

Generative Artificial Intelligence for Education and Pedagogy

Chairs: Kavita Bala, Alex Colvin

Committee members: Morten H. Christiansen, Allison Weiner Heinemann, Sarah Kreps, Lionel Levine,
Christina Liang, David Mimno, Sasha Rush, Deirdre Snyder, Wendy E. Tarlow, Felix Thoenmes,
Rob Vanderlan, Andrea Stevenson Won, Alan Zehnder, Malte Ziewitz

Cornell University

July 18, 2023

Table of Contents

Executive summary.....	1
Section 1: Introduction.....	1
Recommendations	2
Section 2: Generative AI capabilities and limitations.....	3
Understanding the limitations of GAI technologies.....	3
Section 3: Guidelines and recommendations for educational settings.....	4
Section 3.1: Academic integrity and accessibility	4
Section 3.1.1: Prohibiting GAI tools.....	5
Section 3.1.2: Attribution, authorship, access, and accountability in the use of GAI	6
Section 3.2: Encouraging the responsible use of GAI tools.....	7
Section 3.3: Use of GAI by instructors for course content creation	9
Section 4: Recommendations for faculty and administration.....	9
Section 5: Conclusions.....	10
Appendix A: State of the art in Generative AI.....	12
Section A1.1: Language.....	12
Section A1.2: Code.....	13
Section A1.3: Images.....	13
Section A1.4: Other modalities	13
Section A1.5: Generative AI in software tools.....	14
Appendix B: Courses that develop writing as a skill.....	15
Appendix C: Creative courses for music, literature, and art	20
Appendix D: Courses in social sciences	23
Appendix E: Mathematics, physical sciences, and engineering	25
Appendix F: Courses in programming.....	29
Appendix G: Courses in law	34

Executive summary

Educators must take generative artificial intelligence (GAI) into account when considering the learning objectives for their classes, since these technologies will not only be present in the future workplace, but are already being used by students. While these tools have the opportunity to customize the learning experience for individual students and could potentially increase accessibility, they also hold risks. The most obvious risk is that GAI tools can be used to circumvent learning, but they may also hide biases, inaccuracies, and ethical problems, including violations of privacy and intellectual property. To address the risks of GAI while maximizing its benefit, we propose a flexible framework in which instructors can choose to *prohibit*, *to allow with attribution*, or *to encourage* GAI use. We discuss this framework, taking into consideration academic integrity, accessibility, and privacy concerns; provide examples of how this framework might be broadly relevant to different learning domains; and make recommendations for both faculty and administration.

Section 1: Introduction

Generative artificial intelligence (GAI) has attracted significant attention with the introduction of technologies like ChatGPT, Bard, and Dall-E, among others. This new technology has spurred major investments by Amazon, Google, Microsoft, and spawned many new startups. While there is much excitement about GAI's potential to disrupt various industries, many have voiced significant concerns about its potential for harmful use [[1](#)]. This excitement and concern has been echoed in the context of education.

In Spring 2023, the Cornell administration assembled a committee to develop guidelines and recommendations for the use of GAI for education at Cornell, with the following charges:

1. Evaluate the feasibility and benefits and limitations of using AI technologies in an educational setting and its impact on learning outcomes.
2. Assess the ethical implications of use of AI technologies in the classroom.
3. Identify best practices for integrating AI technologies into curriculum and teaching methodology. Recommend guidelines for the safe and effective use of AI technologies in an educational setting. Provide recommendations for ongoing evaluation and improvement of the use of AI technologies in an educational setting.

The committee included a broad spectrum of educators who span disciplines across the university. Over a series of meetings in Spring 2023, the committee developed the guidelines shared in this report.

Opportunity: GAI has been touted as a potential paradigm shift in education. Proposed benefits include providing a customized learning experience for all learners matching their individual needs; increasing accessibility for students with learning disabilities, anxiety, or language barriers; allowing instructors to scale constructive critiques for iterative learning and improvement in writing; and assisting in tasks in a number of domains including coding, creative composition, and more.

Concerns: Currently, GAI output can include inaccurate information, toxic output, biases embedded in the model through the training process, and infringement of copyrights on material and images. Students can use GAI to circumvent the process of learning and assessment in classes. In cases when GAI tools can serve

learning outcomes, lack of affordable access for all students could exacerbate systemic inequalities. In addition, overreliance on these tools risks atrophying students' ability and willingness to interact with instructors and peers.

Recommendations

1. **Rethink learning outcomes.** GAI requires rethinking our goals in teaching and the learning outcomes we hope to achieve. With the tremendous projected impact of GAI in many industries, students will be working in a GAI-enabled world. We strongly encourage instructors to integrate GAI into learning outcomes to focus student education on higher-level learning objectives, critical thinking, and the skills and knowledge that they will need in the future.
2. **Address safety and ethics.** We expect that GAI technology will continue to improve, and Cornell will play an important role in enabling its ethical and safe use. However, instructors must educate their students about the pitfalls of current technology. They must teach them to approach GAI critically and to validate GAI-produced information rigorously.
3. **Explicitly state policies for use of GAI.** Instructors should clearly and consistently communicate to students their expectations on the use of GAI in their assignments and classes, including when it is and is not allowed, and what uses of GAI are considered violations of academic integrity. Further, when GAI is permitted, it should be correctly attributed, and instructors should discuss the importance of student validation of the information generated. Many foundational skills will still need to be developed without the use of GAI. In such cases, instructors must directly explain to students *why* the process of achieving the specified learning outcomes for a class, without reliance on tools that create "shortcuts," is integral to a student's academic and personal growth.

We recommend instructors consider three kinds of policies either for individual assignments or generally in their courses.

1. To **prohibit** the use of GAI where it interferes with the student developing foundational understanding, skills, and knowledge needed for future courses and careers.
2. To **allow with attribution** where GAI could be a useful resource, but the instructor needs to be aware of its use by the student and the student must learn to take responsibility for accuracy and correct attribution of GAI-generated content.
3. To **encourage** and actively integrate GAI into the learning process where students can leverage GAI to focus on higher-level learning objectives, explore creative ideas, or otherwise enhance learning.

Roadmap. We structure this report as follows. In Section 2, we present a brief, high-level overview of Generative AI technologies in various domains, with links to detailed resources in Appendix A. In Section 3, we present our guidelines for the three policies described above, including discussions on academic integrity and accessibility issues, and examples and recommendations of use cases for various styles of teaching in diverse disciplines (further expanded in Appendices B-G). In Section 4, we make specific recommendations for faculty and administration, and we summarize our discussion in Section 5.

Section 2: Generative AI capabilities and limitations

The last year has seen qualitative leaps in the capability of Generative AI that took even experienced AI researchers by surprise. These tools share certain commonalities in their use and training, but also have critical differences in capabilities, use cases, and availability. Generative AI models of natural language, such as ChatGPT or Bard, are known as Large Language Models (LLMs) and are trained on massive corpora of textual data. A different method, diffusion, is used to generate images with models trained on massive collections of image data, including artwork. The capabilities of these systems will likely change drastically in the coming years. While we provide resources in Appendix A to survey existing models, we do not intend these to be comprehensive, but rather to provide a snapshot of capabilities and common models to ground the report in specific approaches. Appendix A gives more details on how these models are trained for various domains like text, code, images, and other modalities, and how the technology is currently made available to users.

The current generation of GAI is based on large-scale, deep neural networks. While this technology is not new, recent developments in model design, hardware, and the ability to aggregate large amounts of training data have made it possible to create models that are larger by orders of magnitude. The models are pretrained, which means they are fit to a large number of human-generated examples. The more data that is available, the better generative outputs become in the precision, detail, and memory of their training data, and also in their ability to generalize based on abstract patterns. Sourcing data to train better models has led to many of the ethical concerns about GAI, including (but not limited to): violations of copyright; bias, or even toxicity, deriving from the training data; and forms of plagiarism within the model outputs. However, it is critical to note that these models are not just memorizing and regurgitating these data, and outputs are rarely identical to any single observed training instance.

Current GAI provides capabilities that would not have been thought possible even at the beginning of this decade. The most popular use patterns allow users to "ask" for output using natural language, and then refine those requests based on the system's output. Systems can produce fluent, grammatical, detailed, and mostly accurate descriptions of historical events, in the style of a Wikipedia page, without including any sentence that actually occurs in an original page. When prompted with a description of a computational process and user interface, systems can generate executable program code that could have come from a Stack Overflow post or a GitHub repository, but does not actually occur in either of those services. Diffusion models can produce images of "a hedgehog driving a train in the style of Botticelli" that have never actually existed. The compelling, immediately satisfying nature of these tools led to the explosion of GAI from a niche research area to popular culture, almost overnight.

Understanding the limitations of GAI technologies.

New users often trust GAI models more than they should. While chatbots are increasingly being trained to avoid low-confidence statements, LLMs answer questions and justify their answers using plausible and confident-sounding language, regardless of the quality of the available evidence. LLMs have only one function: given a history of a conversation, they can predict the next word in the conversation. Thus, if asked to justify a sensible response, an LLM will often provide a reasonable-sounding answer based on its

training data. However, when asked to justify a nonsensical response, it will use the same techniques, resulting in a reasonable-sounding but false answer.

