GENERATIVE AI AND COMPUTING EDUCATION

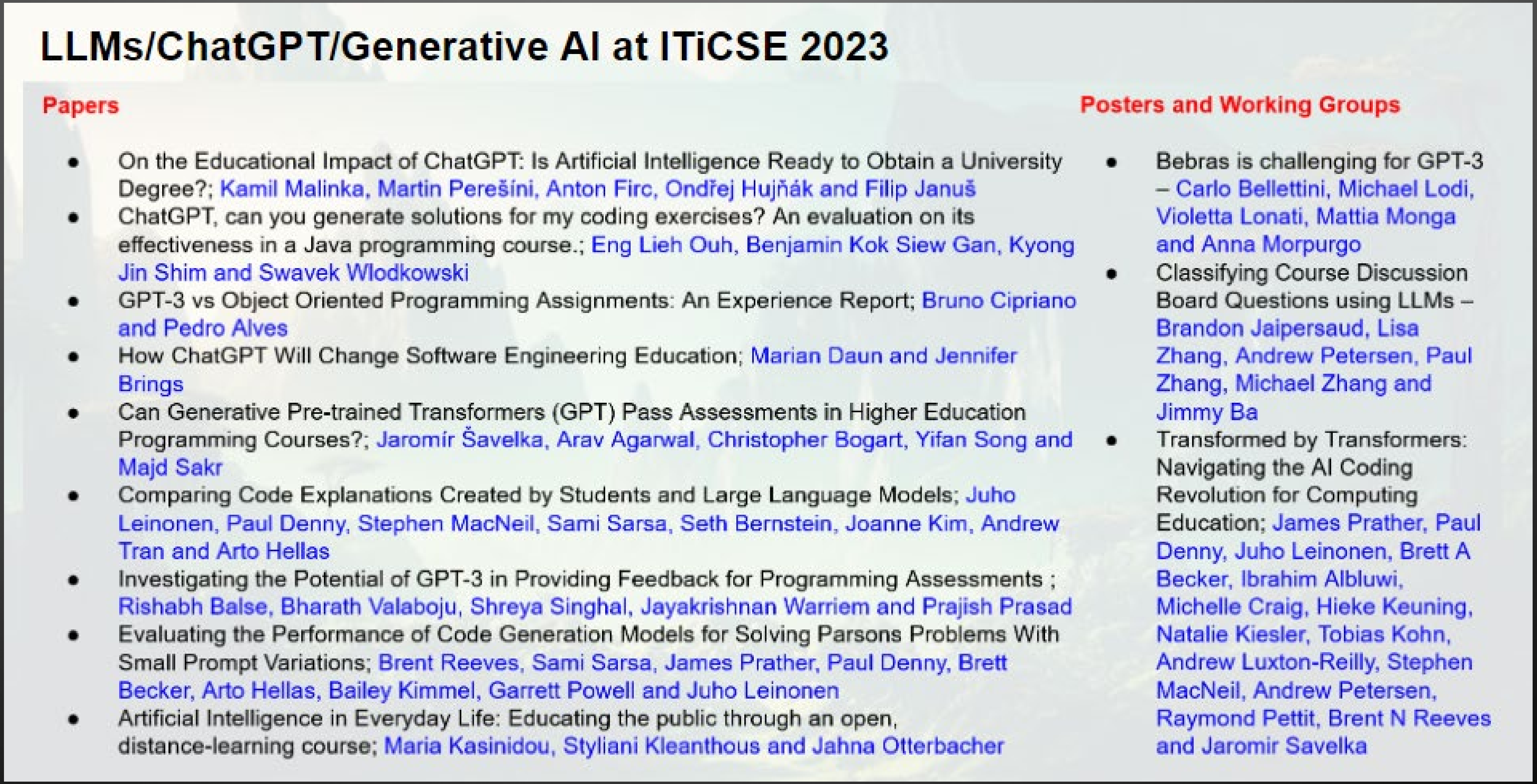
A PERSPECTIVE FROM SEPTEMBER 28, 2030 - LIKELY TO BE OUT OF DATE BY SEPTEMBER 29, 2023

BRETT A. BECKER

UNIVERSITY COLLEGE DUBLIN

Number of papers on Generative AI at ITiCSE 2022 (Dublin):

0



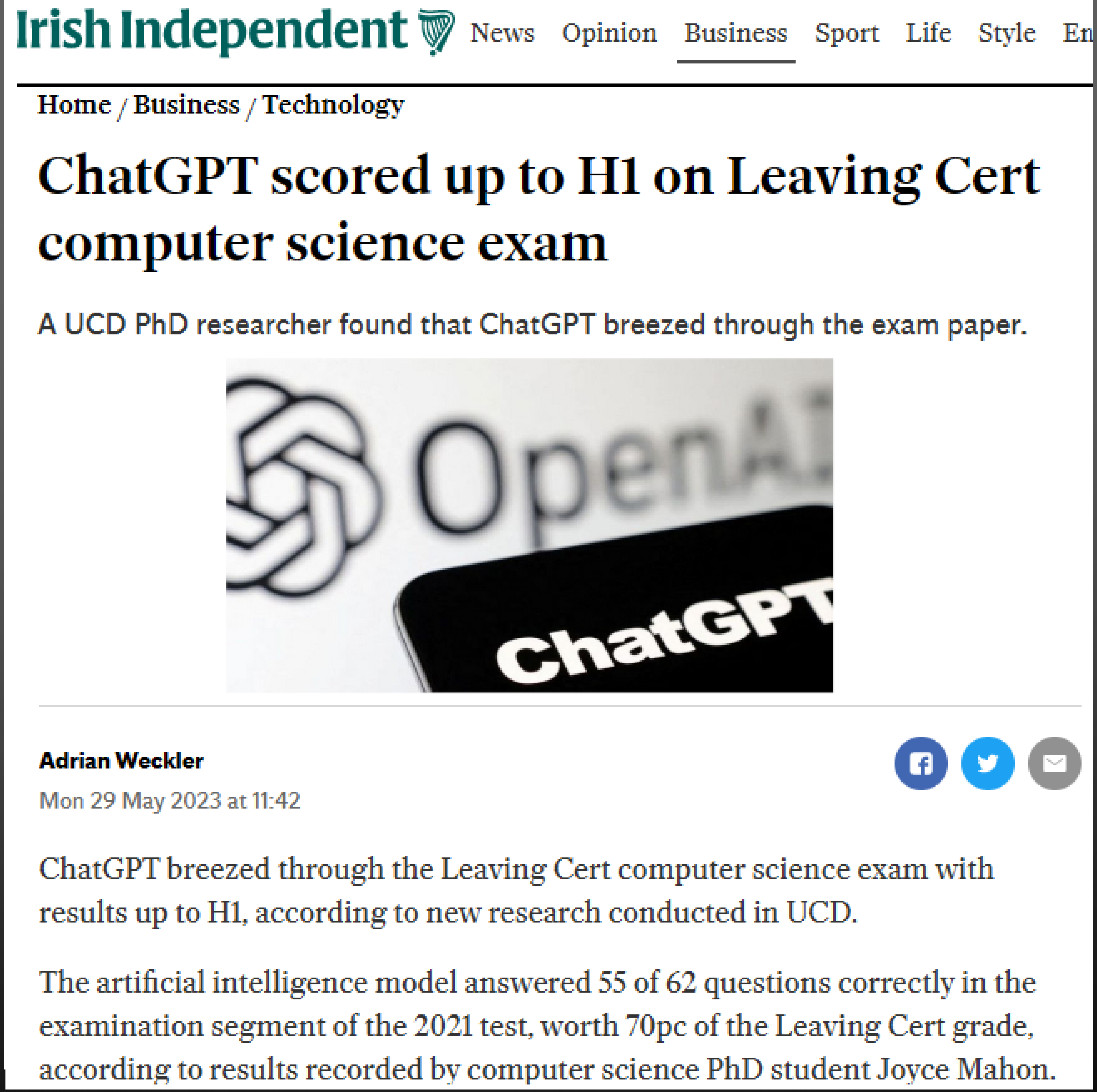
LLMs/ChatGPT/Generative AI at ITiCSE 2023

Papers

- On the Educational Impact of ChatGPT: Is Artificial Intelligence Ready to Obtain a University Degree?; Kamil Malinka, Martin Perešíni, Anton Firc, Ondřej Hujňák and Filip Januš
- ChatGPT, can you generate solutions for my coding exercises? An evaluation on its effectiveness in a Java programming course.; Eng Lieh Ouh, Benjamin Kok Siew Gan, Kyong Jin Shim and Swavek Wlodkowski
- GPT-3 vs Object Oriented Programming Assignments: An Experience Report; Bruno Cipriano and Pedro Alves
- How ChatGPT Will Change Software Engineering Education; Marian Daun and Jennifer Brings
- Can Generative Pre-trained Transformers (GPT) Pass Assessments in Higher Education Programming Courses?; Jaromír Šavelka, Arav Agarwal, Christopher Bogart, Yifan Song and Majd Sakr
- Comparing Code Explanations Created by Students and Large Language Models; Juho Leinonen, Paul Denny, Stephen MacNeil, Sami Sarsa, Seth Bernstein, Joanne Kim, Andrew Tran and Arto Hellas
- Investigating the Potential of GPT-3 in Providing Feedback for Programming Assessments ; Rishabh Balse, Bharath Valaboju, Shreya Singhal, Jayakrishnan Warriem and Prajish Prasad
- Evaluating the Performance of Code Generation Models for Solving Parsons Problems With Small Prompt Variations; Brent Reeves, Sami Sarsa, James Prather, Paul Denny, Brett Becker, Arto Hellas, Bailey Kimmel, Garrett Powell and Juho Leinonen
- Artificial Intelligence in Everyday Life: Educating the public through an open, distance-learning course; Maria Kasinidou, Styliani Kleanthous and Jahna Otterbacher

Posters and Working Groups

- Bebras is challenging for GPT-3 – Carlo Bellettini, Michael Lodi, Violetta Lonati, Mattia Monga and Anna Morpurgo
- Classifying Course Discussion Board Questions using LLMs – Brandon Jaipersaud, Lisa Zhang, Andrew Petersen, Paul Zhang, Michael Zhang and Jimmy Ba
- Transformed by Transformers: Navigating the AI Coding Revolution for Computing Education; James Prather, Paul Denny, Juho Leinonen, Brett A Becker, Ibrahim Albluwi, Michelle Craig, Hieke Keuning, Natalie Kiesler, Tobias Kohn, Andrew Luxton-Reilly, Stephen MacNeil, Andrew Petersen, Raymond Pettit, Brent N Reeves and Jaromir Savelka



Irish Independent

Home / Business / Technology

ChatGPT scored up to H1 on Leaving Cert computer science exam

A UCD PhD researcher found that ChatGPT breezed through the exam paper.

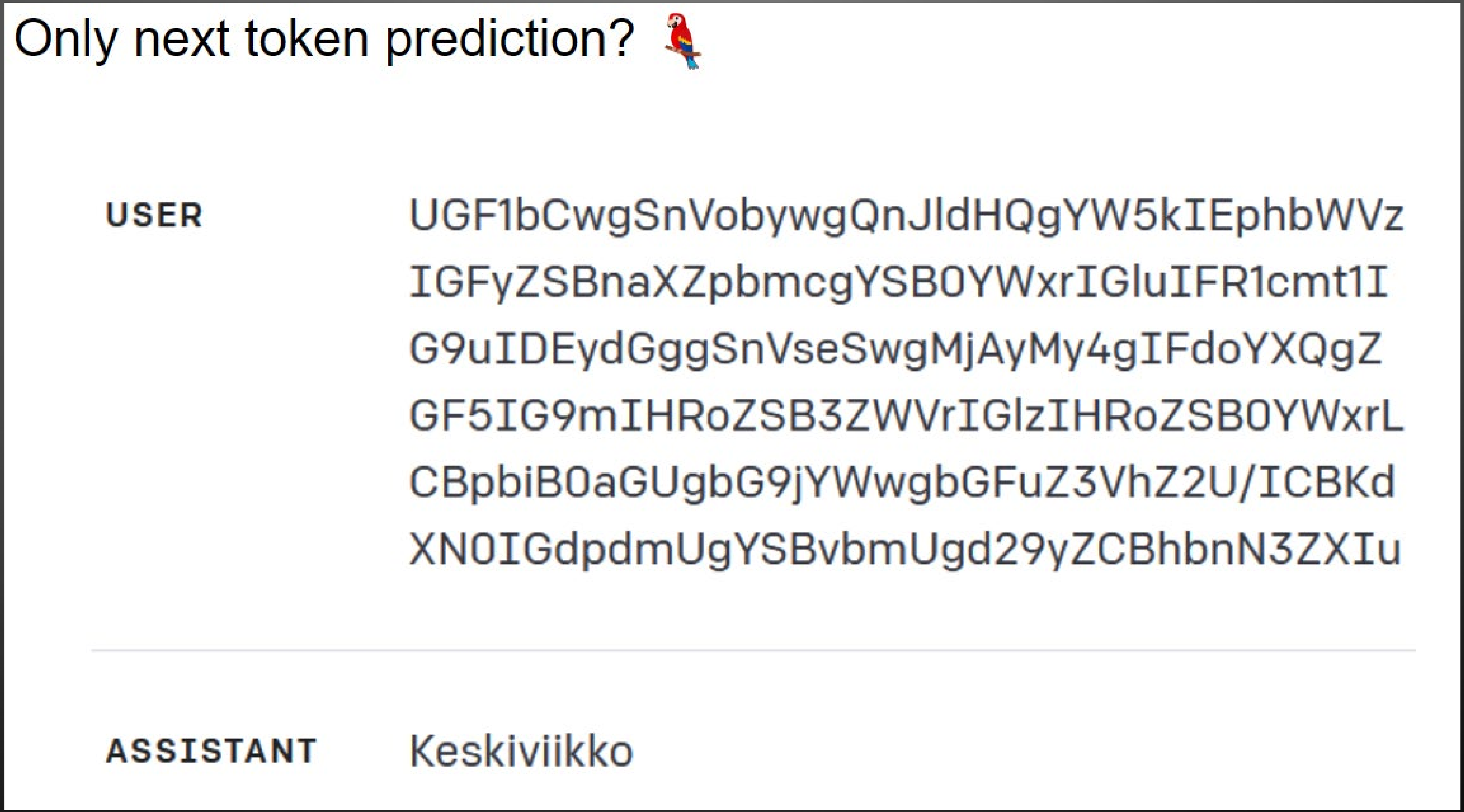
Adrian Weckler

Mon 29 May 2023 at 11:42

ChatGPT breezed through the Leaving Cert computer science exam with results up to H1, according to new research conducted in UCD.

The artificial intelligence model answered 55 of 62 questions correctly in the examination segment of the 2021 test, worth 70pc of the Leaving Cert grade, according to results recorded by computer science PhD student Joyce Mahon.

