Generative AI and the Curriculum

Introduction

Generative AI technologies have begun to revolutionize the ways students learn. While it is too early to confidently predict how they will change computer science education, it is instructive to consider some of the ramifications already apparent. In this section, a few specific implications of generative AI are explored by competency and knowledge areas. The overall observations follow.

- Educators can use generative AI to create course syllabi, design projects and assignments, and automate grading.
- Students can use generative AI for individualized tutoring, undergraduate research, enhanced productivity, etc. The skill set expected of students is changing, but the foundational knowledge they need to know is not.
- Tasks that are intellectually challenging, yet mechanical in nature will be automated. So, students must learn to work at higher levels of abstraction.
- Issues of society, ethics, and the profession are taking on much more importance and significance and must not only be actively addressed in the curriculum, but also periodically revisited and revised.

Implications by Competency Area - Software

Algorithmic Foundations (AL)

- The immediate impact of generative AI on Algorithmic Foundations is expected to be less than what is found in other knowledge areas since AL topics are primarily focused on students understanding the foundations of algorithmic computation. As an example, while generative AI may be prompted to produce an algorithmic solution utilizing less memory or improved runtime performance, students are expected to be able to evaluate the results of generative AI from an AL perspective of time-space tradeoff and complexity analysis.

Foundations of Programming Languages (FPL)

- The next paradigm in programming might be conversational programming that involves providing appropriate prompts to help generative AI produce and test desired programs. Generative AI is also likely to be able to translate code from one programming language to another, and from one paradigm to another. Program correctness proofs and validating program behavior against a specification may also be areas in which generative AI makes advances because of the intellectually challenging, yet mechanical, nature of the process.

Software Development Fundamentals (SDF)

- While generative AI can write simple programs, the responsibility to verify the correctness of the programs still falls on the user. So, even for verification purposes only, computer science students

still need to learn how to write programs. In order to provide appropriate prompts to generative AI, students need to be able to design and plan larger programs. How students design and write programs will evolve as generative AI technologies improve. How computer science educators must adapt to teaching programming with the help of generative AI is currently an open question. It seems likely that reading programs will see a sharpened focus and the idea of learning to program by writing, and reliance on learning syntax, will change towards learning to program by prompting, comprehending, verifying, editing, modifying, adapting, and testing code. It is also likely that other aspects of the curriculum such as problem decomposition will see increased focus.

Software Engineering (SE)

- Generative AI is expected to have substantial impact on several aspects of the software process, including (but not limited to) development of new code, comprehension of complex logging and debugging artifacts, static analysis, and code reviewing. The most understandable and visible change is likely to be in the development of (rote) new code—assistance technologies like GitHub's Copilot and other advanced auto-complete mechanisms can meaningfully improve development time, to a point. Effective use of such tools requires a deeper investment in design and code comprehension, while potentially decreasing the need for hands-on programming time. Similar advances in static analysis and code review are anticipated to have a meaningful impact on code quality and clarity, ideally also reducing the impact of implicit bias by increasing consistency and quality of comments and diagnostics.

Implications by Competency Area - Systems

Architecture and Organization (AR)

- The evolution needed in computer architecture to better support generative AI technologies may itself become a critical area of study in the future. Another might be the impact of using generative AI for hardware design.

Data Management (DM)

- Generative AI systems appear to excel at accomplishing intellectually challenging, yet mechanical tasks. It seems evident that given the expression of a query in pseudocode (or relational algebra/calculus), the translation of a description of the desired query result into SQL will become a routine generative AI task. In one sense, generative AI allows database querying at a higher (natural language, pseudocode) level. However, regardless of the level at which students query a database, they will still need the skills to validate the results returned by an (AI-generated) SQL query.

Networking and Communication (NC)

- Generative AI technologies may be increasingly used in computer science education of networking and communications. For example, it can generate code for new or existing networking protocols, suggestions for improvement, generate network traffic data for experimentation purposes, and

generate scenarios for which a network architecture needs to be designed. Furthermore, it can verify whether networking requirements are ambiguous or complete. It can be used to automatically generate configuration scripts, perform security assessments of existing networked configurations such as identifying vulnerabilities, and suggesting countermeasures, and perform reliability assessments and capacity planning.

Operating Systems (OS)

- Generative AI will be helpful with deployment scripts and system optimization. In the short term, generative AI will not invalidate the need for students to understand the layer between applications and the architecture. With that said, AI eventually will provide insight into the expected performance or security implications of software that is developer- or AI-generated. Students will need to continue to innovate and reason at higher-levels of abstraction.

Systems Fundamentals (SF)

- Providing system support for generative AI applications is expected to turn into a robust area of research and education.