Users also often overestimate GAI ability by assuming they can do certain computational tasks well that current computing systems are good at; for example, mathematical calculations or finding references. But at time of writing, GAI models are not able to mathematically reason about floating point numbers, although they provide *incorrect*—but confident-sounding—answers to complex trigonometric questions. GAI models also fluently create plausible but false academic and other references. However, developers are actively working to augment GAI with more traditional tools (calculators, search engines) to address these issues, so GAI models may become more reliable over time.

Section 3: Guidelines and recommendations for educational settings

Educational settings vary significantly across Cornell: from large lecture halls to seminars, from lab, field, or studio courses to clinical and practicum-based settings. Generative AI has potential uses in almost all of these settings. For example, educators can use GAI to develop lecture outlines and materials, generate multiple versions of assignments, or compose practice problems. Students can use GAI to research topics and areas, iterate on text to improve written work, program, design and create code, art, and music, and many other uses. Depending on the use, however, GAI could end up “doing the learning” that an assignment is designed to elicit, or distorting the value of assessments, depending on student use of GAI tools.

To address these risks and take advantage of these opportunities, instructors should reassess learning outcomes for each class in light of the advent of GAI. To do so, they should consider the expectations of future courses that may build on the understanding and knowledge their course is expected to develop in the student; the expectations that future workplaces will have for workers’ responsible and ethical use of GAI tools; and the new opportunities for the purposeful use of GAI tools to aid learning.

The introduction of calculators into mathematics education provides a useful, if imperfect, analogy. Students in elementary school still learn how to do long division and multiplication. However, once students have mastered these skills, calculators are used in higher level classes to allow students to solve complex problems without being slowed down by the minutiae of arithmetic, and students are taught to use computational tools to assist in solving hard problems. Education curricula in mathematics have adapted to the increasing availability of calculators to prioritize students’ learning basic skills without calculators at first, and later, allow and/or encourage their use to support higher level learning.

Section 3.1: Academic integrity and accessibility

One major concern around the use of AI is the potential for academic integrity violations where students use these technologies to do assignments when the work itself is meant to develop skills; for example, practicing problem sets, or to assess basic skills before students proceed to higher levels of learning. In these cases, the use of GAI tools may appropriately be **prohibited**. A second area of concern is if students use these tools without properly citing them, and/or without questioning the underlying mechanisms or assumptions that produce the content. In this case, the need for students to learn to appropriately **attribute** and critique these tools is key. However, there are also cases in which the use of GAI should be

encouraged; for example, to promote the universal accessibility of assignments, or to provide tools that will enhance high-level learning and allow students to be more creative and more productive. Below, we discuss each of these options in turn, bearing in mind that clear policies and processes are critical for a constructive learning environment and trust between the instructor and student.

Section 3.1.1: Prohibiting GAI tools

Current free access to at least limited versions of GAI tools greatly accelerates the concern that students may be tempted to violate academic integrity principles by turning in AI-generated content as their own. ChatGPT's ability to almost instantly complete assignments for students makes it significantly more tempting for students facing competing priorities or a last-minute deadline crisis.

Given the widespread use of LLMs, a natural demand has arisen for tools to detect the content of LLM output. Many different tools have been developed for this task including TurnItIn, GPTZero, and OpenAI's classifier. However, in the absence of other evidence, technical methods are not currently very helpful for regulating AI usage in the classroom. The objective of LLMs is to produce text with the same statistical properties of natural language, making the detection problem adversarial as GAI improves. LLMs rarely output long snippets of text that are verbatim copies of existing content, which is the basis of traditional plagiarism detection. Therefore, attempts to identify text generated by GAI can only be done statistically. This method will likely continue to produce both false positives and false negatives, and cannot decisively provide evidence of academic integrity violations. Using these methods currently could lead to unfairly identifying academic integrity violations (for example, bias against non-native speakers), creating a lack of trust between the instructor and students, and damaging the learning environment.

Another potential remedy would be for GAI providers to explicitly limit the outputs of their systems to prevent them from answering common basic homework questions, analogous to the way question forums handle homework questions [2]. While it is technically possible for providers to implement these restrictions, or even for universities to run their own restricted services, the barriers have not proven to be foolproof. There are now many documented examples of GAI "jailbreaks" that allow adversarial parties to work around the restrictions put on GAI models. Jailbreaking works by constructing a long and complex prompt that can be fed into a model to cause it to ignore its constraints and generate in an alternative manner. Jailbreaking has been used to circumvent prohibitions on certain topics or behaviors such as advocating violence; however, it could also easily be used to obtain answers to homework assignments.

Beyond fairness to other students and to faculty, an overarching concern is that if students rely on GAI, they will not put in the practice needed to learn nor gain confidence in their ability to master needed knowledge or skills. Instructors should communicate to students why completing assignments without "shortcuts" is necessary to meet learning outcomes; why meeting specific learning outcomes is necessary to a student's academic and personal growth; and why academic integrity violations are so harmful to both the individual student and to the larger learning communities at Cornell and beyond.

Faculty may take other measures to avoid the risk of academic integrity violations by moving to assessments and assignments less suited to GAI models; for example, aligning assessments more closely to class content; or moving assessments from take-home to in-class, e.g., timed oral and written exams, or

in-class written essays. These forms of assessment could potentially disproportionately impact students with disabilities, although accommodations such as extended time and distraction-free testing zones might help to address these issues. We examine below the dynamic potential for GAI to support students with diverse disabilities; however, we also note that overreliance on these tools can put these students at an even greater disadvantage. Students with disabilities may prefer using these tools to interactions with faculty and other support systems, and become dependent on GAI to meet their needs—especially in the absence of fuller classroom access. They may also face greater vulnerability in proving that they did not violate academic integrity standards.

Section 3.1.2: Attribution, authorship, access, and accountability in the use of GAI

GAI is a very rapidly evolving technology that is in a major state of flux. While companies are swiftly working to identify and fix issues as they are discovered, it is important for instructors to be aware of the risks involved in using GAI as is currently available. Instructors must educate their students about these risks, and develop plans to mitigate the negative impact of risks in their classroom if they decide to use or allow the use of GAI in their teaching.

GAI tools pose potential privacy risks because data that is shared may be used as training data by the third-party vendor providing the service. Therefore, any information that educators are obligated to keep private, for example, under the Family Educational Rights and Privacy Act (FERPA) or the Health Insurance Portability and Accountability Act (HIPAA), should not be shared with such tools or uploaded to these third party vendors of GAI.

GAI tools also have implications for intellectual property rights. Original research or content that is owned by Cornell University, our students, or employees should not be uploaded to these tools, since they can become part of the training data used by the GAI tools. These include student assignments, data produced in projects or research groups, data that contains personally identifiable information, data from research partners (for example, companies) that may contain proprietary information, data that could be protected by copyright, etc.

If a class expects the use of GAI, it is important to make sure all students have equal access to the technology without cost barriers resulting in differential access. Licensing agreements for the use of GAI tools should be provided or negotiated by the institution, while ensuring that the tools do not limit the university's educational activities and academic freedom, respect privacy and intellectual property rights, and do not impose cost barriers or constraints.

At the time of writing, the US Department of Education has issued a new report [[3](#)] encouraging the development of policies that advance learning outcomes while protecting human decision making and judgment; that focus on data quality of AI models to ensure fairness and unbiased decisions in educational applications; that understand the impact on equity and increase the focus on advancing equity for students. We agree with these important considerations and expect Cornell researchers and educators will contribute to these improvements in GAI.

Section 3.2: Encouraging the responsible use of GAI tools

GAI will inevitably be part of the future workplace, and thus a tool that *all* students will eventually need to learn to use appropriately. Consequently, instructors now have the duty to instruct and guide students on ethical and productive uses of GAI tools that will become increasingly common in their post-Cornell careers. GAI also has the potential to provide support for students with disabilities, particularly for individuals who experience difficulties with cognitive processing and focusing, “social scripting” (i.e., neurotypical means of communication), and anxiety. We have discussed above some of the dangers of reliance on GAI in ways that are counterproductive to learning. However, it is also important for faculty to recognize the barriers that students with disabilities face, and how GAI tools can help implement and sustain fuller modes of access and inclusion for all students in the classroom.

Below, we identify settings across a range of different areas of study where use of GAI could advance teaching goals and learning objectives, and make recommendations based on the different needs of each category. Appendices B-G provide detailed examples. This list is not exhaustive, but can help identify immediate practical use cases to instructors in a wide range of disciplines. While we see a range of possible ways in which GAI can be useful in teaching, common themes include:

- Use of GAI for individualized practice, help, and tutoring
- Use of GAI to generate material for students to analyze, organize, and edit
- Use of GAI for routine, preparatory work leading to higher order thinking and analysis
- Analysis of GAI’s use and impact in a domain
- Practice of the use of GAI as a tool

We now describe the uses within various disciplines and areas of study.