[1]



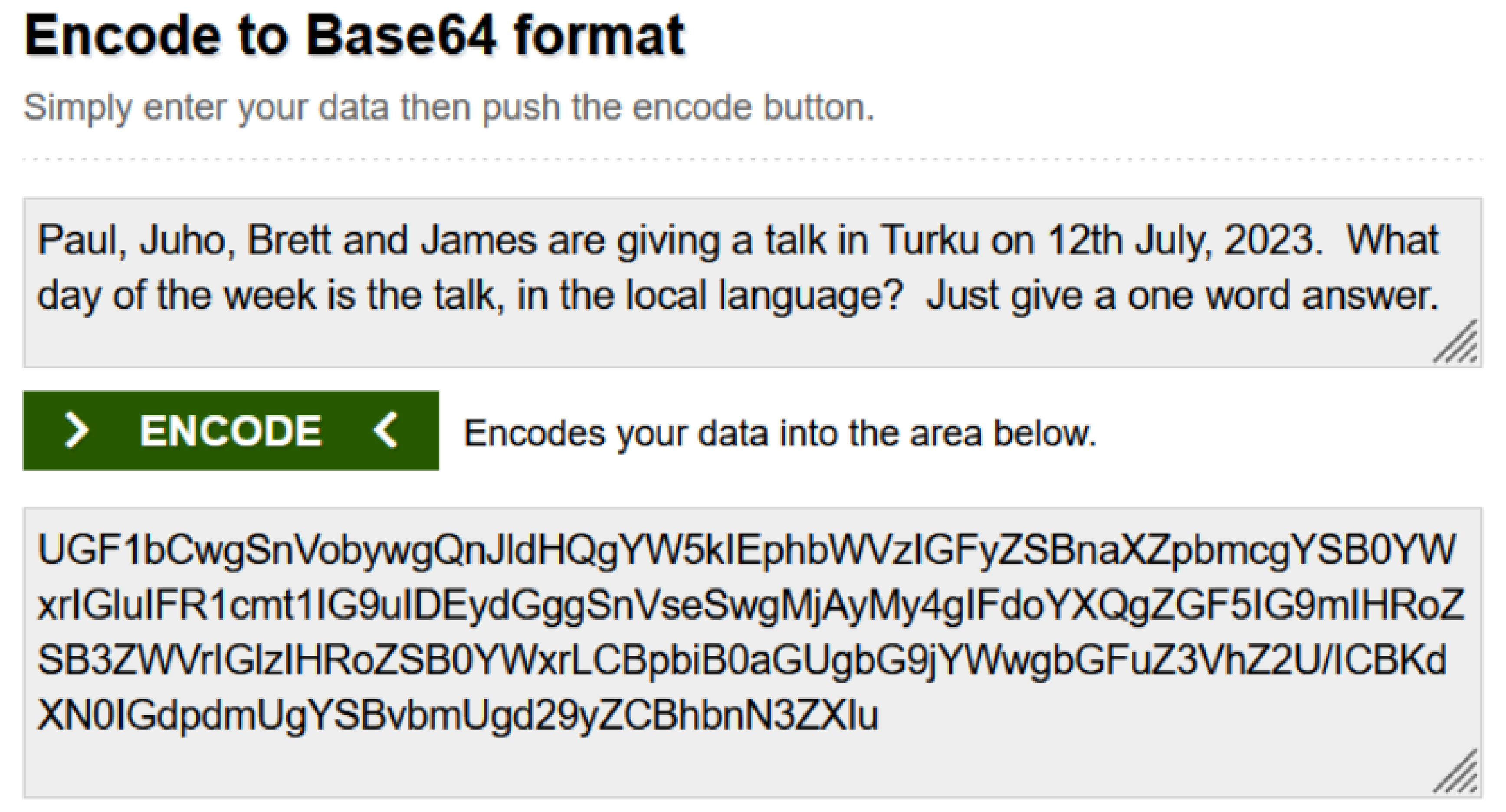
Only next token prediction? 🦜

USER

```
UGF1bCwgSnVobywgQnJldHQgYW5kIEphbWVz
IGFyZSBnaXZpbmcgYSB0YWxrIGluIFR1cmt1I
G9uIDEydGggSnVseSwgMjAyMy4gIFdoYXQgZ
GF5IG9mIHRoZSB3ZWVrIGlzIHRoZSB0YWxrL
CBpbiB0aGUgbG9jYWwgbGFuZ3VhZ2U/ICBKd
XN0IGdpdmUgYSBvbmUgd29yZCBhbnN3ZXIu
```

ASSISTANT Keskiviikko

[2]



Encode to Base64 format

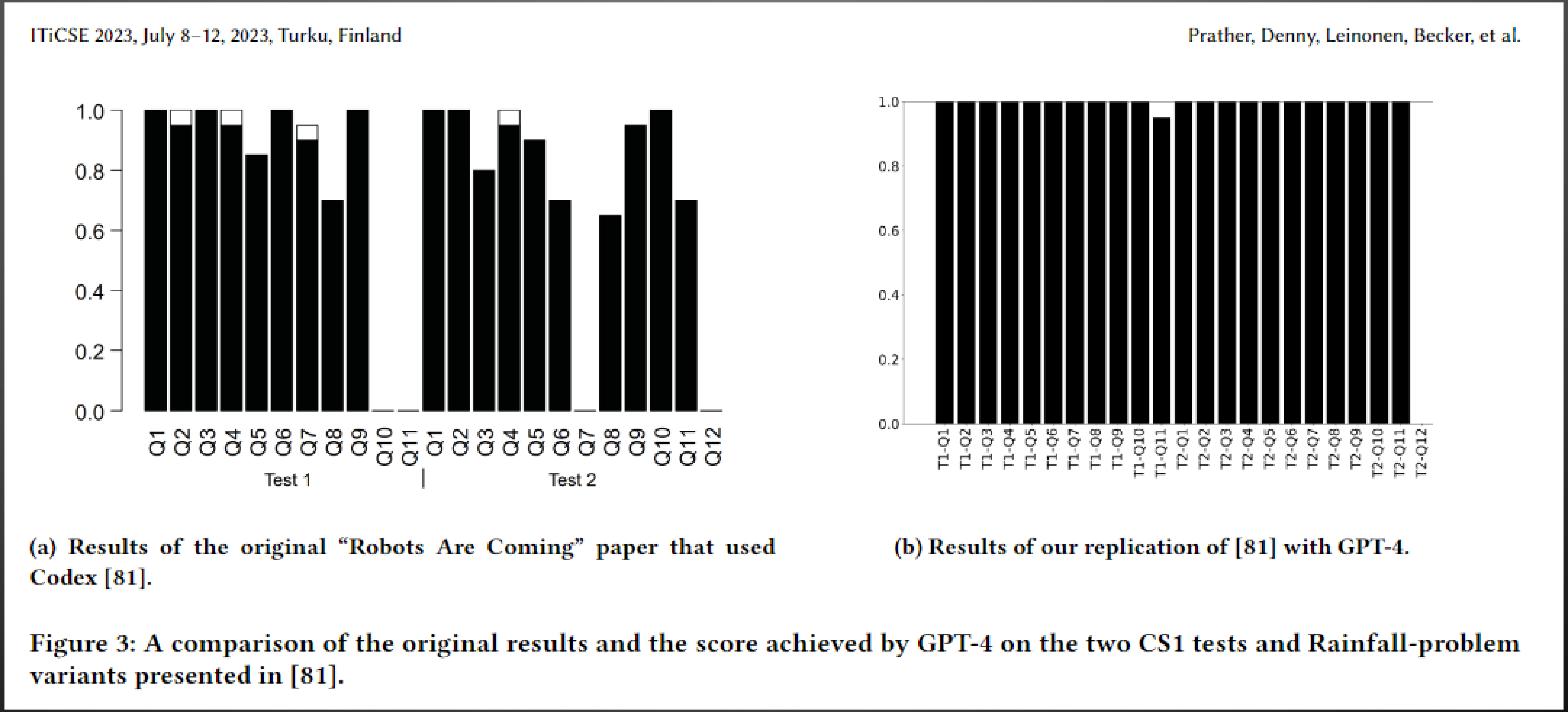
Simply enter your data then push the encode button.

Paul, Juho, Brett and James are giving a talk in Turku on 12th July, 2023. What day of the week is the talk, in the local language? Just give a one word answer.

Encodes your data into the area below.

UGF1bCwgSnVobywgQnJldHQgYW5kIEphbWVzIGFyZSBnaXZpbmcgYSB0YW
xrIGluIFR1cmt1IG9uIDEydGggSnVseSwgMjAyMy4gIFdoYXQgZGF5IG9mIHRoZ
SB3ZWVrIGlzIHRoZSB0YWxrLCBpbiB0aGUgbG9jYWwgbGFuZ3VhZ2U/ICBKd
XN0IGdpdmUgYSBvbmUgd29yZCBhbnN3ZXIu

[2]



ITiCSE 2023, July 8–12, 2023, Turku, Finland

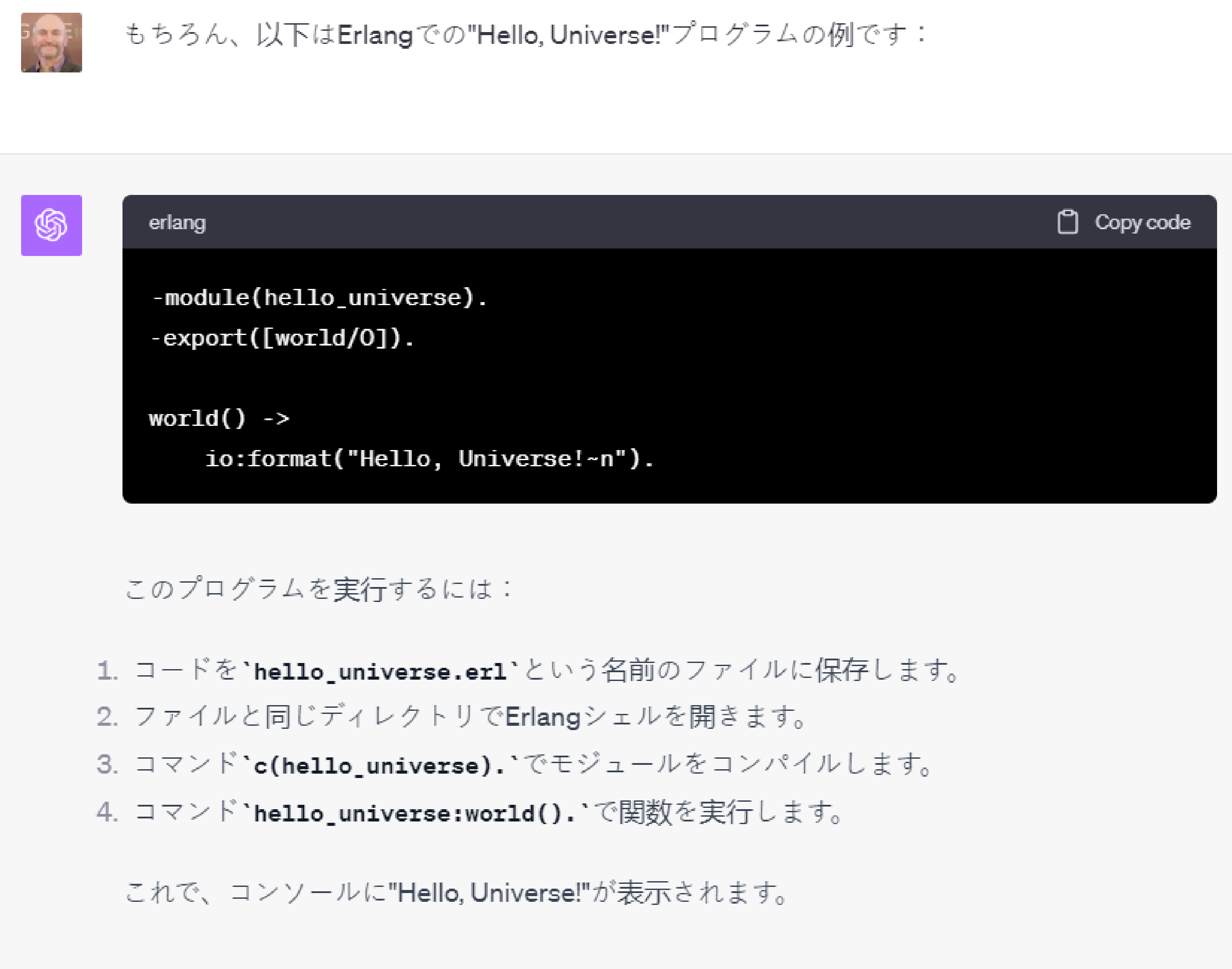
Prather, Denny, Leinonen, Becker, et al.

(a) Results of the original "Robots Are Coming" paper that used Codex [81].

(b) Results of our replication of [81] with GPT-4.

Figure 3: A comparison of the original results and the score achieved by GPT-4 on the two CS1 tests and Rainfall-problem variants presented in [81].

[2]



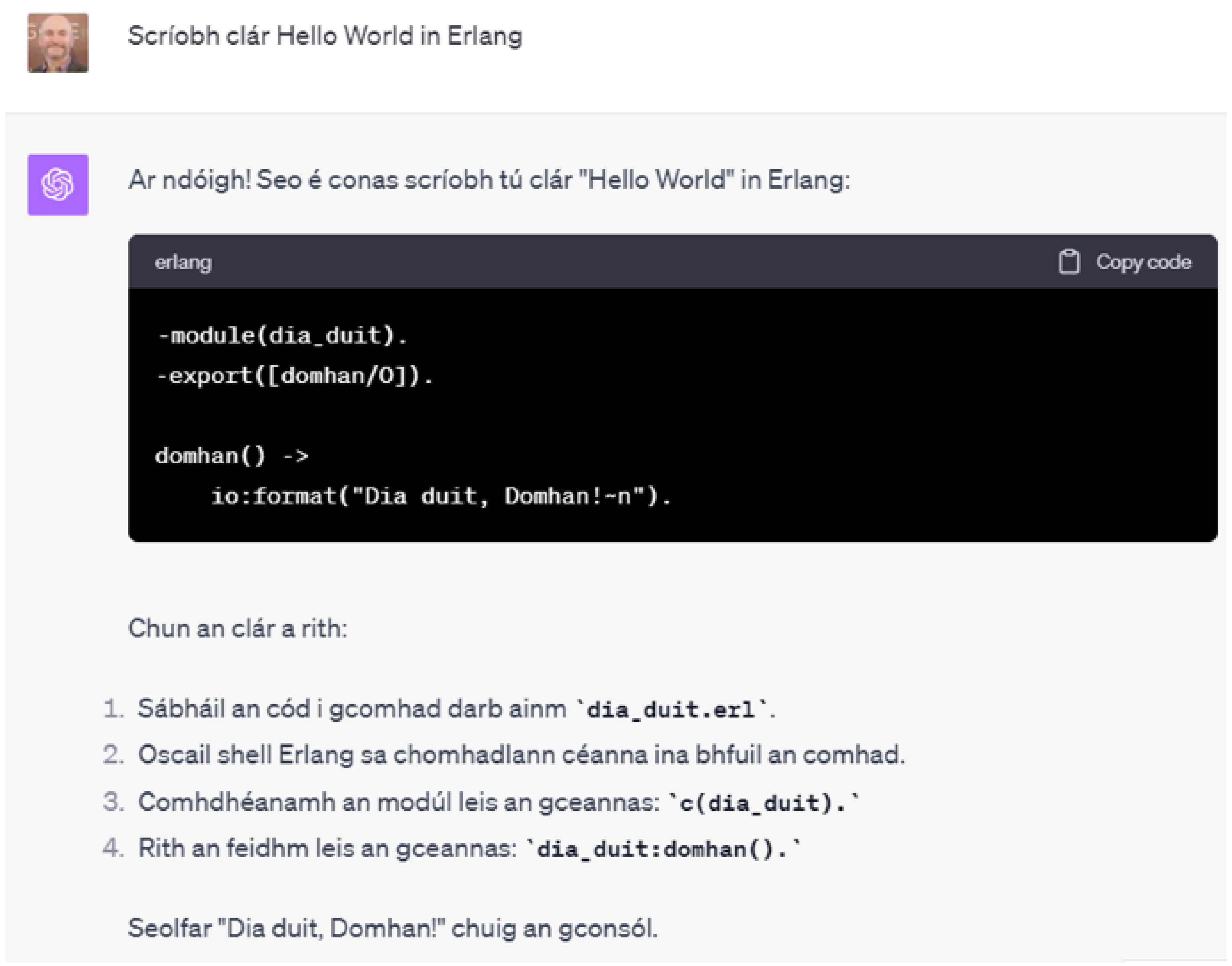
もちろん、以下はErlangでの"Hello, Universe!"プログラムの例です：

```erlang
-module(hello_universe).
-export([world/0]).

world() ->
    io:format("Hello, Universe!~n").
```