Security (SEC)

- Generative AI can be used both by an adversary to launch more sophisticated attacks or develop more dangerous malware and by the defender to protect against such attacks. Other issues in the impact of generative AI on security include interaction between an AI attacker and defender, security risks of data and reverse engineering AI models, AI-based data aggregation / phishing, and security of AI systems themselves. Students need to develop a nuanced understanding of the power and drawbacks of generative AI applied to security concerns.

Implications by Competency Area - Applications

Artificial Intelligence (AI)

- The success of generative AI is already attracting more interest in Artificial Intelligence as a field, both increasing the number of students interested in studying it and increasing its applicability to other subfields of computer science. Generative models provide opportunities to incorporate multimodal analysis into larger pipelines and provide alternatives to techniques such as classical planning (e.g., using an LLM to suggest next steps to solving a problem). However, they currently lack foundational guarantees of correctness, grounded perception, and explanation that are critical to many applications. Emerging techniques for combining subsymbolic and symbolic methods using generative models look promising for resolving these issues.

Graphics and Interactive Techniques (GIT)

- Intellectual property and ethical issues regarding the use of generative AI for 2D and 3D graphics, images, video, and animation are important and evolving. These apply both to the media content and/or creative work used to train AI as well as the media created using generative AI.

Human-Computer Interaction (HCI)

- Generative AI will have a broad impact on the ways that people regard, and therefore interact, with computers and their output. Programs that employ generative AI can offer a simple interface that responds to user prompts and rapidly produce acceptable output. Such products, however, require user education to set reasonable expectations and acknowledge generative AI's ability to err and to hallucinate. Users must also know how to engineer a prompt to achieve the desired effect. Ideally, a product that employs generative AI should have safeguards that evolve with their use and be adaptive for accuracy and sensitivity to the user's cultural norms. Finally, users should become aware of issues surrounding intellectual property, transparency, and the extensive human labor required to train these models. The speed at which this technology continues to evolve requires careful attention to both current and future potential pitfalls, failure modes, and human consequences.

Specialized Platform Development (SPD)

- From an educational standpoint, generative AI tools offer valuable assistance to both students and developers. They enable the rapid creation of prototypes and the generation of code templates for common mobile app elements. Moreover, these tools produce code snippets for routine tasks, reducing the necessity for manual coding. This functionality is particularly advantageous for novices and those in the initial stages of learning mobile development. Generative AI Integrated Development Environment (IDE) plugins also deliver real-time feedback and suggestions to students as they write code.

  Furthermore, generative AI enriches teaching by crafting interactive simulations and virtual environments tailored to mobile app development. These environments allow students to experiment and hone their skills effectively. In the educational context, generative AI is crucial in raising awareness of security and ethical considerations within mobile app development. Additionally, AI serves as an invaluable resource for developers by providing continuous updates on the latest trends and technologies in specialized platform development, ensuring developers maintain up-to-date skills and knowledge in this rapidly evolving field.

Implications for Crosscutting Areas

Mathematical and Statistical Foundations (MSF)

- Mathematics is learned best by "doing" with sustained practice in working through problems. Although generative AI tools may help in explaining, they are just as likely to present complete solutions to students, robbing them of the learning gains that occur in struggling through problems. In contrast, generative AI tools could be adapted to help instructors automate grading of proofs that would alleviate what is now a laborious task. In the longer term, a balanced approach could offer educator support (automated grading) and student support through customized tutoring to address skill gaps either before starting a new course or while taking a course as long as the during-course usage is calibrated carefully. Perhaps the best option at the moment is to encourage rigorous on-

going experimentation with generative AI so that the academic computing community can identify best practices for the future.

Society, Ethics, and the Profession (SEP)

- The education computer science students receive needs to incorporate all dimensions and roles of the computing profession: technical, philosophical, and ethical. Generative AI raises unique risks for SEP, for instance deepfakes and misinformation will become more pervasive and harder to identify. Being technically capable of ethically questionable action and being technically able to identify and prevent such actions puts greater burden on SEP lessons to make students aware of these responsibilities and ensure that as they enter the workforce as graduates, they work to keep society on a just path.

Implications for the Curriculum

- It is clear that students must learn how to correctly use generative AI technologies for coursework. The boundary between using generative AI as a resource and using it to plagiarize must be clarified. The limitations (e.g., hallucinations) of the technology must be discussed, as should the inherent biases that may have been baked into the technology by virtue of the data used to train them.
- Many new technologies have already redefined the boundary between tasks that can be mechanized and those that will need human participation. Generative AI is no different. Correctly identifying the boundary will be the challenge for computer science educators going forward.
- Generative AI may be used to facilitate undergraduate research – more innovative research by more students at all levels of technical ability. Generative AI may be used by students for various research-related tasks: to fill the gaps in their understanding of prior research, build systems with which to test hypotheses, help interpret the results, etc.
- Students may use generative AI to summarize assigned readings, help explain gaps in their understanding of course material, fill in gaps in the presentations of the classes they missed, and interactively quiz themselves to assist in their studying.