Section 3.2.1: Courses which develop writing as a skill (e.g., the writing seminars)

GAI tools offer opportunities to help students develop their writing skills through assisting in planning, outlining, editing, as well as by providing individualized feedback. However, use of GAI for generating text and editing in the writing process raises major concerns about attribution of work, academic integrity, plagiarism, and failure to develop foundational and advanced writing skills and judgment. Creativity and originality in writing are key learning outcomes that could be threatened by dependency on GAI in the writing process. Guided use of GAI is encouraged as the best approach to further, rather than undermine, learning outcomes. Examples could include the use of GAI to: generate an outline for a written report that students practice revising; summarize themes from a meeting transcript that students organize and prioritize; brainstorm ideas that students then evaluate; or generate lists of sources that students validate and assess. [Appendix B](#) outlines detailed examples and scenarios.

Section 3.2.2: Creative courses for music, literature, and art

Creative fields such as art and music have long engaged in discussions on what is “original work” and how technology can enhance creativity. Practitioners, including students, are highly motivated to develop their skills, but may also be eager to use new technologies to create. While there are many opportunities, concerns exist around ethical attribution of sources and copyright violations. This is an evolving field with companies attempting to add attribution into their processes. Current cases already exist of using GAI for

creative brainstorming, or development of final artifacts, as a partner to enable higher level creation by the artist, and many other uses. The rest of the academy might look to creative fields for help working through their own disciplinary considerations of how GAI tools might change notions of original work. [Appendix C](#) discusses these considerations in more detail.

Section 3.2.3: Courses in the social sciences

In the social sciences, the advent of GAI raises particular concerns for the written assignments, homework, papers, and exams that are a core component of student work in many courses. Passive reliance on GAI by students to generate literature reviews or written work risks undermining the learning objectives of assignments, producing poor quality work, and violating academic integrity standards. Instructors are also encouraged to explore ways to purposefully incorporate GAI into social science courses in ways that enhance student learning. This can include having students evaluate GAI output and explore ways to test its validity. [Appendix D](#) and [Appendix B](#) discuss use cases.

Section 3.2.4: Mathematics, physical sciences and engineering

Technical and mathematical courses have adjusted well in the past to incorporate new technologies, such as computing and visualization tools. GAI may provide similar opportunities to enhance education in the space of mathematics, physical sciences, and engineering. For example, students and instructors may use LLMs in an explanatory capacity or use LLMs to synthesize code to support data analysis and visualizations. Some of the biggest concerns with current systems is their inaccuracy (“hallucinations”) and circular reasoning. Instructors should make themselves aware of the capabilities of current systems and the fast changing behavior of these systems on mathematical and engineering problems. We recommend educating students on the capabilities and limitations of these systems, prohibiting their use where basic skills need to be developed, and encouraging their use in cases where LLMs can improve student learning. [Appendix E](#) gives detailed examples of use cases for courses in this domain.

Section 3.2.5: Courses in programming

GAI is already used heavily in industry to assist in coding through applications like GitHub Copilot. Opportunities for LLMs in programming education exist in: the generation of code from specifications (text to code), the generation of ancillary tools such as tests (code to code), and the generation of explanations or suggestions (code to text). However, the concern is that students will rely on GAI and will not learn the skills necessary to generate working, understandable, and updatable code. They may be unable to move beyond the solutions favored by an AI system, to identify and fix problems, or at worst, to even recognize that alternatives exist. We recommend using GAI in advanced courses as a tutor or helper in programming, but not as the sole creator of code. See [Appendix F](#) for details.

Section 3.2.6: Courses in law

For law, GAI threatens the integrity of the take-home exams that are a common feature of many courses. For foundational courses, particularly in the first year core, use of in-person written exams with restricted access to the internet and ability to access GAI is recommended to ensure the validity and integrity of exams. At the same time, GAI tools are in increasingly widespread use in the practice of law and it is important that this shift be addressed in legal education. This could be done through addressing use of GAI in legal practice explicitly in second and third year classes, including examination of these tools in legal

research and writing courses. The strong ethical core of the discipline and practice of law should be reflected in how GAI is addressed. [Appendix G](#) elucidates further.

Section 3.3: Use of GAI by instructors for course content creation

Instructors can use GAI to create content; for example, as a first draft for course structure/syllabi, lecture structure, examples, figures and diagrams, etc. Instructors can also generate large banks of practice problems or assessment questions, though it is important to validate any questions assigned to students for accuracy and appropriateness.

We recommend that instructors also follow the guidelines of attribution if they choose to use GAI to produce course materials. This way, faculty can model for students how to use GAI with attribution. This will also provide clarity for students about where GAI is not being used and avoid assumptions by students that they are being provided educational material that the instructor has not personally created or vetted.

While GAI may have selective utility in assisting in providing feedback for low-stakes formative assessment (for example in practice problems), we currently do NOT recommend it be used in summative evaluation of student work. Evaluation and grading of students is among the most important tasks entrusted to faculty, and the integrity of the grading process is reliant on the primary role of the faculty member.

Section 4: Recommendations for faculty and administration

Based on the aforementioned opportunities and concerns, we believe that the use of GAI technologies can be integrated into teaching in ways that enhance learning objectives, but that these implementations must be accompanied by strategies to improve students' understanding, and practice of academic integrity. Such strategies may include: 1) instructing students on the necessity of academic integrity, and what it constitutes; 2) guiding students toward scholarly and applied practices consistent with academic integrity; and 3) clarifying faculty's intentions around learning outcomes. Students should be taught why using GAI in prohibited ways is not just unethical, but also counterproductive to learning essential content and skills. In addition, faculty must instruct students in best practices for using GAI.

We make the following recommendations for **faculty**:

1. Faculty should be explicit in identifying expectations regarding the use of GAI tools in each course, and potentially for individual assignments. Cornell resources such as the [Center for Teaching Innovation](#) may be helpful in identifying standardized language and clear examples.
2. Faculty are encouraged to identify well-defined learning outcomes to provide rationales for how and when GAI can/cannot be used in a particular course.
3. When GAI is permitted, faculty should be clear about student expectations in terms of documentation and attribution, what work is expected to be produced by the student themselves, and how the student is expected to validate or verify output from GAI.
4. Faculty members are encouraged to engage in ongoing conversations about the importance of academic integrity, including the fact that basic academic integrity principles remain important and

still apply regardless of the existence of GAI tools. (See "[Communicating Why Academic Integrity Matters](#)").

5. Integrating critique of current practices and uses of GAI, including ethical issues, into all stages of learning is vital.
6. We currently *discourage* the use of automatic detection algorithms for academic integrity violations using GAI, given their unreliability and current inability to provide definitive evidence of violations.
7. While faculty may use GAI as a tool for developing teaching materials, we encourage them to adhere to the same standards of attribution that they require of their students.
8. We do not recommend the use of GAI for student assessment.

The Center for Teaching Innovation is available to consult with departments and individual faculty on how to best implement these recommendations.

We make the following recommendations for **university administrators**:

1. The Code of Academic Integrity should be updated with clear and explicit language on the use of GAI, specifically indicating that individual faculty have authority to determine when its use is prohibited, attributed, or encouraged, **and that use of GAI on assignments by students is only allowed when expressly permitted by the faculty member.**
2. When considering a move (back) to more in-person assignments and assessment, policies and practices should consider how doing so could have a disproportionate impact on students with disabilities and other marginalized students.
3. The university should recognize the additional burden on instructors to adapt to the rapidly changing effects of GAI on education, and provide additional support to teaching faculty and staff.
4. The university administration, in consultation with faculty and academic staff, should develop and issue best practices on assessments, in light of the growing tension between the need to ensure academic integrity and the need to ensure access and inclusion for marginalized students. Specifically, aligning pedagogical practices with [Universal Design for Learning \(UDL\)](#) can promote fuller access and inclusion for all students. While this does necessitate rethinking the current design of classroom and assessment practices, doing so can achieve the dual goals of greater access for students and appropriate integration of AI tools into the classroom.

Finally, GAI technology continues to increase in capability and ubiquity. Tech companies are actively working to incorporate GAI in every aspect of their products, making it increasingly difficult to avoid or even identify its use. The recommendations in this report should provide a framework for immediate and future actions, but they are not the last word. There must be procedures in place to monitor new advances, communicate new capabilities widely, and adapt policies and course technologies.

Section 5: Conclusions

The impact of Generative AI tools on education is likely to grow over time. The use of these tools already threatens some standard educational approaches and poses challenges for academic integrity. At the same time, GAI is likely to become an important tool across many domains and students must learn about its

strengths and limitations. If used thoughtfully and purposefully, GAI has the potential to enhance educational outcomes. For these reasons, we recommend that Cornell adopt a forward-looking approach that incorporates the use or non-use of GAI specifically into learning objectives.

Our core recommendations to faculty are that they reconsider their learning objectives in light of GAI tools, and incorporate explicit directions regarding use of GAI into their syllabi and assignments. We recommend that faculty formally adopt one of the three different approaches, depending on the learning objectives of the course or assignment.