このプログラムを実行するには：

1. コードを`hello_universe.erl`という名前のファイルに保存します。
2. ファイルと同じディレクトリでErlangシェルを開きます。
3. コマンド`c(hello_universe).`でモジュールをコンパイルします。
4. コマンド`hello_universe:world().`で関数を実行します。

これで、コンソールに"Hello, Universe!"が表示されます。



Scríobh clár Hello World in Erlang

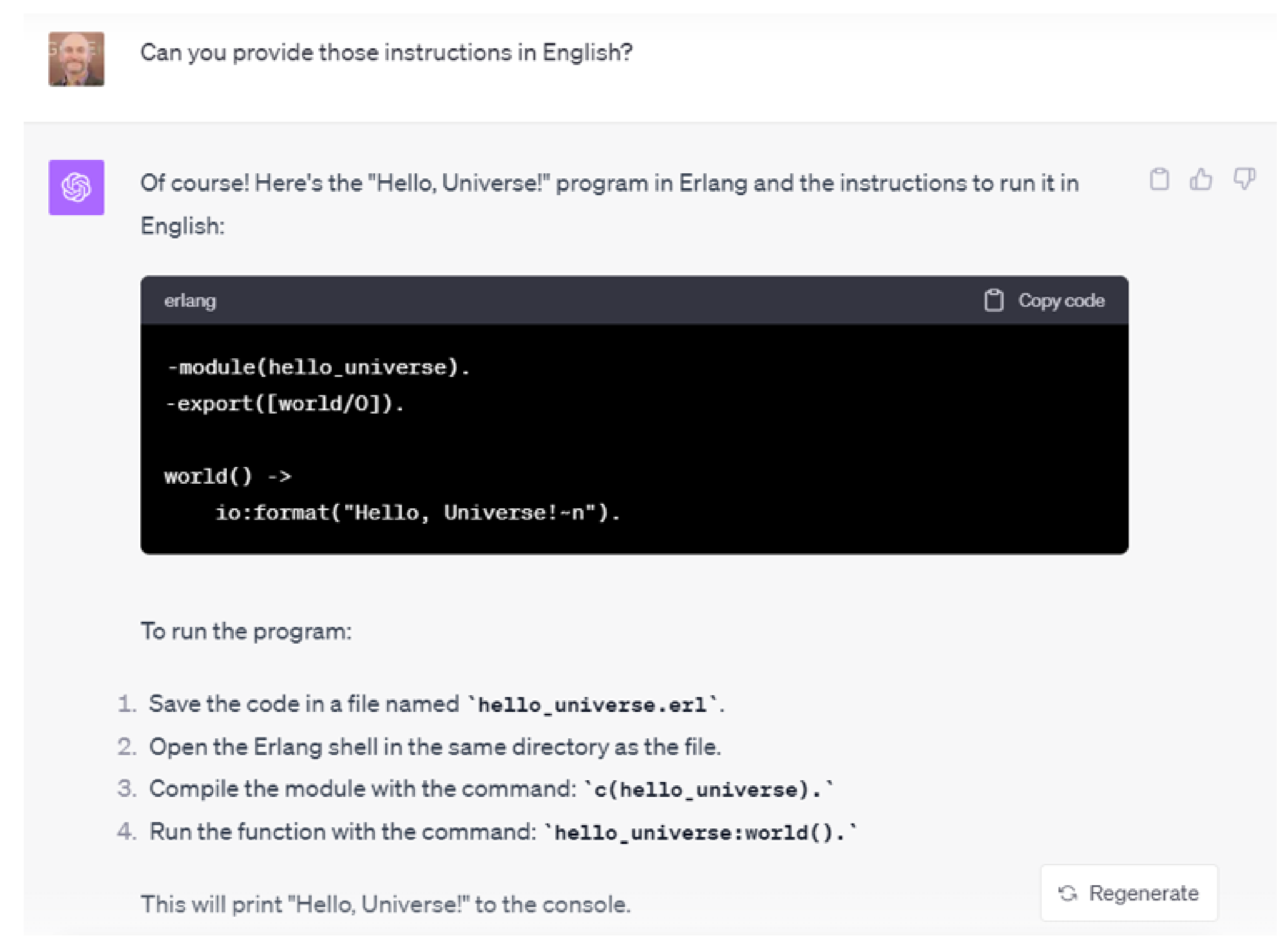
Ar ndóigh! Seo é conas scríobh tú clár "Hello World" in Erlang:

```erlang
-module(dia_duit).
-export([domhan/0]).

domhan() ->
    io:format("Dia duit, Domhan!~n").
```

Chun an clár a rith:

1. Sábháil an cód i gcomhad darb ainm `dia_duit.erl`.
2. Oscail shell Erlang sa chomhadlann céanna ina bhfuil an comhad.
3. Comhdhéanamh an modúl leis an gceannas: `c(dia_duit).`
4. Rith an feidhm leis an gceannas: `dia_duit:domhan().`

Seolfar "Dia duit, Domhan!" chuig an gconsól.



Can you provide those instructions in English?

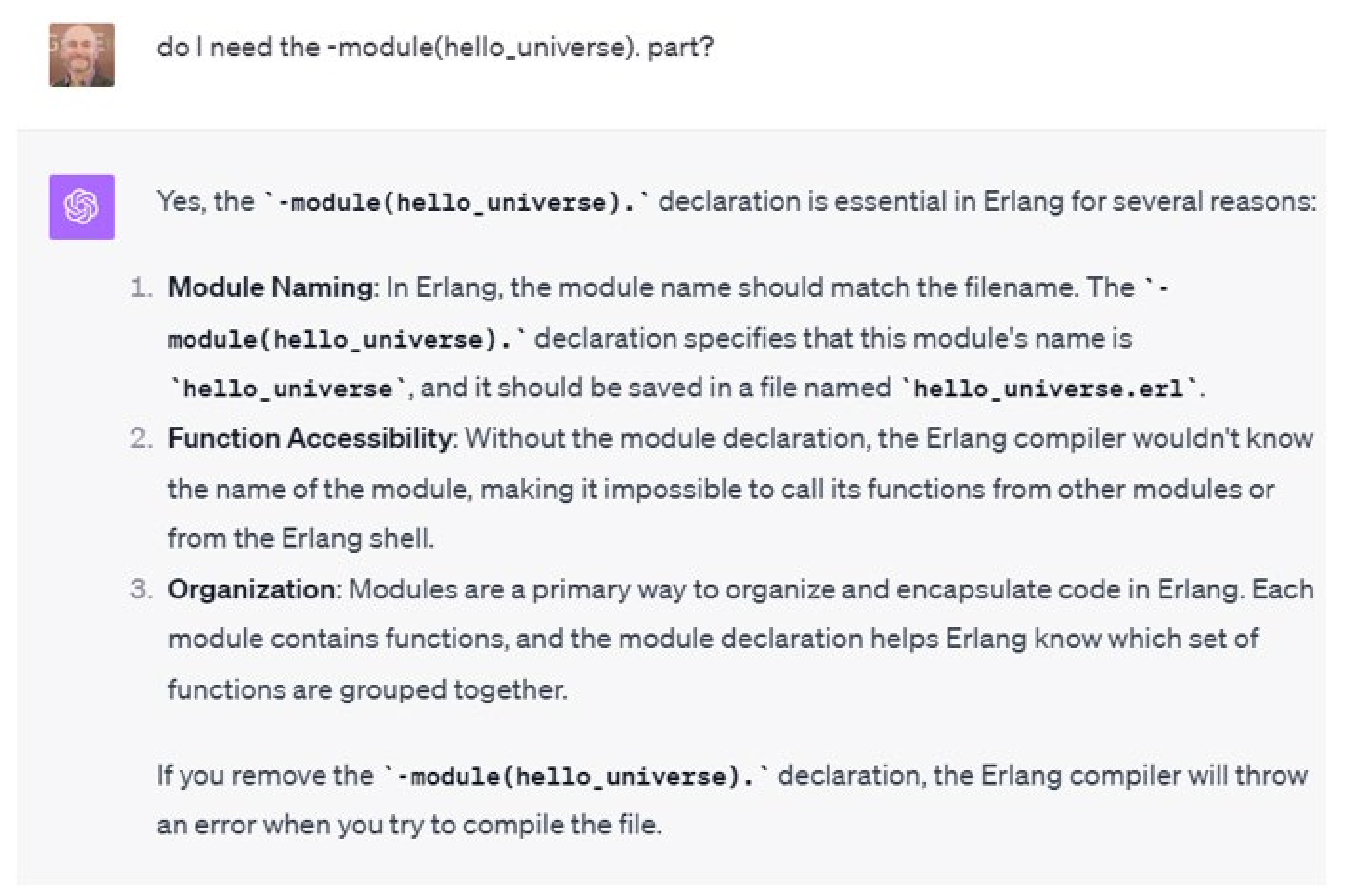
Of course! Here's the "Hello, Universe!" program in Erlang and the instructions to run it in English:

```erlang
-module(hello_universe).
-export([world/0]).

world() ->
    io:format("Hello, Universe!~n").
```

To run the program:

1. Save the code in a file named `` `hello_universe.erl` ``.
2. Open the Erlang shell in the same directory as the file.
3. Compile the module with the command: `` `c(hello_universe).` ``
4. Run the function with the command: `` `hello_universe:world().` ``

This will print "Hello, Universe!" to the console.

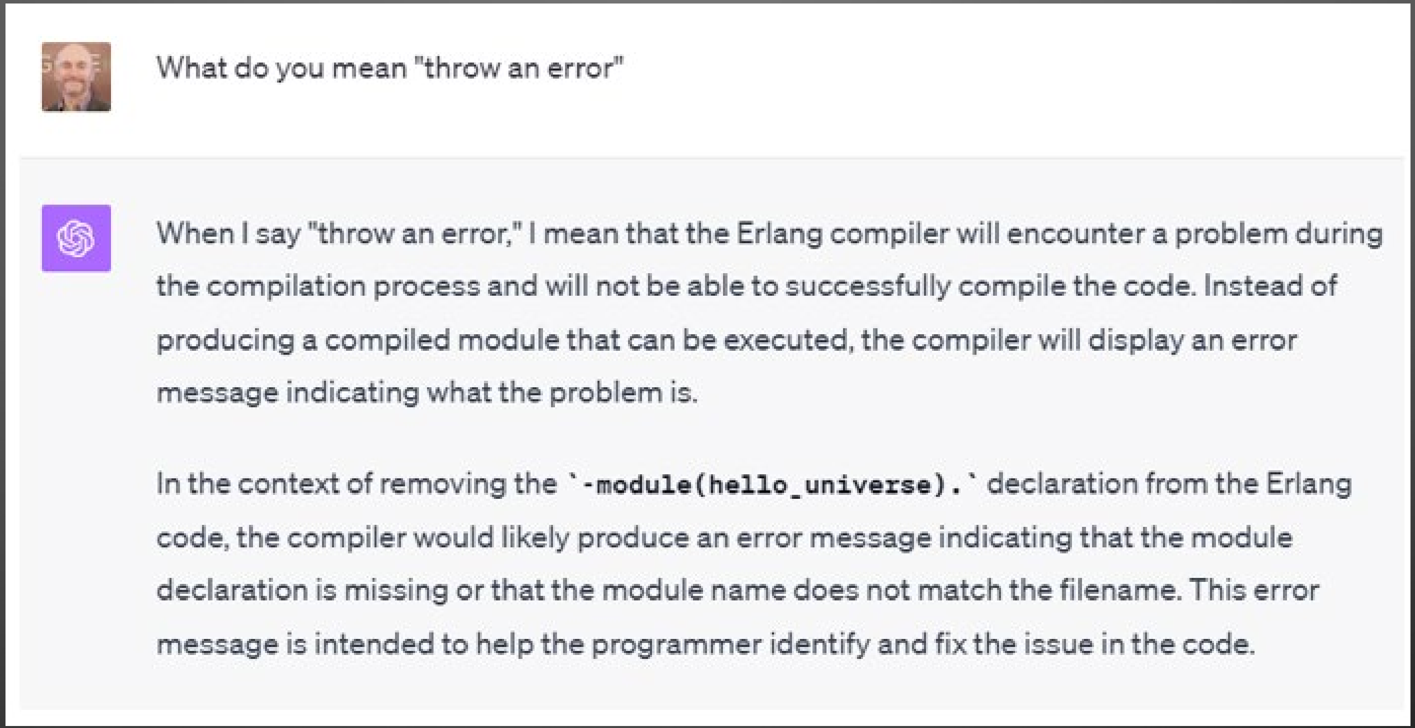


do I need the -module(hello_universe). part?

Yes, the `-module(hello_universe).` declaration is essential in Erlang for several reasons:

1. **Module Naming**: In Erlang, the module name should match the filename. The `-module(hello_universe).` declaration specifies that this module's name is `hello_universe`, and it should be saved in a file named `hello_universe.erl`.
2. **Function Accessibility**: Without the module declaration, the Erlang compiler wouldn't know the name of the module, making it impossible to call its functions from other modules or from the Erlang shell.
3. **Organization**: Modules are a primary way to organize and encapsulate code in Erlang. Each module contains functions, and the module declaration helps Erlang know which set of functions are grouped together.

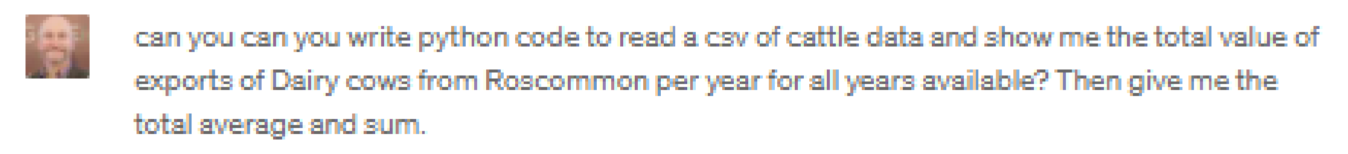
If you remove the `-module(hello_universe).` declaration, the Erlang compiler will throw an error when you try to compile the file.



