

6 Future Work and a Call to the Community

A comprehensive, publicly shareable dataset of computer networking course syllabi—covering grade breakdowns, GenAI policies, and learning objectives across the top research institutions—would provide the field with a baseline against which to measure change over time. We plan to extend our sample toward the full top-100 list and to apply the policy taxonomy of Ali et al. [2], whose analysis of 98 computing course syllabi goes considerably deeper than our first-pass classification. Longitudinal data tracking how these grade distributions and policies change would also be valuable. Beyond our work, the field needs more controlled studies of specific assessment interventions—oral comprehension checks, process reflections, AI-collaborative labs—in networking courses.

The last comparable disruption to undergraduate education was the COVID-19 pandemic, which forced rapid experimentation with remote labs, asynchronous formats, and virtual testbeds. GenAI's impact on assessment is equally disruptive. This is a rare opportunity for the networking education community to collectively examine what an undergraduate networking course actually teaches, and whether what it teaches is what we intend. That question is worth engaging even for those who object to the use of AI tools on principle: the value of the curriculum does not depend on one's position on GenAI.

Acknowledgments

We thank the reviewers for their helpful comments.

References

[1] [n. d.]. CSRankings: Computer Science Rankings. Accessed July 2026. https://csrankings.org/

[2] Areej Ali, Aayushi Hingle Collier, Umama Dewan, Nora McDonald, and Aditya Johri. 2025. Analysis of Generative AI Policies in Computing Course Syllabi. In *Proceedings of the 56th ACM Technical Symposium on Computer Science Education V. 1* (Pittsburgh, PA, USA) *(SIGCSETS 2025)*. Association for Computing Machinery, New York, NY, USA, 18–24. doi:10.1145/3641554.3701823

[3] Mubashir Anwar and Matthew Caesar. 2025. Understanding Misunderstandings: Evaluating LLMs on Networking Questions. *SIGCOMM Comput. Commun. Rev.* 54, 4 (Feb. 2025). doi:10.1145/3717512.3717515

[4] Andres Bejarano, Ethan Dickey, and Rhianna Setsma. 2025. Implementing the AI-Lab Framework: Enhancing Introductory Programming Education for CS Majors. In *Proceedings of the 56th ACM Technical Symposium on Computer Science Education V. 2* (Pittsburgh, PA, USA) *(SIGCSETS 2025)*. Association for Computing Machinery, New York, NY, USA, 1383–1384. doi:10.1145/3641555.3705201

[5] Dennis J. Bouvier, Bruno Pereira Cipriano, Richard Glassey, Olga Petrovska, Emma Anderson, Anastasiia Birillo, Ryan Dougherty, Raymond Pettit, Nuno Pombo, Ebrahim Rahimi, Charanya Ramakrishnan, Alexander Steinmaurer, Shubbhi Taneja, Muhammad Usman, and Annapurna Vadaparty. 2025. The Rest of the Robots: Generative AI in Post-introductory Computing Education. In *Proceedings of the 2025 Working Group Reports on Innovation and Technology in Computer Science Education*. ACM, Nijmegen Netherlands, 61–107. doi:10.1145/3760545.3783970

[6] Chameleon Cloud. [n. d.]. Chameleon: A Large-Scale, Reconfigurable Experimental Environment for Cloud Research. Accessed June 2026. https://www.chameleoncloud.org

[7] Ronnie de Souza Santos, Italo Santos, Mariana Bento, Giuseppe Destefanis, Cleyton Magalhães, and Mairieli Wessel. 2026. LLM Use, Cheating, and Academic Integrity in Software Engineering Education. arXiv:2603.17060 [cs.CY] https://arxiv.org/abs/2603.17060

[8] FABRIC Testbed. [n. d.]. FABRIC: Adaptive Programmable Research Infrastructure for Computer Science and Science Applications. Accessed June 2026. https://portal.fabric-testbed.net

[9] James Finnie-Ansley, Paul Denny, Brett A. Becker, Andrew Luxton-Reilly, and James Prather. 2022. The Robots Are Coming: Exploring the Implications of OpenAI Codex on Introductory Programming. In *Proceedings of the 24th Australasian Computing Education Conference* (Virtual Event, Australia) *(ACE '22)*. Association for Computing Machinery, New York, NY, USA, 10–19. doi:10.1145/3511861.3511863

[10] James Finnie-Ansley, Paul Denny, Andrew Luxton-Reilly, Eddie Antonio Santos, James Prather, and Brett A. Becker. 2023. My AI Wants to Know if This Will Be on the Exam: Testing OpenAI's Codex on CS2 Programming Exercises. In *Proceedings of the 25th Australasian Computing Education Conference* (Melbourne, VIC, Australia) *(ACE '23)*. Association for Computing Machinery, New York, NY, USA, 97–104. doi:10.1145/3576123.3576134

[11] Tim Gorichanaz. 2023. Accused: How students respond to allegations of using ChatGPT on assessments. *Learning: Research and Practice* 9, 2 (July 2023), 183–196. doi:10.1080/23735082.2023.2254787

[12] Catherine Hartmann. 2025. Oral Exams for a Generative AI World: Managing Concerns and Logistics for Undergraduate Humanities Instruction. *College Teaching* (2025). doi:10.1080/87567555.2025.2558563

[13] Irene Hou, Sophia Mettille, Owen Man, Zhuo Li, Cynthia Zastudil, and Stephen MacNeil. 2024. The Effects of Generative AI on Computing Students' Help-Seeking Preferences. ACM, Sydney NSW Australia, 39–48. doi:10.1145/3636243.3636248

[14] Panos Ipeirotis and Konstantinos Rizakos. 2026. Scalable and Personalized Oral Assessments Using Voice AI. *arXiv preprint arXiv:2603.18221* (2026). https://arxiv.org/abs/2603.18221

[15] OpenAI. 2023. How can educators respond to students presenting AI-generated content as their own? OpenAI Help Center. Accessed July 2026. https://help.openai.com/en/articles/8313351-how-can-educators-respond-to-students-presenting-ai-generated-content-as-their-own

[16] Timothy Paustian and Betty Slinger. 2024. Students are using large language models and AI detectors can often detect their use. *Frontiers in Education* Volume 9 - 2024 (2024). doi:10.3389/feduc.2024.1374889

[17] James Prather, Paul Denny, Juho Leinonen, Brett A. Becker, Ibrahim Albluwi, Michelle Craig, Hieke Keuning, Natalie Kiesler, Tobias Kohn, Andrew Luxton-Reilly, Stephen MacNeil, Andrew Peterson, Raymond Pettit, Brent N. Reeves, and Jaromir Savelka. 2023. The Robots are Here: Navigating the Generative AI Revolution in Computing Education. In *Proceedings of the 2023 Working Group Reports on Innovation and Technology in Computer Science Education*. 108–159. arXiv:2310.00658 [cs]. doi:10.1145/3623762.3633499

[18] Joanna Slater. 2025. College Professors Use Oral Exams to Combat AI. The Washington Post. https://www.washingtonpost.com/education/2025/12/12/ai-artificial-intelligence-college-oral-exam/

[19] Steffen Wendzel, Simon Volpert, Sebastian Zillien, Julia Lenz, Philip Rünz, and Luca Caviglione. 2025. A Survey of Internet Censorship and its Measurement: Methodology, Trends, and Challenges. *arXiv preprint arXiv:2502.14945* (2025). https://arxiv.org/abs/2502.14945

[20] Nicole Wetsman. 2025. Net Neutrality Was Back, Until It Wasn't. The Verge. https://www.theverge.com/report/851629/net-neutrality-fcc-retrospective-2025