- **Prohibit** use of GAI where its use would substitute for or interfere with core learning objectives, particularly in courses where students are developing foundational knowledge or skills.
- **Allow with attribution** the use of GAI where it can serve as a useful resource to support higher level thinking or skill development.
- **Encourage** use of GAI in courses or assignments where it can be used as a tool to allow exploration and creative thinking, or level the playing field for students with disparate abilities and needs.

Our core recommendation to the administration is to provide material support to faculty as they grapple with adapting individual courses to the new reality of GAI tools. For example, the administration should provide assistance in implementing accommodations for new assignment and assessment mechanisms, provide additional TA support when needed for course redesigns, and support faculty as they implement new teaching techniques that may be unfamiliar, and initially perhaps unwelcome, to students.

To guide students to obtain the potential benefits from GAI in enhancing higher order thinking and learning, and to avoid the dangers of GAI undermining the gain of key skills and knowledge, Cornell must take a proactive approach to the use of GAI in education. Our students need to understand both the value and limitations of GAI, not only because they will encounter it on a regular basis in their future careers and lives, but also because many of them are likely to guide its development and use in the future.

Appendix A: State of the art in Generative AI

This appendix gives a layperson’s description of how Generative AI models are trained for various domains like language, coding, images, and other modalities, and how these models are deployed to users in current systems.

Section A1.1: Language

Generative AI models of natural language are known as Large Language Models (LLMs) or Generative Pretrained Transformers (GPT, from OpenAI’s branding). It is worth breaking these acronyms down to understand the current state of this approach. “Large” refers to the capacity of the underlying system and is proportional to the amount of data it is trained on. Current open large models are trained on more than 1 trillion word tokens, whereas proprietary models are likely trained on orders of magnitude more.

Transformer is the name of the model design used for language applications and is critical to their generative ability. The current generation of LLMs for human language are created with a two step procedure. First, the models are pretrained on a large set of text collected from articles, published books, research papers, social media, and other collected sources. Then a second Instruction training stage is applied to train the models to be helpful and responsive to common human instructions. This second stage data comes from human interaction with the published system, as well as large teams of human annotators who write answers to challenging questions and provide feedback on model answers. This stage gives companies more control over system behavior.

There are several major commercial chat LLM interfaces, of which notable choices include: ChatGPT (from OpenAI based on GPT 3.5 / 3), Bard (from Google based on PaLM), and Claude (from Anthropic). These present a relatively similar interface where users can chat with the models and receive textual responses. Evaluation of these models is still challenging, but ChatGPT is currently considered more advanced than the others, particularly in terms of reasoning and factuality. For this reason, ChatGPT has become somewhat synonymous with Generative AI, and will act as the representative model for this category throughout the report. In addition to chat, LLM providers offer models to third-party services, which use them for products. Notable examples include DuoLingo, which offers a language learning bot, Grammarly, which uses LLMs as part of its pipeline, and Khan Academy.

There have been major efforts to produce high-quality openly-available LLMs. Most notable has been LLaMa (from Meta), a large language model that is available for extension and development. This is a first stage LLM trained on a large amount of available text data. The release of this and related models has led to several different community projects to produce an instruction-tuned LLM variant, including: OpenAssistant, GPT4Aall, Alpaca. These models can be used throughout HuggingFace Chat. Currently commercial versions are free-to-use, but given the cost of serving these models, it might be necessary in the future for universities to run open-variants of the systems. These models are also critical for supporting domain-specific language models for domains such as medicine, science, and mathematics.

Section A1.2: Code

While technically similar to LLMs for language, generative AI for code are impactful enough to be considered a separate modality. For code LLMs, the pretraining data is supplemented to include large amounts of code. Specifically, these models use code scraped from sites like GitHub. As with LLMs, it is not public knowledge what code bases proprietary LLM models have seen, which makes it challenging to know if specific problems are in the training data. There do exist open replications of this data collection process, e.g., TheStack, which expose this information for open models. Even without seeing any running instances of the code, these models learn both to generate working code, and also how to generate code directed by natural language (either from code comments or from language data). Code models can be used to explain complex code inputs. Models can further be instruction trained to act like responsive agents in a chat environment, even updating code conversationally.

In terms of code model usage, LLMs such as ChatGPT and Bard are the primary proprietary models. Both generate code, as well as amend it, based on users' corrections and updates. As with language, ChatGPT is currently better at producing precise code output, although Bard has added additional features for code, such as including citations to original sources. Another popular tool is CoPilot, based on OpenAI's tools, which integrates into a user's IDE and provides both powerful autocomplete as well as integrated question answering. Researchers are also developing open-source coding LLMs such as StarCoder, which are more transparent as to their training data and may allow for customized or constrained tools.

Section A1.3: Images

Generative models for images are also produced using a large amount of training instances consisting of text / image pairs from the web. Notably, though, the technology for image generation uses a different method, known as diffusion. Unlike in LLMs, which generate each word one at a time, image models synthesize images by adding detail to generation step by step. To handle text prompts, these models are paired with a LLM text model. One important consequence of this approach is that generative models of images do not need to be as large or trained on as many instances to perform well. As such, there has been more progress on development of open-source systems for generative image models.

The most commonly used systems for image generation are Stable Diffusion (originally an academic system, connected to Stability AI), Dall-E (from OpenAI), and Midjourney (a commercial system). These systems allow the user to enter a prompt and produce a relevant image based on the content and style requested. Image generation systems also allow for more complex edits, such as combining images, in-painting content, and tuning to specific domains.

Section A1.4: Other modalities

Generative AI is additionally being utilized in other modalities. Recently, many demos for video generation have shown that it is possible to generate short clips based on textual prompts, most notably being the model from Runway (Gen1). There are also speech-generation models, e.g., Vall-e (from Microsoft), that can mimic the voice of the speaker from a short example. Using similar techniques, companies have released music generation systems that can generate music from descriptions or from demonstrations, e.g. MusicLM (from Google).

Section A1.5: Generative AI in software tools

Current user-facing generative AI apps are primarily technical demos of the underlying technologies. Companies also provide or sell interfaces to the models and inference code to facilitate application development. Likely in the coming years, user interaction will happen through customized applications. One example of this mentioned above is Github CoPilot, a tool that uses OpenAI LLMs that is integrated directly into VSCode, a common programming tool. Another example of this are recent versions of Adobe Photoshop, which include Generative Fill to allow for automatic generation within image editing environments. Related tools like Canva have incorporated AI methods into design and editing processes in addition to full image generation. New user-facing tools are offering AI first interfaces that directly incorporate AI systems into their structure. For example, sudowrite is an online writing assistant tool that uses LLMs to complete and expand upon story ideas.

Appendix B: Courses that develop writing as a skill

When teaching writing skills and courses in which writing assignments form a key component of the learning objectives, GAI offers both opportunities and concerns. GAI may help students generate and improve outlines, find and evaluate sources, provide equal access to editing suggestions, and build student confidence as they learn to critically evaluate both their own writing and that produced by GAI. However, GAI can also undermine the ability of students to learn fundamental writing skills, such as basic grammar rules, sentence structure, and paragraph coherence, as well as mislead them with fake or biased content. While faculty may choose to prohibit the use of GAI in a course, we would encourage faculty to consider how incorporating some use of GAI, via specific guidelines, supervision, and dialogue, into in-class activities and certain assessments can offer unique and important pedagogical opportunities.

Opportunities

Generative AI tools offer significant opportunities for assisting students with the writing process, from generating outlines to evaluating one's own draft or other written materials; by identifying and evaluating key points for logic, flow, and evidence; to a host of editing capabilities. This technology not only offers a potentially more efficient means of engaging in planning, drafting, and editing, but also a customized and adaptable mechanism for offering personalized feedback to students, helping them to reflect on their writing choices, and to build and improve upon foundational writing skills. Further, these tools can assist faculty in [responding to the needs of individual students](#), offering tailored suggestions, prompts, and resources based on students' strengths and weaknesses.

An essential part of most writing pedagogy and practice involves peer editing. Using GAI as a peer editor can address two problems that typically arise with peer editing: (1) It removes the social comparisons that can keep peers from asking for and offering effective feedback, and (2) It provides access to a "competent" editor that puts in equal effort towards offering suggestions. Students' assessment and application of the feedback that GAI tools can generate may be best maximized, however, when students further analyze the generated suggestions for their efficacy in consultation with faculty. Incorporating dialogue about writing choices, which writing-intensive courses can do through mandatory faculty-student conferences, may both further enhance students' critical thinking and writing skills, as well as ensure that students do not passively over rely on feedback that may not be beneficial, either in the short- or long-term.

Ultimately, a guided approach to using GAI in writing courses can help students develop keener awareness of what makes for effective—as well as ethical—writing choices, and ideally, to spur greater confidence in their own writing abilities, independent of technological assistance.

Concerns

While the opportunities for GAI in the writing process are considerable, the concerns are significant. Using AI for generating text or peer editing raises concerns about academic integrity, plagiarism, authenticity, and the need for human judgment in the writing process.