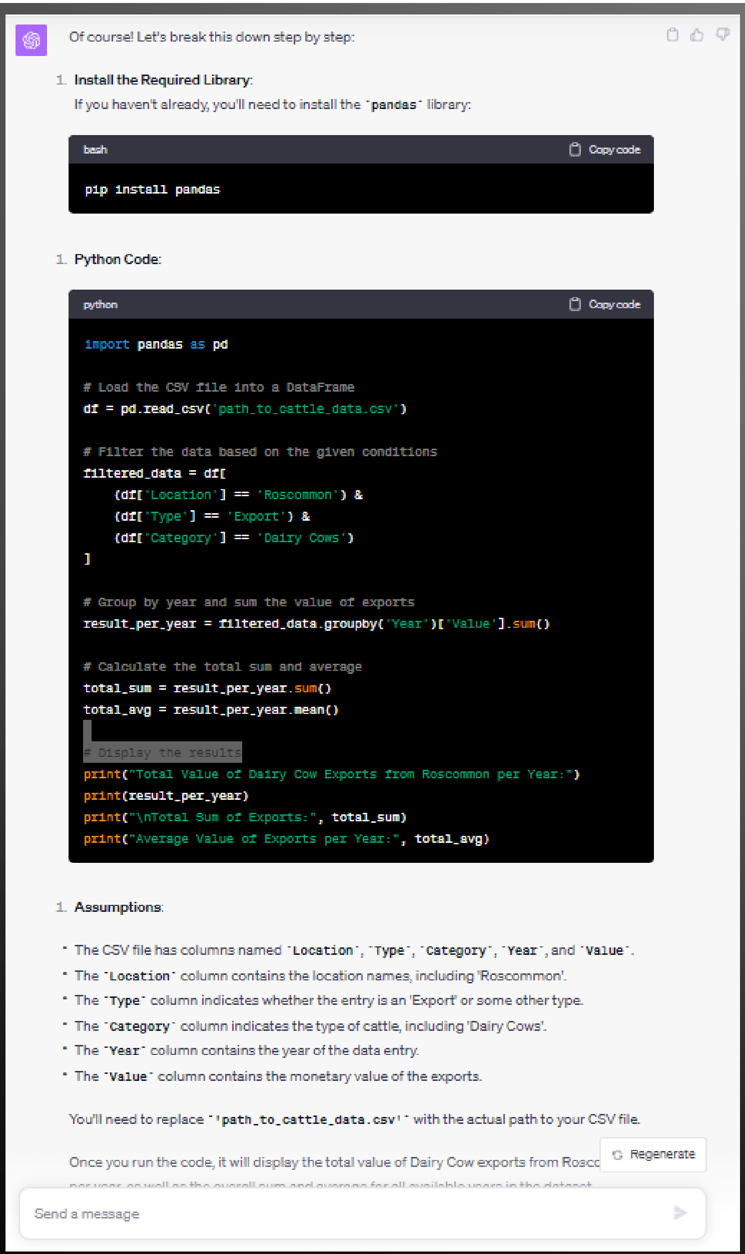
What do you mean "throw an error"

When I say "throw an error," I mean that the Erlang compiler will encounter a problem during the compilation process and will not be able to successfully compile the code. Instead of producing a compiled module that can be executed, the compiler will display an error message indicating what the problem is.

In the context of removing the `` `-module(hello_universe).` `` declaration from the Erlang code, the compiler would likely produce an error message indicating that the module declaration is missing or that the module name does not match the filename. This error message is intended to help the programmer identify and fix the issue in the code.



can you can you write python code to read a csv of cattle data and show me the total value of exports of Dairy cows from Roscommon per year for all years available? Then give me the total average and sum.



Of course! Let's break this down step by step:

1. **Install the Required Library**:

   If you haven't already, you'll need to install the `pandas` library:

```bash
pip install pandas
```

1. **Python Code**:

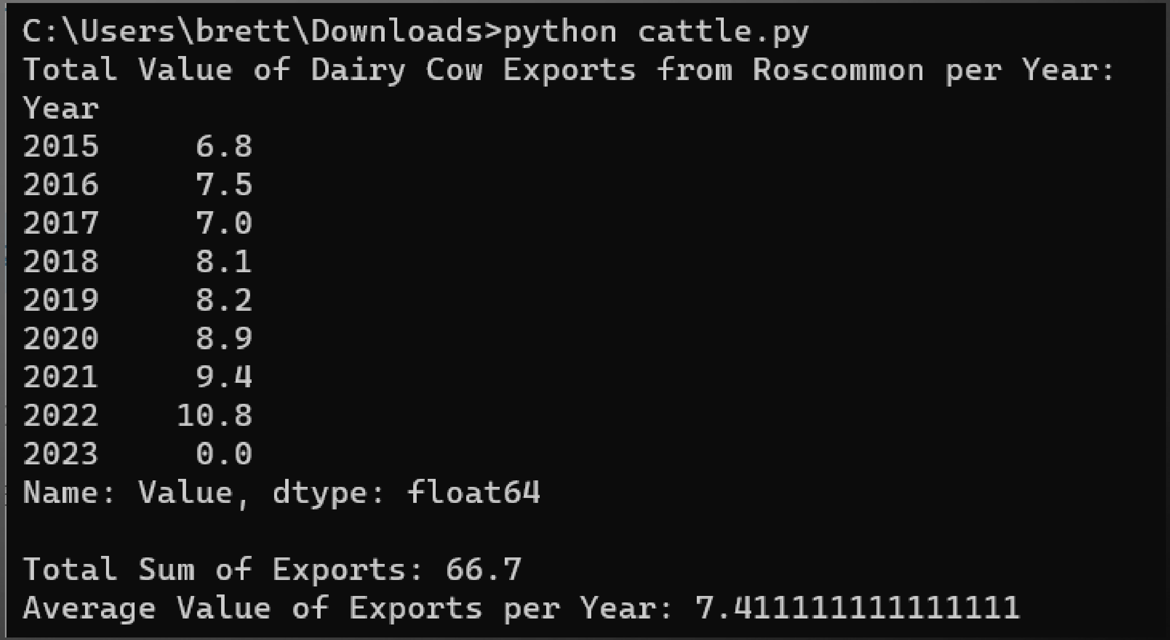
```python
import pandas as pd

# Load the CSV file into a DataFrame
df = pd.read_csv('path_to_cattle_data.csv')

# Filter the data based on the given conditions
filtered_data = df[
    (df['Location'] == 'Roscommon') &
    (df['Type'] == 'Export') &
    (df['Category'] == 'Dairy Cows')
]

# Group by year and sum the value of exports
result_per_year = filtered_data.groupby('Year')['Value'].sum()

# Calculate the total sum and average
total_sum = result_per_year.sum()
total_avg = result_per_year.mean()

# Display the results
print("Total Value of Dairy Cow Exports from Roscommon per Year:")
print(result_per_year)
print("\nTotal Sum of Exports:", total_sum)
print("Average Value of Exports per Year:", total_avg)
```

1. **Assumptions**:

- The CSV file has columns named `Location`, `Type`, `Category`, `Year`, and `Value`.
- The `Location` column contains the location names, including 'Roscommon'.
- The `Type` column indicates whether the entry is an 'Export' or some other type.
- The `Category` column indicates the type of cattle, including 'Dairy Cows'.
- The `Year` column contains the year of the data entry.
- The `Value` column contains the monetary value of the exports.

You'll need to replace `'path_to_cattle_data.csv'` with the actual path to your CSV file.

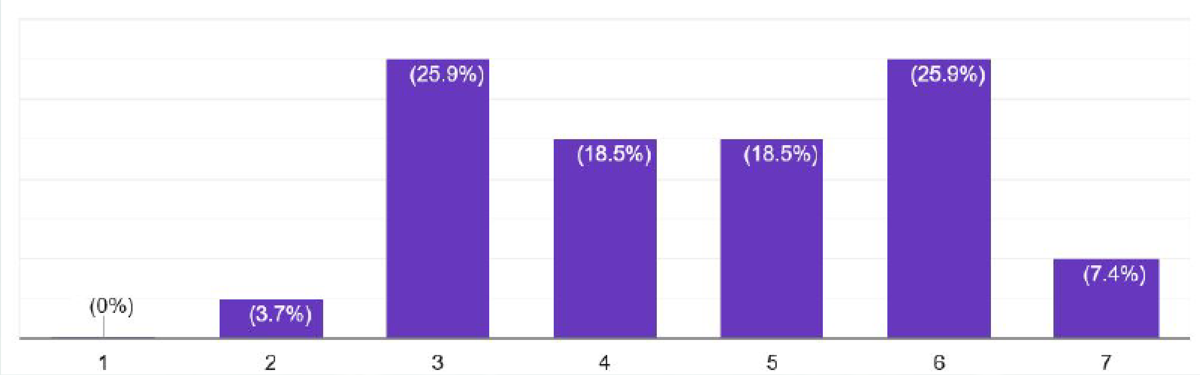
Once you run the code, it will display the total value of Dairy Cow exports from Rosc



```
C:\Users\brett\Downloads>python cattle.py
Total Value of Dairy Cow Exports from Roscommon per Year:
Year
2015     6.8
2016     7.5
2017     7.0
2018     8.1
2019     8.2
2020     8.9
2021     9.4
2022    10.8
2023     0.0
Name: Value, dtype: float64

Total Sum of Exports: 66.7
Average Value of Exports per Year: 7.411111111111111
```

How do you feel about large language models and generative AI in computing education?

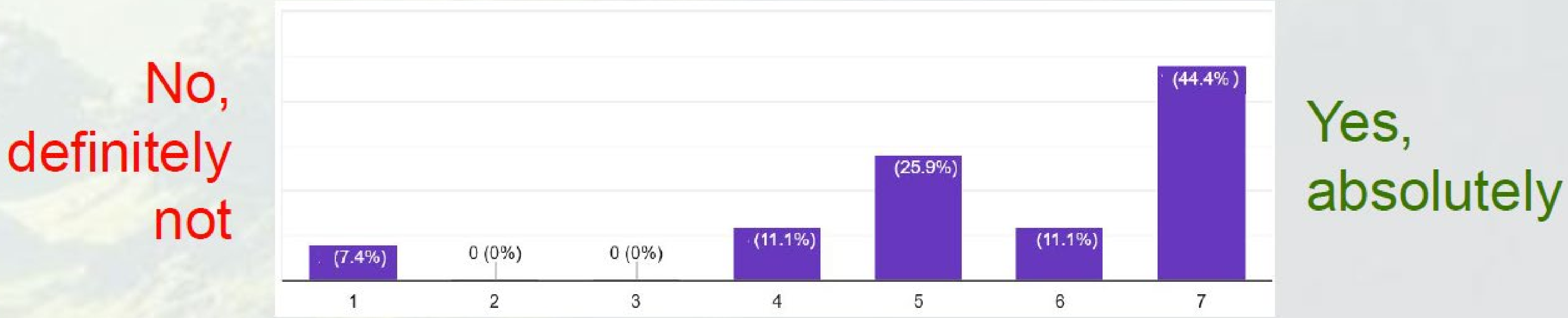
Negative, pessimistic, anxious, etc.



Positive, optimistic, excited, etc.

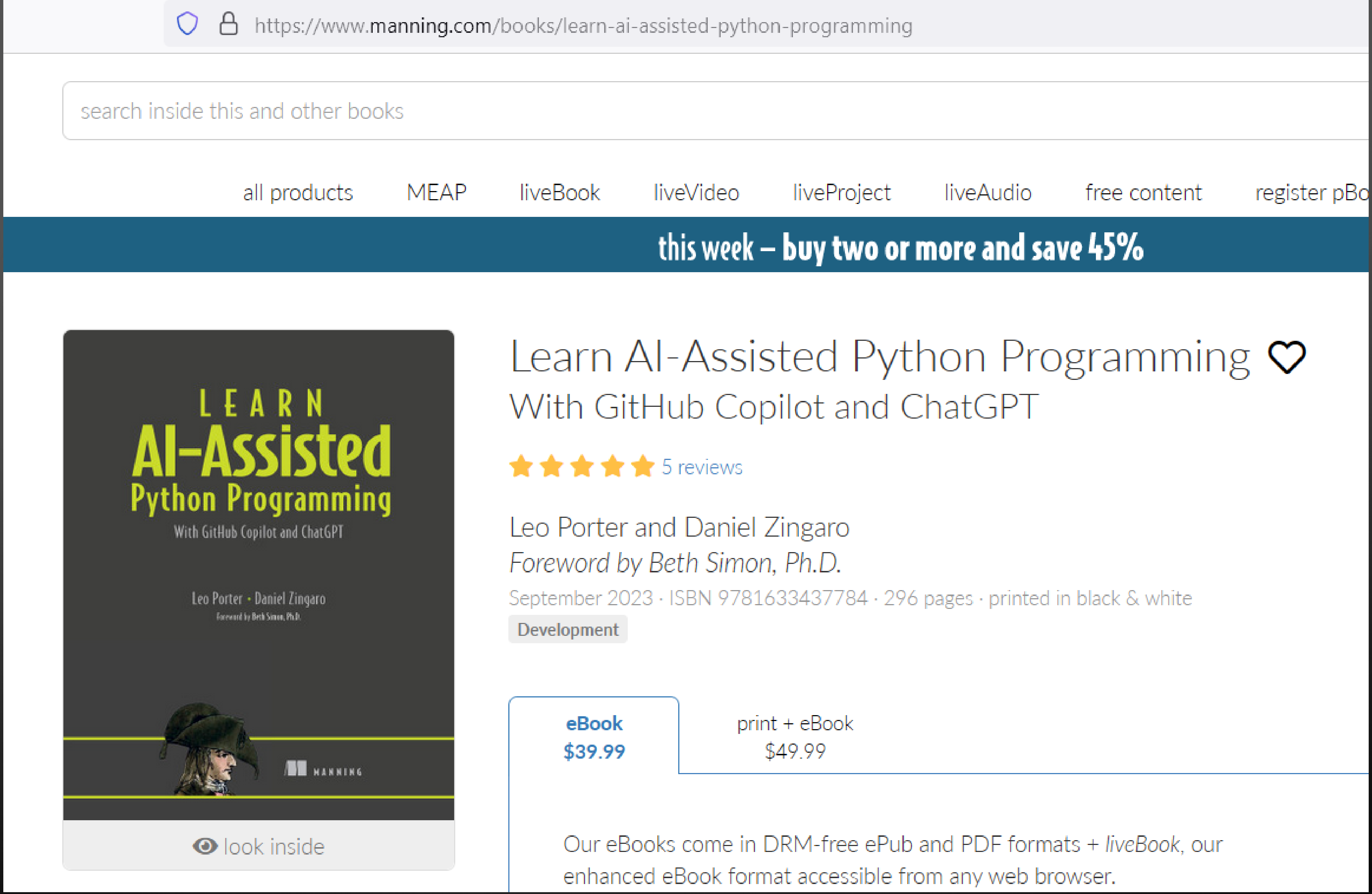
[2]

I plan to integrate / incorporate generative AI tools into my computing courses.



[2]

www.manning.com/books/learn-ai-assisted-python-programming


https://www.manning.com/books/learn-ai-assisted-python-programming
search inside this and other books
all products
MEAP
liveBook
liveVideo
liveProject
liveAudio
free content
register pBo
LEARN
AI-Assisted
Python Programming
With GitHub Copilot and ChatGPT
Leo Porter · Daniel Zingaro
Foreword by Beth Simon, Ph.D.
MANNING
look inside

this week – buy two or more and save 45%

Learn AI-Assisted Python Programming

With GitHub Copilot and ChatGPT

★★★★★ 5 reviews

Leo Porter and Daniel Zingaro

Foreword by Beth Simon, Ph.D.

