Chat Overflow: Artificially Intelligent Models for Computing Education – renAIssance or apocAlypse?

Paul Denny∗
University of Auckland
Auckland, New Zealand
paul@cs.auckland.ac.nz

Brett A. Becker∗
University College Dublin
Dublin, Ireland
brett.becker@ucd.ie

Juho Leinonen∗
University of Auckland
Auckland, New Zealand
juho.leinonen@auckland.ac.nz

James Prather∗
Abilene Christian University
Abilene, Texas, USA
james.prather@acu.edu

ABSTRACT
Recent breakthroughs in deep learning have led to the emergence of generative AI models that exhibit extraordinary performance at producing human-like outputs. Using only simple input prompts, it is possible to generate novel text, images, video, music, and source code, as well as tackle tasks such as answering questions and translating and summarising text.

However, the potential for these models to impact computing education practice is only just beginning to be explored. For example, novices learning to code can now use free tools that automatically suggest solutions to programming exercises and assignments; yet these tools were not designed with novices in mind and little to nothing is known about how they will impact learning. Furthermore, much attention has focused on the immediate challenges these models present, such as academic integrity concerns. It seems that even in the AI-era a pending apocalypse sells better than a promising renaissance.

Generative AI will likely play an increasing role in people’s lives in the reasonably foreseeable future. Model performance seems set to continue accelerating while novel uses and new possibilities multiply. Given this, we should devote just as much effort to identifying and exploiting new opportunities as we do to identifying and mitigating challenges.

In this talk, we begin by discussing several concrete and research-backed opportunities for computing educators. Many of these have already shown great promise in positively impacting current practice. We then discuss more short- to medium-term possibilities in areas such as student recruitment, and curricular changes. Finally – against our better judgement – we speculate over the longer-term, including rethinking the very fundamentals of the practice of teaching introductory and advanced computing courses. In these discussions we suggest potential research questions and directions. Although making remotely accurate predictions in such a fast-changing landscape is foolhardy, we believe that now is the time to explore and embrace opportunities to help make positive change in as many computing classrooms as possible.

CCS CONCEPTS
• Social and professional topics → CS1: Computer science education; Software engineering education; • Computing methodologies → Neural networks; Machine learning; Artificial intelligence.

KEYWORDS
AI; artificial intelligence; ChatGPT; computer programming; computer science education; computing education; Copilot; deep learning; generative AI; large language models; LLM; machine learning

ACM Reference Format:

1 SPEAKERS
1.1 Paul Denny
Paul Denny enjoys exploring how computing students engage with online learning tools [4, 9, 10], and is particularly interested in how their experience can be impacted through user interface design and tool feedback [5, 6]. His fascination with large language models began in August 2021, after seeing the extraordinary performance of Codex following its release in private beta.

1.2 Brett Becker
Brett Becker is interested in how humans learn to program and how they perceive this process [1]. He is fascinated by the interaction between humans and computers, exemplified by his obsession with programming error messages [3] and what AI has to do with them [13]. He is far from alone in his belief that that generative AI will dramatically change the way programming is taught and learned [2] and is keen to try to keep up with the seemingly non-stop acceleration of the capabilities of AI. He is not sure if he is surprised or not that LLMs have offered yet another parallel between programming and natural languages, in that LLMs have demonstrated similar capabilities in both domains through very similar mechanisms.

1.3 Juho Leinonen
Juho Leinonen explores how to best support and engage diverse learner populations with educational technology and artificial intelligence. Recently, he has researched the potential opportunities that large language models could provide for introductory programming instructors such as automatically creating personalised exercises [8, 10], enhancing programming error messages with LLMs [13], and creating code explanations for students using LLMs [12, 14].

1.4 James Prather
James Prather is very interested in how novices learn to code. His research has examined novice programmer interaction with compiler error messages [3, 7] and novice programmer metacognition and self-regulation [15, 16]. Recently he has worked on multiple papers on the impact of LLMs on introductory computing education [2, 11, 13, 17].

REFERENCES