Beyond the critical ethical considerations at stake, GAI tools offer other concerns that may detract from student learning outcomes in writing courses. Namely, GAI may discourage creativity and original thought

in student writing, potentially leading to homogenized output. If GAI output is relied upon solely without additional student analysis, reflection, and modification, students may not learn how to write effectively for a given audience, prioritize content and structure their writing, or learn how to make critical choices to strengthen their written arguments. Moreover, if used as a peer editor, GAI may have difficulty understanding the nuanced requirements of writing assignments, leading to less precise or relevant suggestions.

In addition, students—especially the very students who may feel most hesitant about their writing skills and interacting directly with faculty—may become overly dependent on GAI tools, potentially to the detriment of building foundational critical thinking and writing skills. Use of these tools may further mask challenges that students are experiencing with developing skills, which can become difficult to address later on. Finally, overreliance on GAI tools may also lead to decreased direct interaction between instructors and students, potentially impacting the quality of guidance and mentorship, which is essential in writing-intensive courses.

Recommendations

At the specific writing-intensive class-level, we suggest that faculty engage in a thoughtful discussion with their students and employ a guided use of GAI that can:

1. Emphasize what constitutes effective and ethical uses and applications of GAI, and thus allow for building of students' critical thinking and writing skills in being able to discern the limitations of the tools in practice;
2. Teach students the importance of analyzing and evaluating information independently of GAI tools, and how its use can inhibit students' ability to build both foundational and advanced writing skills; and
3. Clarify *why* developing certain foundational and advanced writing skills, along with good academic integrity practices, is fundamental to their academic and personal growth.
4. Prepare students to ethically use these tools in the workforce.

Case Studies

We offer several sample assignments that can be used in writing intensive courses. In addition to the below examples, the Knight Institute offers suggestions on how faculty might use GAI to help students [complete research papers](#) for their First-Year Writing Seminar courses.

Sample Editing Assignments

1. Editing Skills – pre-work for class discussion, in-class activity

Provide students with a paragraph of text that is relevant to your course, such as a particularly complex paragraph from a scholarly journal article or from a primary source.

Ask students to:

- A. Edit the paragraph on their own, using a prompt that makes sense for your course (e.g, make this paragraph more concise, strengthen the argument in this paragraph, explain the concept in this paragraph).
- B. Students run the original paragraph through GAI and ask it to do the same thing that students did in the first step.
- C. Have students review their editing choices as compared to the editing choices made by GAI and submit a final revised paragraph.
- D. Students complete a written or in-class reflection on why they included their final edits and how they were able to improve on what GAI provided.

2. Revise for new audience & reflect – graded assignment

Turn a two-page proposal into a one-page proposal for a new audience.

Ask students to:

- A. Using an assignment that students wrote earlier in the semester, identify a new audience and rewrite their assignment for this new audience (e.g., proposal was originally written for the manager and will now be rewritten for a vice president; proposal was originally written for a scientific audience and now needs to be rewritten for a lay-person).
- B. Use GAI to help shorten the original proposal.
- C. Critically analyze how GAI shortened the work, paying attention to what the new audience knows and needs to know.
- D. Using edits suggested by GAI and their own critical thinking, shorten the original proposal to one page.
- E. Utilize document design features (e.g., headings, sub-headings, bulleted lists or tables) to make the writing easy for the new audience to read quickly.
- F. Students write a reflection on how and why they modified their original proposal for the new audience, including how they were able to improve on what GAI provided.

3. Reverse outlining to strengthen structure and evidence–in-class or for outside of class

Ask students to:

- A. Use GAI to generate a reverse outline of a written draft: have it identify what the main focus/claim of each paragraph is, and what supporting evidence is within that paragraph.
- B. Reviewing what GAI has generated, students then consider whether the main claim that was identified for each paragraph is the one that they intended to be the topic of that paragraph. Further, does the evidence that GAI identified sufficiently support that claim?
- C. In subsequent written reflection (and/or in class discussion), students should assess what they found to be effective about the process in helping them pinpoint issues with structure, logic, and evidence. Students should also reflect on where GAI was limited or misguided in assisting them with reverse outlining.

Similar to assignment 1, students might first produce a reverse outline on their own, and then compare/contrast what they created with that from GAI, further reflecting on those differences.

4. GAI as a peer editor — in-class activity or homework outside of class

Peer editing can be a valuable tool to help students learn to edit their own and others' work, as well as learn how to respond effectively to feedback.

Ask students to:

- A. Ask GAI to offer constructive feedback on their writing, with specific suggestions for where and how to make improvements in grammar, logic, and flow (tailor this prompt to your desired outcomes).
- B. Students review the GAI feedback and make edits to their original work.
- C. Students respond to several reflection prompts including:
 - Which suggestions did you find to be effective? Not effective? Why?
 - As you review the feedback, did GAI help you notice patterns in the kinds of suggestions that you received? If so, identify and reflect on these patterns: were you previously aware of these patterns? Do you feel that GAI pointed out all of the different areas in which you have concern, or did it miss some?
 - How do you intend to apply the feedback you found effective in future writing?

Content creation assignments

1. Generate an outline for a written report

Ask students to:

- A. Use GAI to assist them in generating an outline for their report.
- B. Emphasize that GAI should be used as a tool for inspiration and guidance, rather than relying solely on its suggestions.
- C. Students modify and improve their outline, then evaluate it based on their logical structure, coherence, and adherence to the assignment requirements.
- D. In-class discussion or written reflection: Students critically analyze the usefulness of AI in the writing process.

2. Summarize arguments and produce themes

Ask students to:

- A. Use GAI to summarize a meeting transcript into key themes (e.g., congressional meetings, central banker transcripts, corporate earnings reports).
- B. Students produce a memo that prioritizes the themes and adds contextual information on power dynamics or other relevant background information.
- C. In-class discussion or written reflection: Students critically analyze the usefulness of AI in the writing process.

3. Brainstorm ideas

Ask students to:

- A. Use GAI to brainstorm ideas for an assignment that is relevant to the course (e.g., topics for a research paper, research study ideas, new business ideas, marketing taglines).
- B. Students evaluate the ideas based on the established criteria for the assignment.
- C. Students further develop their preferred idea and complete the assignment.
- D. In-class discussion or written reflection: Students critically analyze the usefulness of AI in the brainstorming process.

4. Assess validity of sources for accuracy and bias

Ask students to:

- A. Use GAI to find sources for an assignment relevant to the course.
- B. Students review sources, evaluating the usefulness and relevance to the assignment.
- C. Students conduct a similar search using library resources.
- D. Students produce an annotated bibliography that assesses the quality, relevance, and reliability of GAI and library sources.
- E. In-class discussion or written reflection: Students critically analyze the usefulness of AI to find reliable and unbiased sources.

Classroom discussion assignments

1. Compare/Contrast

Ask students to:

- A. Use GAI to generate content on a particular topic that is relevant to the course (e.g., academic writing vs. business writing, Greek civilization vs. Roman civilization, differential calculus vs. integral calculus).
- B. Assign a reading or book chapter on the same content.
- C. In-class discussion or written reflection: Students critically analyze what GAI got correct, incorrect, or missed completely.

2. Pro/Con debate

Ask students to:

- A. Use GAI to generate pro and con arguments for a debate that is relevant to the course.
- B. Review course material and readings to augment and strengthen these arguments.
- C. Come to class prepared to engage in a debate with their classmates.
- D. After the debate, discuss which arguments were strongest and why.
- E. In-class discussion or written reflection: Students critically analyze what makes a more effective argument,

Appendix C: Creative courses for music, literature, and art

For creative fields such as art and music, AI was a topic of interest and concern before the release of ChatGPT in November 2022. AI tools that “create” artistic images and “compose” music based on existing images and corpi of musical work have been available longer, and artistic communities are already engaging in robust conversations about how AI can contribute to creation. These conversations have expanded since the release of ChatGPT and the greater attention on AI. We use the more general term “AI tools” here, since it is more appropriate than only large language model (LLM)-based tools such as ChatGPT.

Artistic fields have long standing traditions of discussing what it means for an individual to create “original work”, although this conversation is constantly shifting as it engages with new technology. Disciplinary practices around ideas of authorship and practices for attribution seem well suited to guide faculty and students in discussing whether, how, and when to integrate AI tools into their creative practice, and how to properly and ethically attribute AI generated work. This is not to say that there is consensus or agreement on how questions of authorship or originality should be answered (see the recent Supreme Court Andy Warhol case), but that such conversations fit into prevailing disciplinary discourses. The rest of the academy might look to creative fields for help working through their own disciplinary considerations of how AI tools might change notions of original work, attribution, and creation.

Additionally, students in artistic disciplines are highly motivated to develop the necessary skills needed to create and interpret art. They generally do not need to be convinced of their need to learn specific skills, or to be warned away from tools that may be able to do that work for them.

Opportunities

In addition to LLMs that write prose and poetry, artificial intelligence tools can generate music, images, and video. This provides many ways in which instructors can **embrace** these tools, building them into assignments (including examples shared below). One typical way is for AI tools to generate source material for the student to then build upon, riff off, interrogate, or extend in their work.