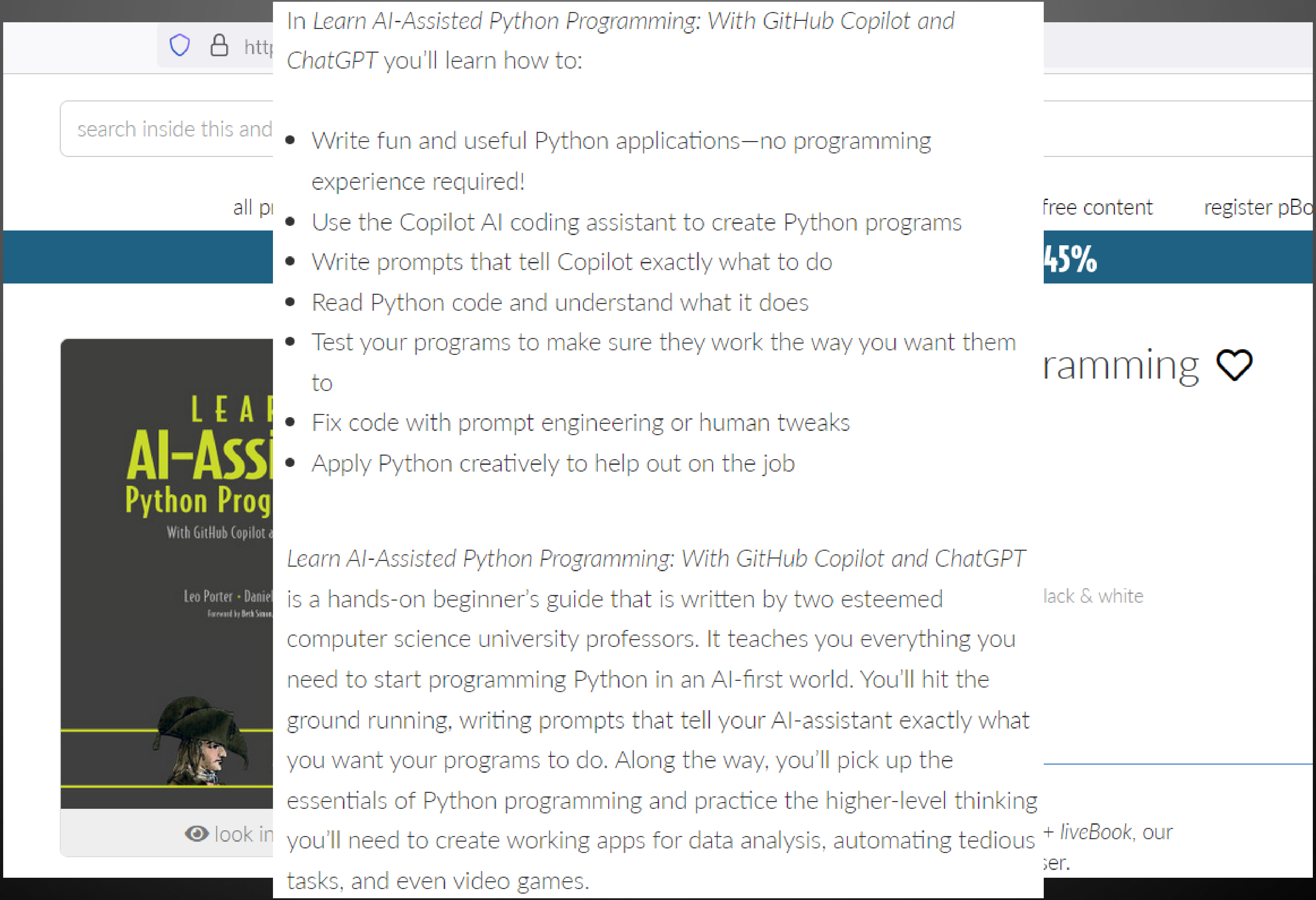
September 2023 · ISBN 9781633437784 · 296 pages · printed in black & white

Development

eBook **$39.99**	print + eBook $49.99

Our eBooks come in DRM-free ePub and PDF formats + *liveBook*, our enhanced eBook format accessible from any web browser.

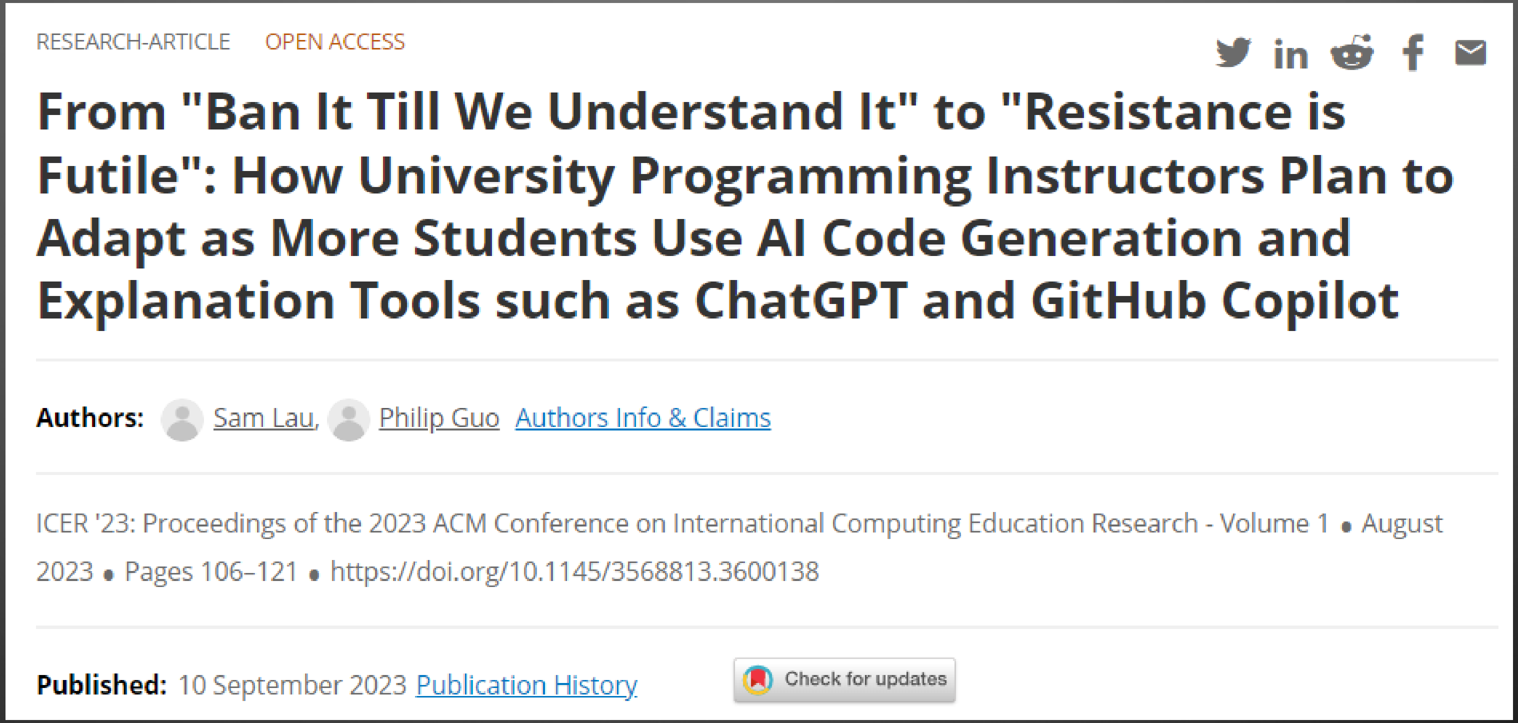
www.manning.com/books/learn-ai-assisted-python-programming



In *Learn AI-Assisted Python Programming: With GitHub Copilot and ChatGPT* you'll learn how to:

- Write fun and useful Python applications—no programming experience required!
- Use the Copilot AI coding assistant to create Python programs
- Write prompts that tell Copilot exactly what to do
- Read Python code and understand what it does
- Test your programs to make sure they work the way you want them to
- Fix code with prompt engineering or human tweaks
- Apply Python creatively to help out on the job

Learn AI-Assisted Python Programming: With GitHub Copilot and ChatGPT is a hands-on beginner's guide that is written by two esteemed computer science university professors. It teaches you everything you need to start programming Python in an AI-first world. You'll hit the ground running, writing prompts that tell your AI-assistant exactly what you want your programs to do. Along the way, you'll pick up the essentials of Python programming and practice the higher-level thinking you'll need to create working apps for data analysis, automating tedious tasks, and even video games.



RESEARCH-ARTICLE OPEN ACCESS

From "Ban It Till We Understand It" to "Resistance is Futile": How University Programming Instructors Plan to Adapt as More Students Use AI Code Generation and Explanation Tools such as ChatGPT and GitHub Copilot

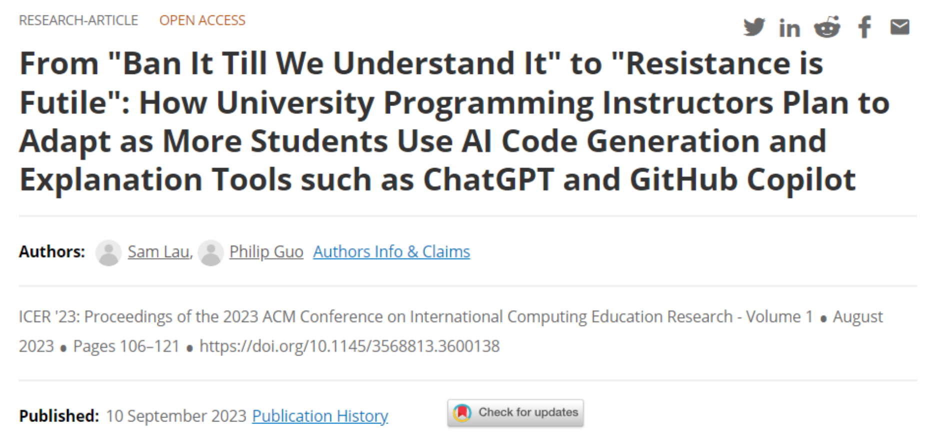
Authors: Sam Lau, Philip Guo Authors Info & Claims

ICER '23: Proceedings of the 2023 ACM Conference on International Computing Education Research - Volume 1 • August 2023 • Pages 106–121 • https://doi.org/10.1145/3568813.3600138

Published: 10 September 2023 Publication History

[3]

OPEN RESEARCH QUESTIONS



RESEARCH-ARTICLE OPEN ACCESS

From "Ban It Till We Understand It" to "Resistance is Futile": How University Programming Instructors Plan to Adapt as More Students Use AI Code Generation and Explanation Tools such as ChatGPT and GitHub Copilot

Authors: Sam Lau, Philip Guo Authors Info & Claims

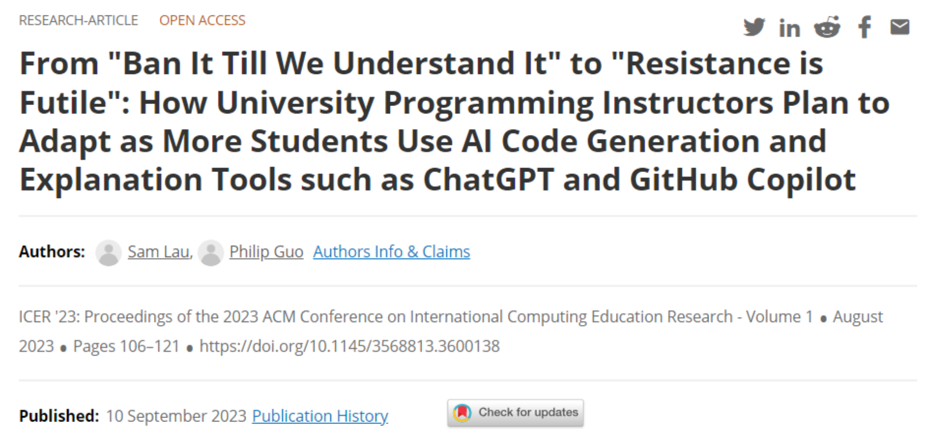
ICER '23: Proceedings of the 2023 ACM Conference on International Computing Education Research - Volume 1 • August 2023 • Pages 106–121 • https://doi.org/10.1145/3568813.3600138

Published: 10 September 2023 Publication History

- Theory Building: e.g. What mental models do novices currently form both about the code that AI generates and about how the AI works to produce that code?
- Scaffolding Novice Understanding: e.g How can we add pedagogical scaffolds to the outputs of AI tools to help novices understand how they are coming up with their code suggestions or explanations?

[3]

OPEN RESEARCH QUESTIONS



RESEARCH-ARTICLE OPEN ACCESS

From "Ban It Till We Understand It" to "Resistance is Futile": How University Programming Instructors Plan to Adapt as More Students Use AI Code Generation and Explanation Tools such as ChatGPT and GitHub Copilot

Authors: Sam Lau, Philip Guo Authors Info & Claims

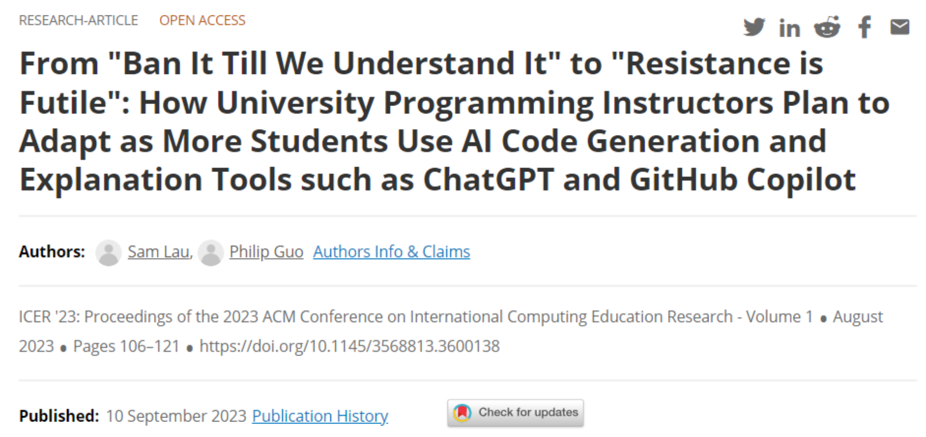
ICER '23: Proceedings of the 2023 ACM Conference on International Computing Education Research - Volume 1 • August 2023 • Pages 106–121 • https://doi.org/10.1145/3568813.3600138

Published: 10 September 2023 Publication History

- Tailoring AI coding tools for pedagogy: Move from efficiency to understanding
- Adapting IDEs for AI-aware pedagogy: How can we redesign IDEs to foster code comprehension and critique?
- Equity and access: How can we design curricula that use these AI tools in such a way to work toward greater equity and access?
- Efficacy studies: How can we tell whether AI tools in introductory courses make students more effective?

[3]

OPEN RESEARCH QUESTIONS



RESEARCH-ARTICLE OPEN ACCESS

From "Ban It Till We Understand It" to "Resistance is Futile": How University Programming Instructors Plan to Adapt as More Students Use AI Code Generation and Explanation Tools such as ChatGPT and GitHub Copilot

Authors: Sam Lau, Philip Guo Authors Info & Claims

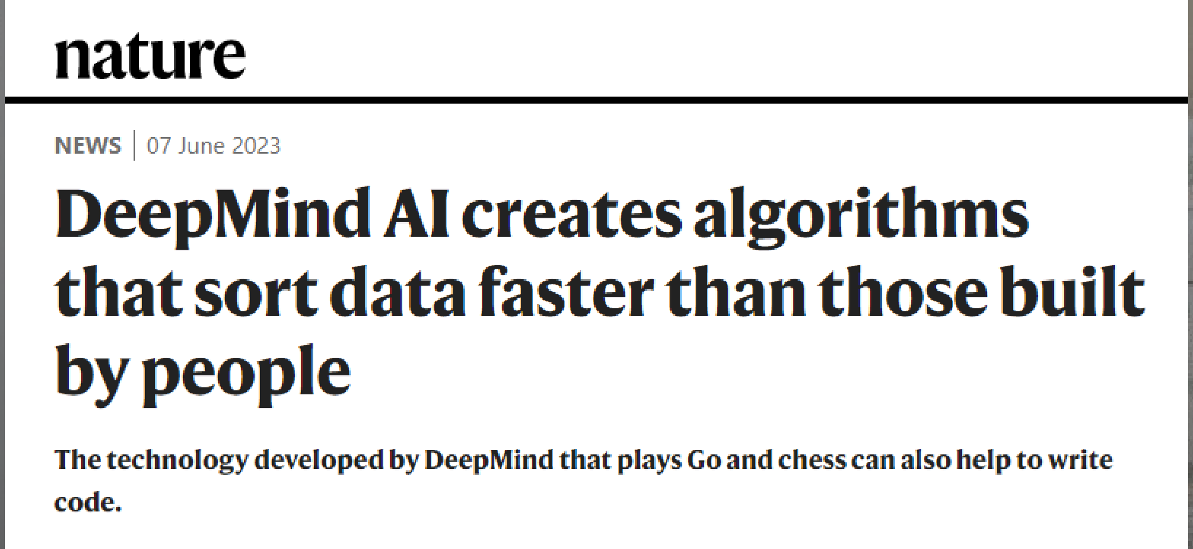
ICER '23: Proceedings of the 2023 ACM Conference on International Computing Education Research - Volume 1 • August 2023 • Pages 106–121 • https://doi.org/10.1145/3568813.3600138

Published: 10 September 2023 Publication History

- Evaluating AI-aware assessments: Can we effectively assess student knowledge if future students collaborate with AI tools on their assignments (and perhaps even on exams)?
- How will these tools affect upper-level courses?
- What about the non-programming aspects of computing? Algorithms? Hardware?
- How can these tools help us deal with scale?

[3]

Not Just Code

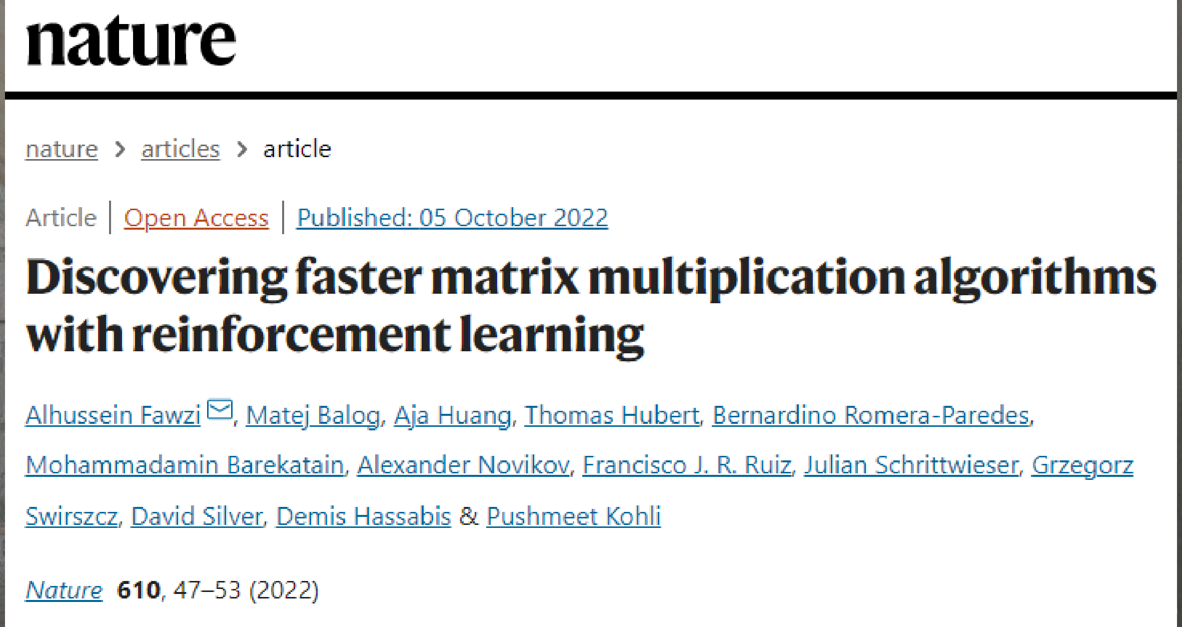


nature

NEWS | 07 June 2023

DeepMind AI creates algorithms that sort data faster than those built by people

The technology developed by DeepMind that plays Go and chess can also help to write code.



nature

nature › articles › article

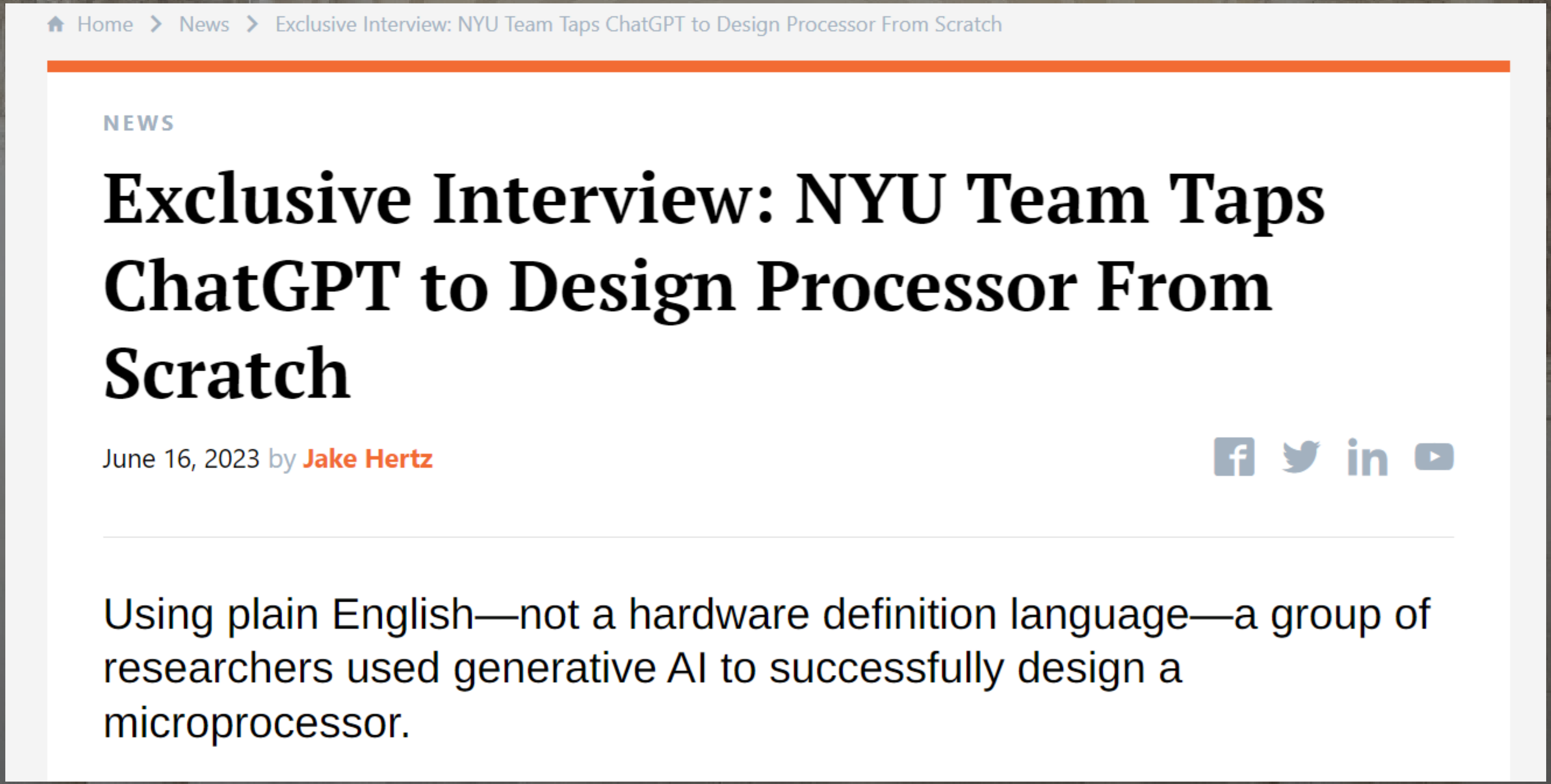
Article | Open Access | Published: 05 October 2022

Discovering faster matrix multiplication algorithms with reinforcement learning

Alhussein Fawzi, Matej Balog, Aja Huang, Thomas Hubert, Bernardino Romera-Paredes, Mohammadamin Barekatain, Alexander Novikov, Francisco J. R. Ruiz, Julian Schrittwieser, Grzegorz Swirszcz, David Silver, Demis Hassabis & Pushmeet Kohli

Nature **610**, 47–53 (2022)

Not Just Software

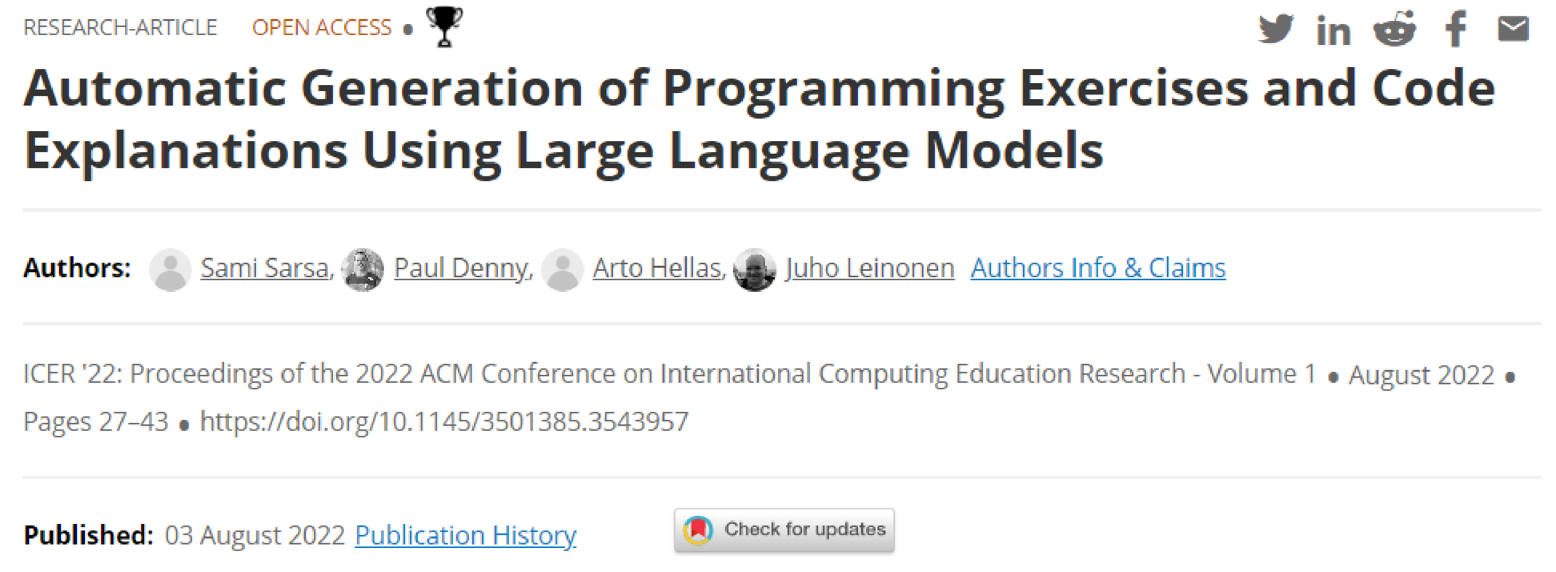

Home > News > Exclusive Interview: NYU Team Taps ChatGPT to Design Processor From Scratch

NEWS

Exclusive Interview: NYU Team Taps ChatGPT to Design Processor From Scratch

June 16, 2023 by **Jake Hertz**

Using plain English—not a hardware definition language—a group of researchers used generative AI to successfully design a microprocessor.



RESEARCH-ARTICLE OPEN ACCESS

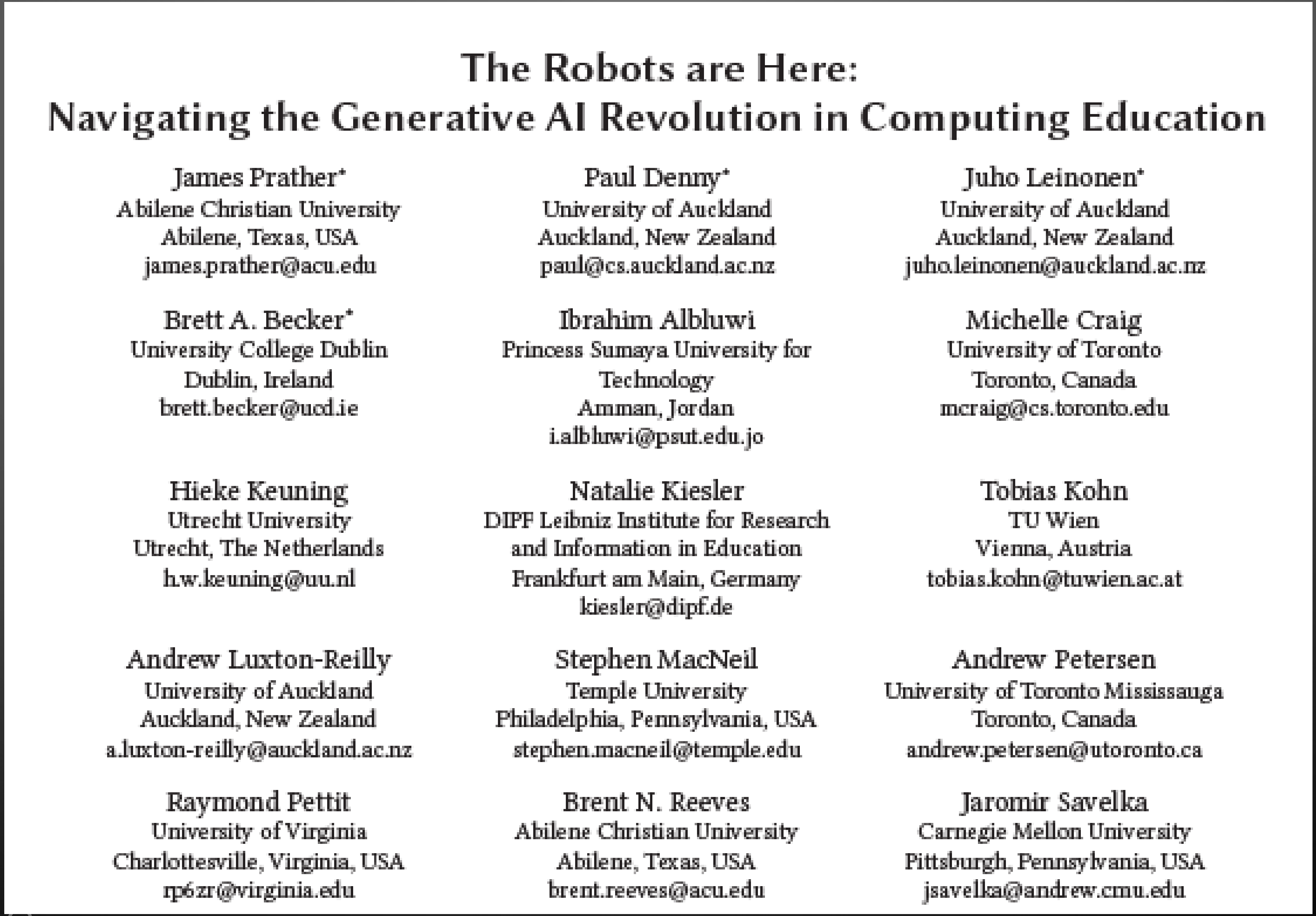
Automatic Generation of Programming Exercises and Code Explanations Using Large Language Models

Authors: Sami Sarsa, Paul Denny, Arto Hellas, Juho Leinonen Authors Info & Claims