These tools are developing and improving rapidly. Text to image or text to video tools such as Dall-E, Stable Diffusion, MidJourney, and Adobe Firefly, are easily available, with new versions often in beta environments. Runway ML is a platform that links various AI models for artists to experiment with, generating and combining text, images, video, music, and more. Artbreeder allows users to create original artworks by combining and evolving existing art.

In addition to GPT-3 and GPT-4, there are several writing tools that can generate many types of writing, from marketing copy to short stories and more. For example, for creative writing, LAIKA lets you collaborate with an AI partner. LAIKA can be trained to coach you using your own writing, or you can “collaborate with writers” ranging from Fyodor Dostoevsky to Agatha Christie.

Concerns

Concerns around the use of AI include inequitable access to tools; concerns that overreliance on AI tools built from particular databases can actually limit creative innovation; and risks of appropriation without

attribution, including harms to individual creators, such as illustrators and musicians whose individual style is easily co-opted by such tools.

These tools often charge for use beyond a limited amount of free trials. There are also many different AI tools, so keeping up to date with available tools will be a significant time commitment. If Cornell decided to provide AI tools to students to reduce inequity issues, selecting representative AI tools for creative fields would be challenging.

The tools currently available draw from a limited range of human artifacts, skewed toward what has been posted online. Artistic creation that draws only on this content may, over time, limit artistic innovation and narrow the scope of artistic work. Critical engagement with these questions will become an important component of education.

Because this source content is disassociated from the identities of its original creators, it also brings unique risks of appropriation and makes **attribution** more challenging. AI tools may disrupt how artistic skills are valued and compensated. These tools threaten the livelihood of individual creators (especially illustrators and others whose individual style has been used to inform the development of tools like Dall-E and Midjourney). If used too heavily/without consideration, they can exacerbate imbalances in whose experience is represented by skewing any output toward the existing dataset.

Recommendations

When considering the impact of LLMs, and how to best provide an equitable learning environment for students, it will be important to consider the range of AI tools relevant to creative disciplines, and the extent to which they are integrated into the current workplace.

To allow students time to develop their own design sense/creative perspective, instructors should include assignments that explicitly **prohibit** the use of GAI tools. Given that students value their individual, creative contributions, instructors should build on this sense of pride and focus on skill-building when encouraging students to avoid overreliance on GAI tools.

Instructors should include critical discussions of the sources of data that are used to build these tools in their pedagogy. There are already conversations around the [images that can be elicited from these tools](#). Such artifacts are interesting in themselves and offer a potential site of critique.

Case studies

We are already seeing the use of GAI in assignments. Some examples of **encouraged** use in music include:

- as a generative tool for images, video, and text as poetic materials to respond to musically. For example, students are invited to generate images from poetic input prompts to Midjourney, using them as inspiration for their compositions.
- as parameter mediation in electronic music, where the AI (or more correctly, a neural network) makes lower/mid-level decisions in a given synthesis system so that the performer/composer can operate on a higher, more "instrumental" level of control.

- in manipulation of large audio data sets (called a "corpus"), letting AI seek, discover, and organize relationships between and among sounds (or pieces of sounds), so they can then be navigated and accessed by the composer or performer for musical purposes.

When an assignment is designed to teach the use of specific tools, then AI might be **prohibited**. For example, if a lesson is meant to teach students how to create content “by hand” using a specific digital imaging software, they should not be allowed to use other GAI tools such as text-to-content to produce a similar result.

Attribution is expected in any discussion of the artistic process, and instructors may want to explicitly require students to discuss their experiences interacting with AI tools when used.

Appendix D: Courses in social sciences

In the social sciences, new forms of generative AI (GAI) are both a topic and a tool for teaching and inquiry. Tools such as ChatGPT may provide new case studies in our classes and assignments. But they also touch upon the very notion of what it means to learn about society, given that these systems tend to reproduce and mimic many of the analytic strategies and methods at the heart of social science disciplines, including literature reviews, textual analysis, exploratory writing, as well as the collection, analysis, and interpretation of data.

Opportunities

There are several ways in which GAI offers opportunities for education in the social sciences. Given that some students have already begun to incorporate GAI tools into their study habits, finding ways to incorporate GAI into courses might be a fruitful way forward for many instructors (while keeping the concerns about them in mind). It would also add a potentially useful “real-world” skill. Possible ways in which GAI may be included into a course or sanctioned as a way of studying (see [Appendix B](#) for detailed case studies):

- to summarize specific publications/books,
- to develop paper outlines,
- to brainstorm hypotheses,
- to garner editing feedback,
- to serve as an interlocutor for developing specific arguments

Instructors can also go beyond merely updating assignments to incorporate GAIs by using them pedagogically to improve [meta-cognitive strategies](#), such as critical thinking and careful evaluation of source materials. This might involve asking students to critically evaluate GAI content, which asks them to reflect on the quality of ideas, feedback, outlines, etc.

Concerns

The widespread use of GAI raises a number of concerns specific (though not necessarily exclusive) to the social sciences. Most importantly, these systems have been shown to be unreliable as sources of analysis and information, often suggesting non-existent facts, misattributing ideas, or inventing sources that may appear deceptively authentic. Therefore, using these tools effectively for learning requires a deep understanding of their strengths and limitations, not unlike the situation students and instructors faced in the early days of Google Search or Wikipedia.

The most pressing concerns arguably pertain to out-of-class activities and assignments. Many of the traditional social science methods and techniques focus on the analysis, interpretation, and production of text—tasks that tools like ChatGPT are claimed to do well, but in practice, often fail to conduct. In addition to the already mentioned challenges of accuracy and reliability, concerns include:

- the loss of writing as a method and a way of thinking in its own right, including the devaluation of seemingly “tedious” work of reading, writing, and rewriting;
- the reification of existing problems in academic scholarship, such as the uncritical adoption of biased citation practices, clunky prose, and formulaic methods;
- the possibility of students taking “shortcuts,” especially under time pressure, and the limited ability of instructors to verify original work;
- the blurring of authorship in exercises and assignments, especially when assistive devices are heavily used in the production of essays, short answers, and data analysis;

- or, the loss of critical capacities when confronted with elegantly written, but methodologically flawed AI-generated reviews of literature and data.

If students rely on GAI passively to merely generate literature reviews or other written content for them, they risk losing the opportunity to build crucial research and analytical skills.

Recommendations

Individual instructors will need to work out if and how they might be able to incorporate GAI use into their courses, ideally emphasizing critical use of GAIs given their limitations at that point in time. If a prohibitive approach is adopted, assignments and prelims allowing computer-use might still be possible if the instructor takes advantage of GAI limitations. For example, asking students to provide their own reflections on class materials or integrating insights from different lectures is likely to make GAI less useful. A potential caveat, though, is that such limitations are moving targets as GAI improves across time. Instructors will need to keep abreast with such changes to prevent potential academic integrity violations.

Integrating GAI into our pedagogy via classroom exercises and discussions might address some of the concerns above, helping us to emphasize thinking and writing as a method and process, and not merely an output. Given the speed of GAI development and the proliferation of more specialized applications, courses that incorporate GAI use will likely need to be routinely updated. Clear and explicit instructions about attribution of GAI use will need to be in place.

Students' knowledge of the kind of data used in specific social science areas and how such data is interpreted in light of theories is often the focus of in-class evaluations or homework. Assignments can be designed in this context to encourage students to engage critically with GAIs. Given the fast pace of GAI development, it might be prudent to have students always note the date of their responses on such assignments.

Appendix E: Mathematics, physical sciences, and engineering

Opportunities

Technical and mathematical courses have adjusted well in the past to incorporate new technologies. For example, the widespread infusion of computing and visualization tools such as Matlab over the past two decades has allowed instructors to ask more of students and to assign exercises to help students gain deeper understanding. As a complement to solving a differential equation, for instance, students can plot out the solution and visualize how results depend on key model parameters. While GAI is not precisely analogous to these methods, there are similar opportunities to enhance education in the space of mathematics, physical sciences, and engineering. These opportunities include utilizing LLMs in an explanatory capacity for technical topics and supplementing technical topics with the ability to synthesize code to support data analysis and visualizations.

While current GAI systems struggle with producing novel mathematical reasoning, they are adept at retrieving and summarizing complex arguments on technical topics. Although they do still often hallucinate or skip steps in the process, a careful and curious student can use them as a supplement to explore a topic. As an example, we consider open-ended physics questions. An instructor might make an assignment that asks a student to have a dialogue with an LLM about a specific question, e.g., “would I weigh more at the equator or the north pole?”. The student could then turn in the transcript plus a reflection about what they have learned. LLMs can also be used in the reverse direction to assess the reasoning of the student in a devil's advocate-like scenario. If a student is asked to make an argument with technical justification, they could be asked to utilize an LLM to assess their own reasoning and justify why or why not the LLM's critique of their technical argument is valid.

In their current form, LLMs present a nice opportunity for data analysis and visualization. Current models are excellent at generating complex graphs and visualizations from user specifications. For example, in an assignment collecting experimental data, an instructor could ask for a graphical presentation of results in a precise manner. Students could then use an LLM in conjunction with a coding environment like Jupyter to produce and iterate on their output. This assignment is possible even in courses where students do not have advanced programming knowledge, as they can observe the output of the system. Similar approaches are also possible for data science and analysis assignments, where LLMs allow students to query and filter data without having to fully understand code output.