ICER '22: Proceedings of the 2022 ACM Conference on International Computing Education Research - Volume 1 • August 2022 • Pages 27–43 • https://doi.org/10.1145/3501385.3543957

Published: 03 August 2022 Publication History

[4]



The Robots are Here: Navigating the Generative AI Revolution in Computing Education

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

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

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

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

Ibrahim Albluwi
Princess Sumaya University for Technology
Amman, Jordan
i.albluwi@psut.edu.jo

Michelle Craig
University of Toronto
Toronto, Canada
mcraig@cs.toronto.edu

Hieke Keuning
Utrecht University
Utrecht, The Netherlands
h.w.keuning@uu.nl

Natalie Kiesler
DIPF Leibniz Institute for Research and Information in Education
Frankfurt am Main, Germany
kiesler@dipf.de

Tobias Kohn
TU Wien
Vienna, Austria
tobias.kohn@tuwien.ac.at

Andrew Luxton-Reilly
University of Auckland
Auckland, New Zealand
a.luxton-reilly@auckland.ac.nz

Stephen MacNeil
Temple University
Philadelphia, Pennsylvania, USA
stephen.macneil@temple.edu

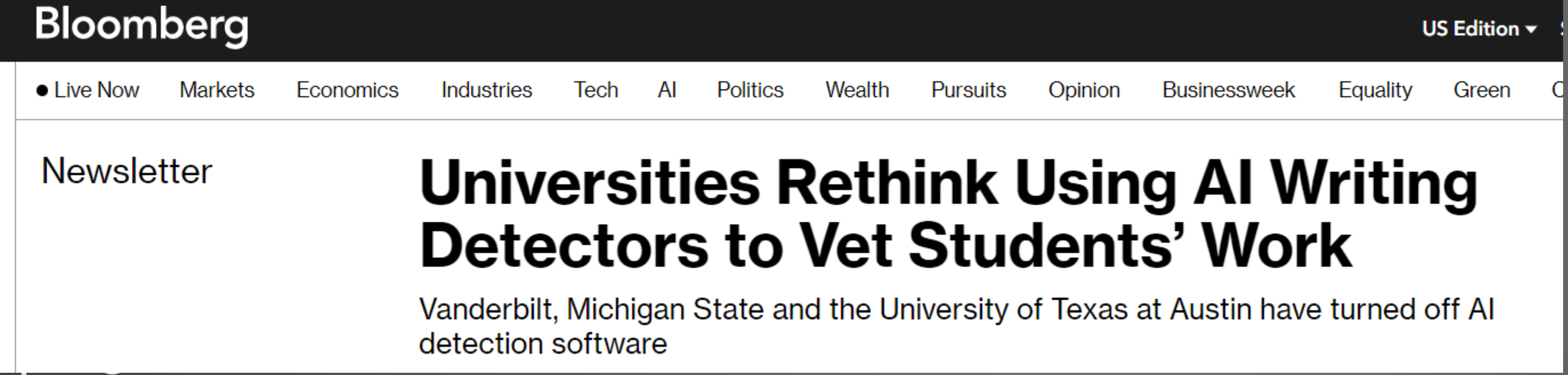
Andrew Petersen
University of Toronto Mississauga
Toronto, Canada
andrew.petersen@utoronto.ca

Raymond Pettit
University of Virginia
Charlottesville, Virginia, USA
rp6zr@virginia.edu

Brent N. Reeves
Abilene Christian University
Abilene, Texas, USA
brent.reeves@acu.edu

Jaromir Savelka
Carnegie Mellon University
Pittsburgh, Pennsylvania, USA
jsavelka@andrew.cmu.edu

[5]

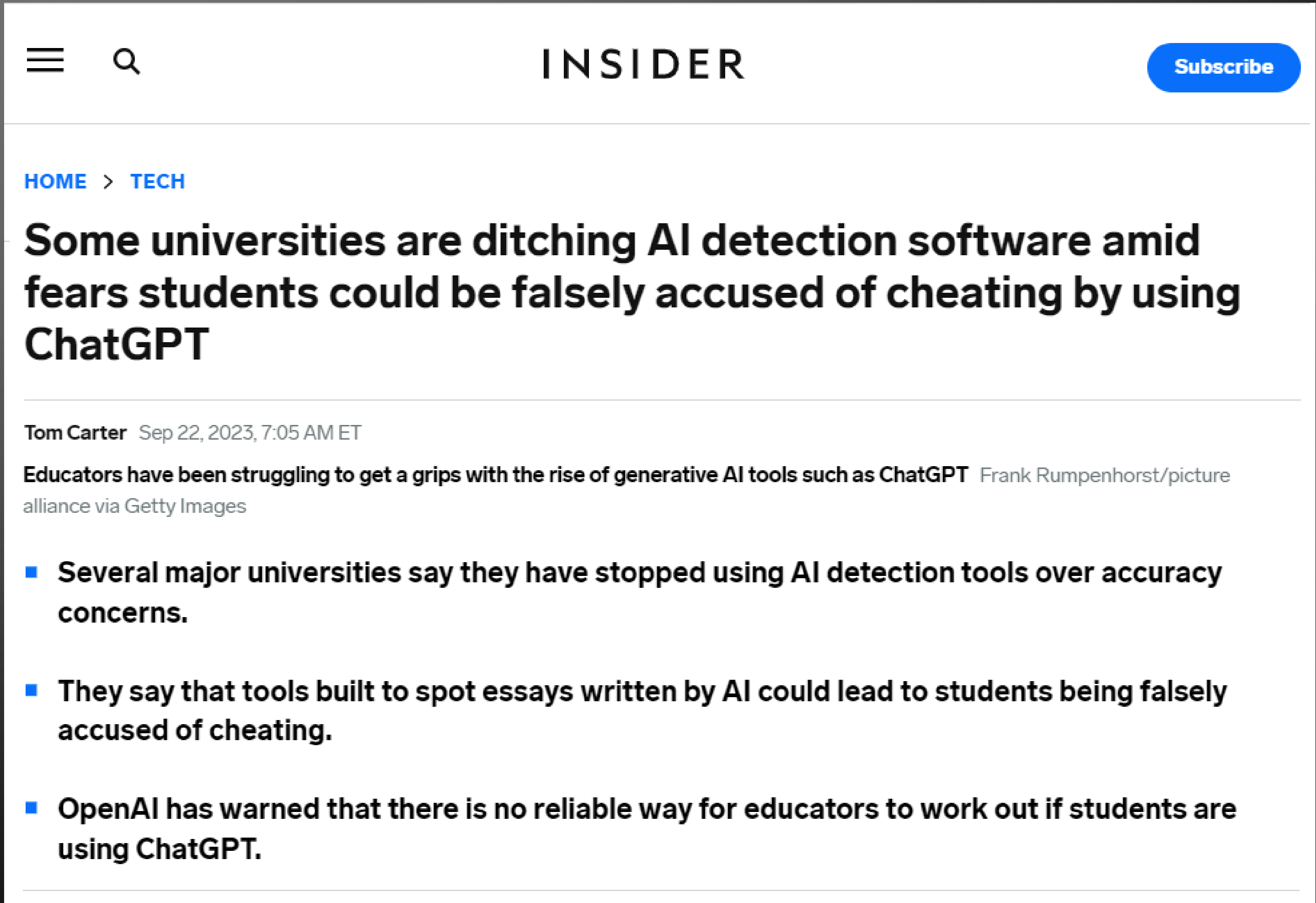


Newsletter

Universities Rethink Using AI Writing Detectors to Vet Students' Work

Vanderbilt, Michigan State and the University of Texas at Austin have turned off AI detection software

www.bloomberg.com/news/newsletters/2023-09-21/universities-rethink-using-ai-writing-detectors-to-vet-students-work



Some universities are ditching AI detection software amid fears students could be falsely accused of cheating by using ChatGPT

Tom Carter Sep 22, 2023, 7:05 AM ET

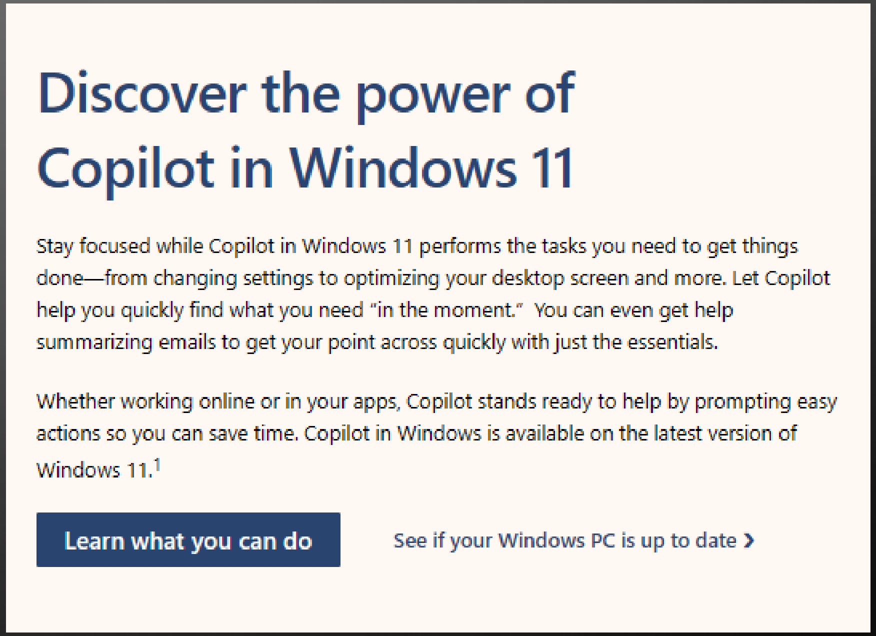
Educators have been struggling to get a grips with the rise of generative AI tools such as ChatGPT Frank Rumpenhorst/picture alliance via Getty Images

- **Several major universities say they have stopped using AI detection tools over accuracy concerns.**
- **They say that tools built to spot essays written by AI could lead to students being falsely accused of cheating.**
- **OpenAI has warned that there is no reliable way for educators to work out if students are using ChatGPT.**

www.businessinsider.com/universities-ditch-ai-detectors-over-fears-students-falsely-accused-cheating-2023-9

WHAT DO WE KNOW RIGHT NOW?

- This is going to be everywhere and it’s not going away (soon, at least)



Discover the power of Copilot in Windows 11

Stay focused while Copilot in Windows 11 performs the tasks you need to get things done—from changing settings to optimizing your desktop screen and more. Let Copilot help you quickly find what you need “in the moment.” You can even get help summarizing emails to get your point across quickly with just the essentials.

Whether working online or in your apps, Copilot stands ready to help by prompting easy actions so you can save time. Copilot in Windows is available on the latest version of Windows 11.[1]

WHAT DO WE KNOW RIGHT NOW?

- Yes, these tools are quite capable
- Yes, we need to keep an eye on assessment but I think the panic is over. This is not a new problem really. And the solutions are the same as always.
- GenAI detectors don't work.
- Students are using GenAI.
- Educators are using GenAI.
- We aren't going to just be teaching about AI any more – we are going to be teaching **with** it, while our students are **using** it.
  - In education, AI is one thing; Generative AI is another.

WHERE ARE WE GOING?

- GenAI as a Pair Programmer (or, copilot) 🙄
- GenAI as a virtual TAs?
- AI in Education community (see AIED conference): Mastery Learning combined with Personalised Tutoring = success. They've been saying that AI was going to get us here for decades. They just didn't see "GenAI" coming.
- Will this change our intakes? Curricula? Hidden/Implicit Curricula?
- Are we going to be forced to not ignore the research we've been ignoring for the last several decades (Pair Programming (see above) being a notable exception)?
- Are we teaching through the lens of the Turing Trap?
- Are we teaching our computing students as not just the (future) creators of this tech, but the consumers?
- Will we need to reshape EVERTHING? Assessment (no, not that assessment) – Will we grade students in terms of the improvement they demonstrate between week 1 and week 12, and not on if everyone met the same learning outcomes?

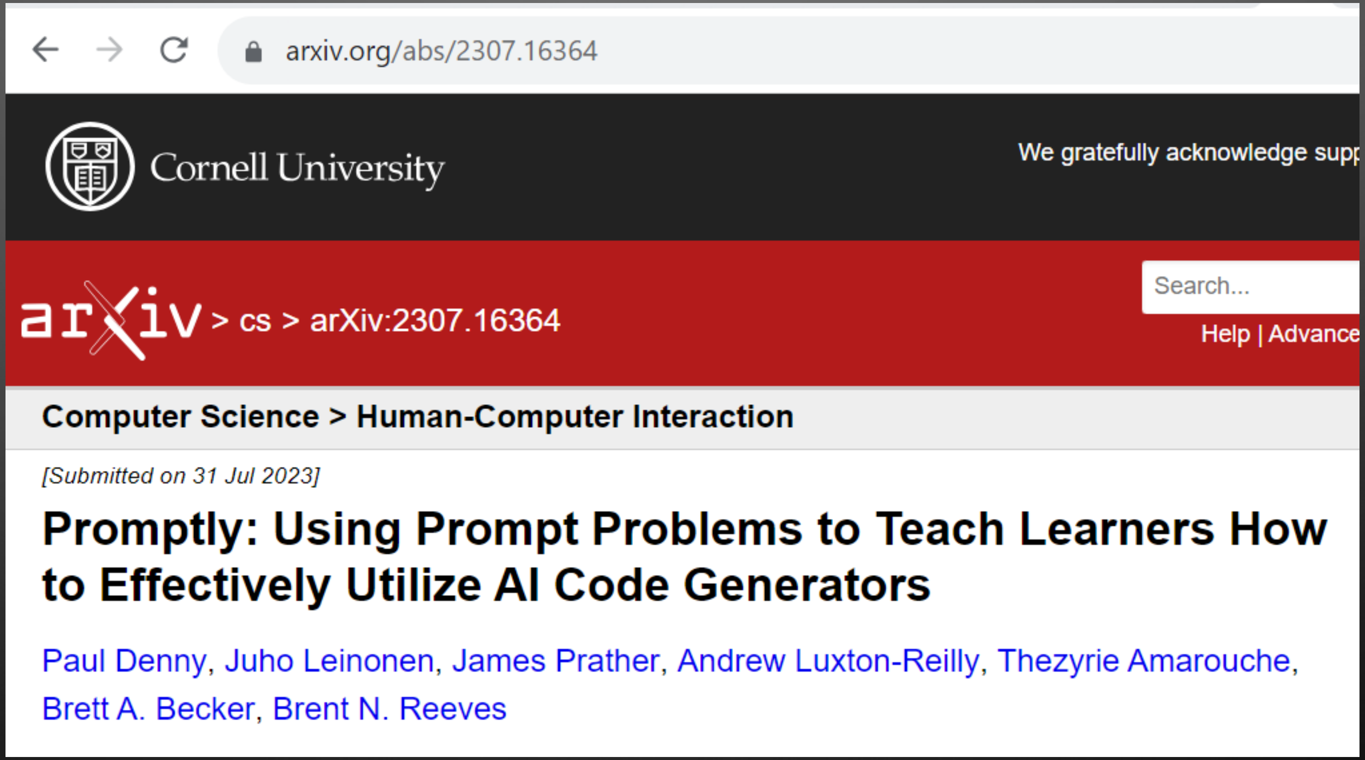
WHERE ARE WE GOING?

- Are we teaching our computing students as not just the (future) creators of this tech, but the consumers?
- Are we teaching through the lens of the Turing Trap?
- Will we need to reshape EVERTHING? For instance, will we grade students in terms of the improvement they demonstrate between week 1 and week 12, and not on if <u>everyone</u> met the <u>same</u> learning outcomes?

PROMPTING

- Images/voice (it’s here now)
- Voice is no big deal, but the point is this is becoming more “natural”
- Is there *pedagogical value* in how students craft prompts, and helping them do so effectively (problem solving, decomposition, abstraction, modularity, …)?

- Will students need help to craft the most effective prompts?
- Can they learn from prompting?



arxiv.org/abs/2307.16364

Cornell University

We gratefully acknowledge supp

arXiv > cs > arXiv:2307.16364

Search...

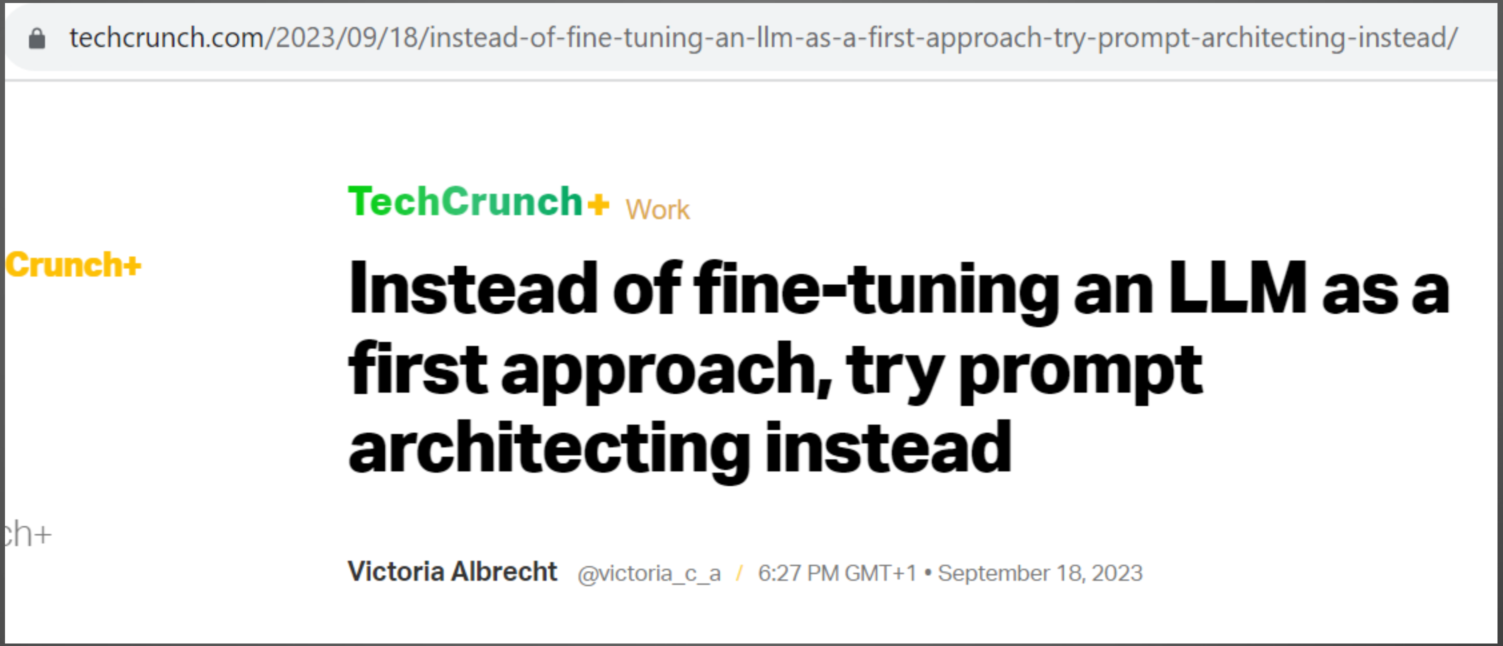
Help | Advance

Computer Science > Human-Computer Interaction

[Submitted on 31 Jul 2023]

Promptly: Using Prompt Problems to Teach Learners How to Effectively Utilize AI Code Generators

Paul Denny, Juho Leinonen, James Prather, Andrew Luxton-Reilly, Thezyrie Amarouche, Brett A. Becker, Brent N. Reeves


techcrunch.com/2023/09/18/instead-of-fine-tuning-an-llm-as-a-first-approach-try-prompt-architecting-instead/

TechCrunch+ Work

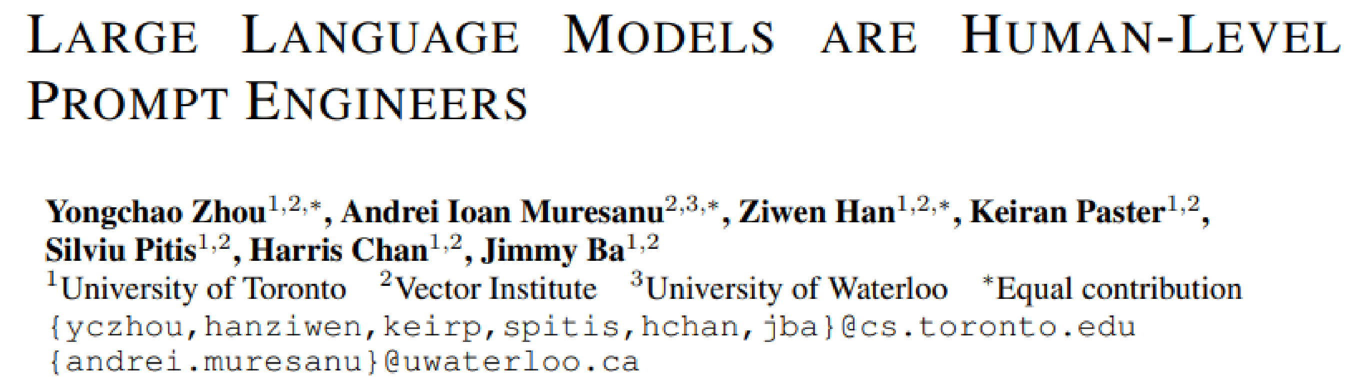
Crunch+

Instead of fine-tuning an LLM as a first approach, try prompt architecting instead

ch+

Victoria Albrecht @victoria_c_a / 6:27 PM GMT+1 • September 18, 2023

- TL;DR: “Given the right sequence of prompts, LLMs are remarkably smart at bending to your will. The LLM itself or its training data need not be modified in order to tailor it to specific data or domain information.”

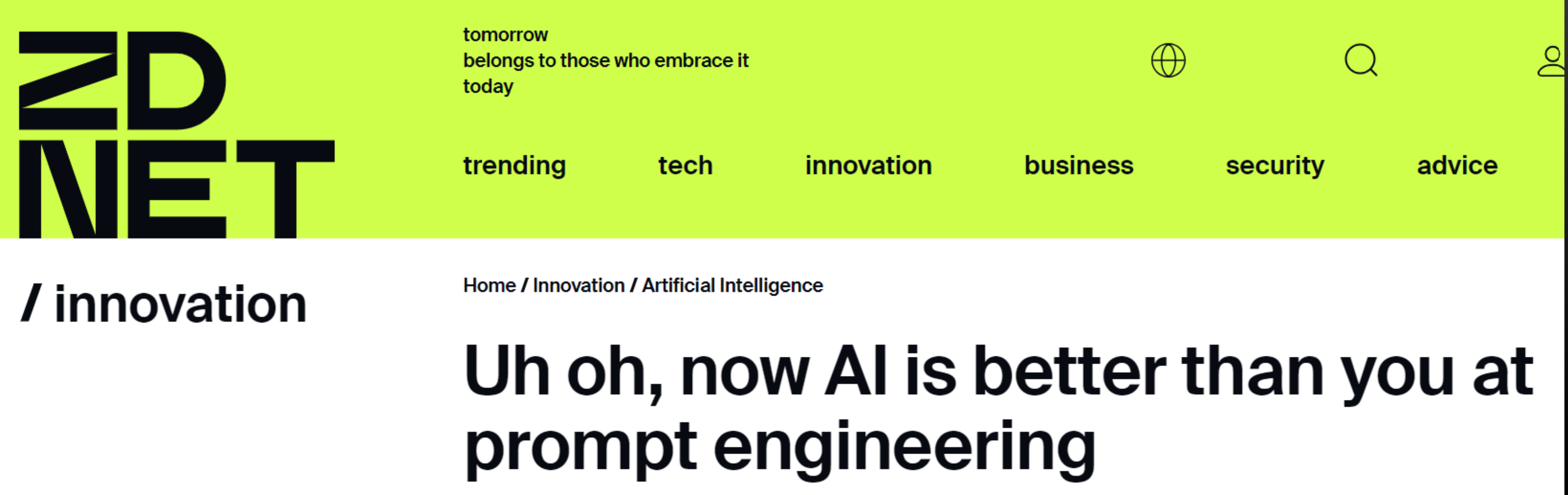


LARGE LANGUAGE MODELS ARE HUMAN-LEVEL PROMPT ENGINEERS

Yongchao Zhou[1,2,*], **Andrei Ioan Muresanu**[2,3,*], **Ziwen Han**[1,2,*], **Keiran Paster**[1,2],
Silviu Pitis[1,2], **Harris Chan**[1,2], **Jimmy Ba**[1,2]
[1]University of Toronto [2]Vector Institute [3]University of Waterloo [*]Equal contribution
{yczhou,hanziwen,keirp,spitis,hchan,jba}@cs.toronto.edu
{andrei.muresanu}@uwaterloo.ca

‘The prompt is the program.’ [6]

www.zdnet.com/article/uh-oh-now-ai-is-better-than-you-at-prompt-engineering



Home / Innovation / Artificial Intelligence

Uh oh, now AI is better than you at prompt engineering

Object Oriented Programming

Functional Programming

Assembly Programming

Concurrent Programming

High Level Programming

Procedural Programming

Prompt Programming

Literate Programming

Natural Language Programming

Logic Programming

Conversational Programming

Declarative Programming

Multi-Agent Programming

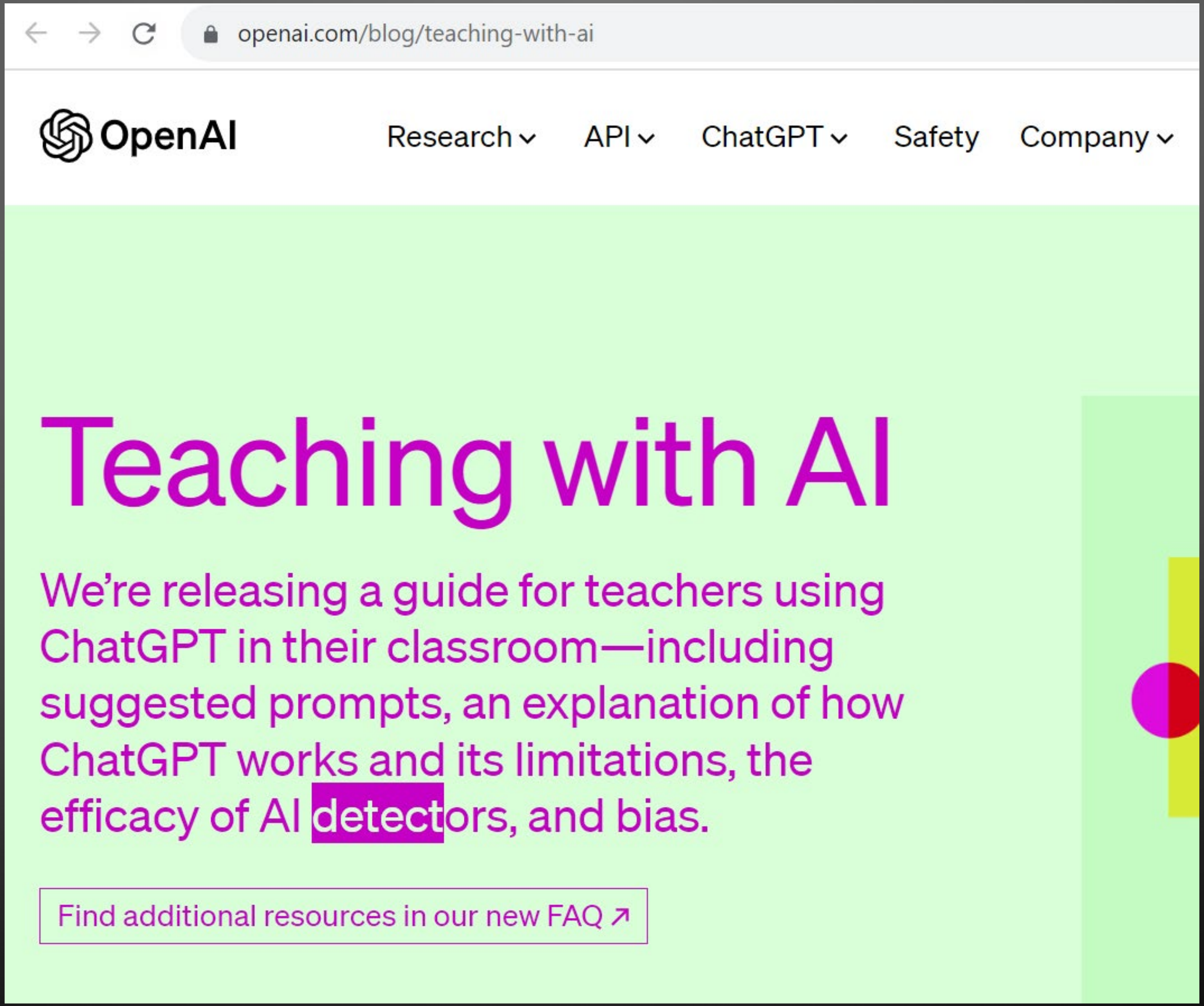
Imperative Programming

Genetic Programming

Parallel Programming

Reactive Programming

Event-Driven Programming


openai.com/blog/teaching-with-ai
OpenAI
Research
API
ChatGPT
Safety
Company

Teaching with AI

We're releasing a guide for teachers using ChatGPT in their classroom—including suggested prompts, an explanation of how ChatGPT works and its limitations, the efficacy of AI detectors, and bias.

Find additional resources in our new FAQ ↗

[1] No More Pencils No More Books: Capabilities of Generative AI on Irish and UK Computer Science School Leaving Examinations. *Mahon, Mac Namee & Becker*. (UKICER 2023)

- brettbecker.com/publications/#[124]

[2] Chat Overflow: Artificially Intelligent Models for Computing Education – *renAIssance or apocAIypse? Paul Denny, Brett A. Becker, Juho Leinonen and James Prather. (*ACM ITiCSE 2023 Keynote).

- brettbecker.com/publications/#iticse23keynote (Video & Slides)

[3] From "Ban It Till We Understand It" to "Resistance is Futile": How University Programming Instructors Plan to Adapt as More Students Use AI Code Generation and Explanation Tools such as ChatGPT and GitHub Copilot. *Sam Lau and Philip Guo.* (ACM ICER 2023)

- dl.acm.org/doi/10.1145/3568813.3600138

[4] Automatic Generation of Programming Exercises and Code Explanations Using Large Language Models. *Sami Sarsa, Paul Denny, Arto Hellas, Juho Leinonen.* (ACM ICER 2022).

- dl.acm.org/doi/10.1145/3501385.3543957

[5] The Robots are Here: Navigating the Generative AI Revolution in Computing Education. *James Prather, Paul Denny, Brett A. Becker et al.* (ACM ITiCSE 2023 Working Group).

- https://arxiv.org/abs/2310.00658

[6] Large Language Models Are Human-Level Prompt Engineers. *Yongchao Zhou, Andrei Ioan Muresanu, Ziwen Han, Keiran Paster, Silviu Pitis, Harris Chan, Jimmy Ba.*

- arxiv.org/abs/2211.01910