Concerns

While the current generation of LLM systems are not yet capable of solving unseen mathematics problems due to limits on their reasoning and computational power, they are extremely capable of replicating and combining previously seen solutions to commonly used problems. As such, it is critical that instructors are aware not only of the capabilities of current systems but also of their behavior on specific problems.

The most transparent short-term concern for technical classes is the student use of LLMs to shortcut learning objectives on homework assignments. Specifically, if students become accustomed to using LLMs in intro classes where it returns correct, or even partially correct answers, they may become reliant on them in more challenging classes or in the workforce. We caution that the capabilities of these systems, specifically in terms of mathematical reasoning, are increasing rapidly, and that just in the last several

months systems have improved greatly in this area. For classes where material is solvable by these systems, and the instructor believes this is problematic, we recommend both openly discussing the benefits of manual computation and course learning objectives with the class, as well as establishing clear guidelines about its use with the course. Additionally, instructors may consider altering assignment specifications or including additional requirements that deviate from standard versions of the problems.

Current LLM systems are more than willing to provide answers for questions that they are nowhere close to being able to solve. For example, when asked to solve a quadratic equation, the current generation of ChatGPT will proceed through the steps of solving by producing hallucinated intermediate values, despite not having access to a calculator. A striking aspect of these responses is that they are written in a tone that mimics the confidence of a textbook. The mismatch between the tone and the technical ability can lead to overconfidence from the reader. For this concern, we recommend an assignment in intro classes that requires students to go through the exercise of understanding these issues. For early proof-based courses, we suggest that students be asked to "provide a plausible proof generated by a language model and to find gaps or mistakes in the proof."

Recommendations

Our overall recommendation is to encourage productive engagement with GAI while highlighting potential pitfalls.

For productive student use, we encourage communicating regularly and clearly to students about when these tools can and should be used. Just like we ask students to solve differential equations by hand when Wolfram Alpha can solve it, instructors should make clear the need to practice and apply concepts, principles, and methods.

Even when LLMs can be used, it should be made explicitly clear to students that they are responsible for their own work, even if they used AI to generate all or part of a solution. If the AI answer is wrong and the student turns it in, the student is still responsible for the submitted answer. Provide examples where they can see how it fails in practice.

For instructors, we encourage understanding the precise abilities and limitations of these models. Submit problems to a LLM before assigning them to students to see what it can solve. What mistakes does it make? What grade would it get? Is this an assignment you still want to give and if so, do you ban the use of LLM, or allow its use with attribution? For example, in an engineering mathematics course where learning to solve certain types of equations is a learning outcome that will be assessed by in-person exams, you may wish to ban the use of GAI. But if the assignment is to use solutions to probe or to understand the physical problem at hand, you might ask students to attribute the use of AI to solve the problem.

Sample assignments

We demonstrate some examples that can be posed to GAI (ChatGPT in this case) and the responses from GAI to help illustrate educational opportunities and issues, (ChatGPT responses edited for brevity). These could be used as written HW assignments or as points of discussion in recitation sections.

1. Conceptual question in physics

Ask students to:

- A. Prompt GAI with a question such as: “If I weigh myself at the equator will I weigh more, less or the same as at the North Pole?”

ChatGPT (for example) will return:

Yes, your weight would be slightly different at the equator compared to the North Pole, but the difference is quite small and may not be noticeable without highly precise equipment. This weight difference is primarily due to the variation in Earth's shape and centrifugal force caused by its rotation. ...

- B. Then ask students to perform the analysis underlying the qualitative answer above and to compute numerically their weight at the pole and at the equator as a way to gain understanding into the magnitude of this effect.

2. Mathematical proofs

Ask students to:

- A. Prompt GAI to “Prove that the solution to the 2D steady-state heat equation is unique”.

ChatGPT (for example) will return:

To prove the uniqueness of the solution to the 2D steady-state heat equation, we can use the method of contradiction.

Consider the 2D steady-state heat equation with homogeneous Dirichlet boundary conditions:

$$\nabla^2 u = 0,$$

where $u(x, y)$ represents the temperature distribution in the xy -plane, and ∇^2 is the Laplacian operator. Suppose there are two solutions to this equation, denoted as $u_1(x, y)$ and $u_2(x, y)$, that satisfy the same boundary conditions.

- B. Ask students to assess the accuracy of the proof (in the above example, the solution is not unique, but nonetheless ChatGPT “proves” that it is, in this case, by restricting the boundary conditions).

- C. As a follow up, and along the lines of teaching students to use GAI prompts, students could be asked to edit the prompt to produce a correct response. For example, given the prompt “Is the solution to the 2D steady-state heat equation unique?”, ChatGPT returns:

No, the solution to the 2D steady-state heat equation is not necessarily unique. The uniqueness of solutions for the 2D steady-state heat equation depends on the specific boundary conditions and the geometry of the problem.

D. Then ask the student to provide examples of problems with and without unique solutions.

3. Understanding assumptions baked into engineering analyses

In engineering problems, there are often many underlying assumptions that students may or may not be aware of.

Ask students to:

A. Prompt GAI with a question such as “What are the underlying assumptions in beam theory?”

ChatGPT (for example) will return

1. *Linear Elasticity: Beam theory assumes that the material of the beam behaves elastically and follows Hooke's law. It assumes that the beam material is isotropic and that it deforms linearly under applied loads.*
2. *Small Deflection: Beam theory assumes that the deflection of the beam is small compared to the dimensions of the beam itself. This assumption ensures that the geometry of the beam remains relatively unchanged during deformation.*
3.

B. Ask students to reflect on and assess the accuracy of the ChatGPT response (in this case the full ChatGPT response is mostly, but not fully correct), or to provide examples of applications where the underlying assumptions (at least those that are correct) are no longer valid.

Appendix F: Courses in programming

Generative AI has shown impressive ability to solve coding tasks, particularly when provided a clear specification of the goals. Researchers have shown that the current generation of models can already solve most of the programming assignments given in introductory classes. As such, programming is one of the areas where it is most critical for instructors to both be aware of the abilities of current models and to incorporate these tools into their future planning.

Current LLMs are trained extensively on program code repositories, so their impact on programming education is of particular concern. Programming is a challenging and game-changing skill. But learning to program requires understanding many abstract concepts, and an extensive knowledge of languages, package APIs, and system environments. Generative AI can both help and hinder these learning objectives.

Two factors are significant in how students relate to programming courses and what support they need. First, courses vary in the expected programming skill: a student just learning about variables, functions, and control structures has different needs than a student who is comfortable with programming. Second, courses vary in motivation: some are intrinsically motivated and aim to increase programming skill, while others are extrinsically motivated and aim to build tools or products that happen to require programming.

In an introductory programming course, the goal is to help emerging programmers gain core skills such as reading code, mentally tracing execution and variable states, and identifying bugs. For such a course, GAI code generation is actively harmful to student learning, because students can generate code that meets specifications without understanding it. Rather, the better use of GAI would be as an "assistant", offering suggestions and helping students get unstuck. In a more application-oriented programming course, such as web development or introductory statistics, direct code generation may be a way to get students to a high level of performance quickly, creating curiosity and empowerment rather than boredom and confusion. Finally, in more advanced classes, students face complicated APIs and frameworks and alternative programming languages. In such cases, generative AI can be a "turbocharger" that decreases time spent consulting tutorials and documentation.

Opportunities

Three primary opportunities for LLMs in programming education are:

- generation of code from specifications (text to code),
- generation of ancillary tools such as test cases (code to code),
- and generation of explanations or suggestions (code to text).

The relative value of these interventions depends on course objectives. There are ample opportunities for teaching the use of LLMs as a programming tool, both to facilitate more efficient programming and to provide guidance to students learning the discipline. It is clear that generative tools will be a valuable and inescapable part of programming practice.

The first area of opportunity is to use generative tools to facilitate students' building of more complex artifacts as part of their training. Building realistic software systems is challenging and often requires writing significant amounts of "boilerplate" code to interact with external libraries and tools. As such, even

in programming intensive courses, it is common to simplify systems or provide large amounts of scaffolding code. When combined with careful instruction, the ability of LLMs to efficiently generate code for specific use cases may make it easier to have students move closer to the realistic environments. For example, current systems can generate code for basic user interfaces from user descriptions. While this may not be critical for the material in the course itself, improving the appearance of the final output can increase student motivation.

A related area of opportunity is supporting software engineering and encouraging best practices for software development. For example, students are often taught to produce detailed specifications and provide unit tests for their systems, but many non-intro courses do not require students to carry out this activity as part of assignments. LLMs can be effective at generating test cases from specifications with much less manual effort from students. This ability provides opportunities to incorporate these best practices into courses directly.

Finally, while it is not yet possible with current systems, future generative systems may open up programming and data science to those without knowledge of the programming language. This presents a remarkable opportunity to teach students for whom programming presents a large barrier of entry, but who nonetheless have an interest in using computer science tools for their applications. Students will be able to build useful systems for application-oriented projects without studying introductory programming.

Generative systems are also potentially a valuable tool for supplementing the teaching process of programming classes. It has already been reported that Harvard's CS50 course is planning to add a tutor bot based on ChatGPT. Students benefit greatly from having teaching assistants answer questions to get them unstuck. However, they can find it intimidating to ask questions or feel too behind to know what to ask. Their reticence leads to widespread use of online resources that are poorly curated and often misleading. GAI can potentially provide better targeted answers to specific issues in code than online forums. In addition to being able to retrieve specific issues that others have experienced, generative models can provide targeted responses to users' code and errors. Current models already provide explanations of code snippets, corrections of mistakes, and guidance for where to look in codebases. Additionally, there are numerous tools to integrate this guidance into code editors and search engines. With the caveat that advice as currently provided is poorly targeted to the specific level and learning objectives of students, this capability could improve the learning and debugging process.

Concerns

At its best, programming is empowering: a few simple primitives can be combined to create an infinite variety of customized solutions. Predictions of the demise of programming due to GAI seem premature. GAI does not replace the core abilities necessary to generate working, understandable, and maintainable code. The primary risk of GAI is that it may prevent students from engaging with the full expressive power of coding. Students may be able to achieve powerful results with little skill, but be unable to move beyond the solutions favored by an AI system, to identify and fix problems, or at worst, to even recognize that alternatives exist.

While intro courses emphasize small, well-structured problems, a large part of software engineering jobs is understanding and managing very large complex code bases written by others. We see this skill becoming

even more important in a world where some fraction of code is written by AI systems. These heterogeneous code bases, written both by humans and AI, will require extra care to understand the limitations of each type of coder. Cornell will need to continue to train students to manage these systems, and it would be a major negative outcome if the existence of AI coders led to an assumption from students that it is no longer important to navigate and reshape large code bases.

Currently, there is a high likelihood that LLMs are widely being used as a shortcut for students in programming-heavy classes. LLMs are very good at many aspects of programming, and are particularly adept at the type of self-contained, short programs in intro programming classes, because they have been trained on many similar programs. They are also adept at translating program specifications directly into working code. For this reason, it is important that instructors convey the learning objectives of their courses in a way that motivates students to learn the underlying languages themselves.

We caution instructors against aiming to outwit the ability of models to solve problems by underspecifying details of assignments; these techniques are likely only beneficial in the short term, and future models will be able to work around these issues. In some cases, it may be possible to have students specify assignments themselves, which at least requires understanding the request and being able to check that the code is correct.

Even in instances when a system is able to generate effective code for an assignment, there is concern that systems generate “Frankenstein” code by gluing together different techniques for solving a problem. This may lead to a short-term answer, but will be hard to maintain and work with in a large-scale code library. This is a good opportunity for teaching elements of high-level engineering through either critiquing the inconsistencies of functions output by a model, or by learning to reprompt a system to produce cleaner and more consistent outputs.

Recommendations

As with other areas, the key goal is to convey to the student why it is critical to learn how to read, understand, and write code, as opposed to the goal of simply producing it. If LLMs are viewed as an assistant in the process of learning to write and edit code, they can be powerful assistants to learning and improving as a programmer. However, if they are treated as a black-box oracle, they can hinder the early developmental phase of coding. In this sense, they are like a friend with experience in the subject. Used as a tutor, they can provide guidance; used as an answer key, they can be a dead end.

The other recommendation is that it is likely there will be no long-term method to prevent use of GAI. The conflict is that homework assignments in programming must be specified in sufficient detail for students to be sure they have done the work correctly and for course staff to grade their work, but any specification that is clear enough for these purposes will necessarily also be solvable by GAI tools. While it may be possible to craft specifications that thread this needle in the short term, the variety of programming chatbots available in the future will make this a time-consuming and ultimately futile pursuit. Course staff will need to know how current bots interact with existing and future assignments, and understand the capabilities they provide to students using them.

Sample assignments

1. Debugging skills – preliminary to coding assignments in intro classes

Provide students with snippets of code that have clear errors or issues with them or strange error messages. Teach how to use GAI as a debugging tool to help out in these situations.

Ask students to:

- A. Run the snippets of code through an interpreter or compiler.
- B. Diagnose the issue themselves based on the output and write a short description of the problem.
- C. Run the code snippet through GAI and construct a prompt to ask for its description of the problem.
- D. Compare the output of the model with the written description: was it sufficient, and does it lead to a different code change than the original student suggestion?

2. Specification skills – assignment in a class emphasizing software engineering

In a class where students are developing their own design and specifications for systems, assignments can explore the use of code generation as a way of translating specifications to code.

Ask students to:

- A. Develop a human-readable specification for a software system in written language, making clear user requirements and expectations.
- B. Write a prompt that communicates to a GAI system these specifications and examples of test cases that would enforce a subset of them.
- C. Run the prompt through a GAI system to produce test cases for the underlying specification.
- D. Assess the output code to argue whether or not it was able to provide full coverage of the expected test cases. Produce examples that either incorrectly pass or fail the tests.
- E. If necessary, reiterate on B.

3. Simulation and visualization – assignment in a non-programming class

In this assignment students will utilize code generation and graphing as a way to run simulations based on natural language descriptions that would have traditionally required more advanced programming knowledge.

Ask students to:

- A. Describe a scenario relevant to the class, for example:
 - a. In a statistics class, the results of summing up weighted coin flips
 - b. In a linguistics class, the co-occurrence of words in a document
 - c. In a biology class, the result of random mutations
- B. Write a prompt that describes the mechanism of the event in detailed language and ask for a graph describing the results of the simulation.

- C. Run the prompt through GAI and then run the resulting code to see the output graph of the system.
- D. Write a description of the result, analyzing whether the graph seems to correctly represent the underlying phenomenon and what aspects of the code either agree or disagree with expectations.

This assignment has potential pitfalls as it asks students to trust code they may not understand. However, the use of graphing and knowledge of the underlying subject should equip them with tools to at least notice and assess errors. We note as well that even in classes where students do know programming often convince themselves incorrect implementations are correct.

4. High-level LLM-assisted design – assignment in an upper level class

Provide students with an extremely detailed assignment specification describing the full requirements of the system. The goal of the assignment is to focus on system design with the use of GAI.

Ask students to:

- A. First run the specification as is through a GAI system to produce an output code base. This can be done through the API to produce a full longform output.
- B. Perform a code review of the AI output. Which parts are good, and which are poorly designed, inefficient, or confused.
- C. Construct a high-level design plan for the codebase: for example, giving the module structure and describing high-level implementation choices.
- D. Produce a detailed prompting plan describing this structure to the GAI system.
- E. Regenerate the code base, and submit the assignment.
- F. Student output will be evaluated based on this output as if they had written it directly. Goal being to convey responsibility for systems even with the use of AI tools.

Appendix G: Courses in law

The development of generative AI tools poses particular challenges for legal education due to the structure of the curriculum, common methods of testing, and professional norms.

Opportunities

Generative AI is being increasingly widely used in the practice of law and this trend can be expected to continue. Law students being trained as professionals in the field need to be prepared for the impact of generative AI on the practice of the profession. A historical analogy can be made to the revolution in how case law was researched when legal databases such as Westlaw and Lexis supplanted the traditional use of books to Shepardize cases. Similarly, the lawyer of the future will need to know how to use generative AI tools effectively and ethically to succeed in practice. Advanced coursework that addresses and develops knowledge about the use of GAI in the practice of law should be part of legal education. Law is a profession with strong ethical principles guiding practice and ethically appropriate professional use of GAI will need to similarly be addressed in legal education.

Generative AI may have uses in teaching legal writing through its ability to generate illustrative examples. This could allow generative AI tools to serve as a type of tutor for the law student in learning legal writing.

Concerns

The development of analytical skills and core legal knowledge through the foundational course work of the first year of the program is a key part of the law school curriculum. Use of generative AI in this setting risks undermining that foundational element of legal education (in several tests, ChatGPT passed the Bar Exam).

Take-home exams, which are widely used in law courses, are particularly vulnerable to the misuse of generative AI due to their unsupervised nature. Students who might be tempted to use generative AI inappropriately to assist in completing assignments may be led further in this direction by take-home exams that create greater time pressures. A return to in-person exams where the setting can control access to outside assistance may be necessary to ensure integrity in the testing process.

Recommendations

The use of generative AI should be proscribed in foundational courses in the legal curriculum where development of key analytical skills and knowledge is critical. Maintenance of the central role of final exams for assessment of learning in the first year curriculum in law will require adjustments to respond to the potential impact of GAI on this form of evaluation. Take-home exams should be avoided as an evaluation method due to the heightened danger of generative AI leading to academic integrity violations in this setting. In-person exams are a preferred option due to the ability to control and prevent outside assistance in that setting.

Judicious use of GAI tools could be incorporated into legal research and writing courses to help students master skills.

Ethical and practical issues involving the use of GAI in the practice of law should be explicitly addressed in legal education, so that students are prepared for it when they enter the profession.