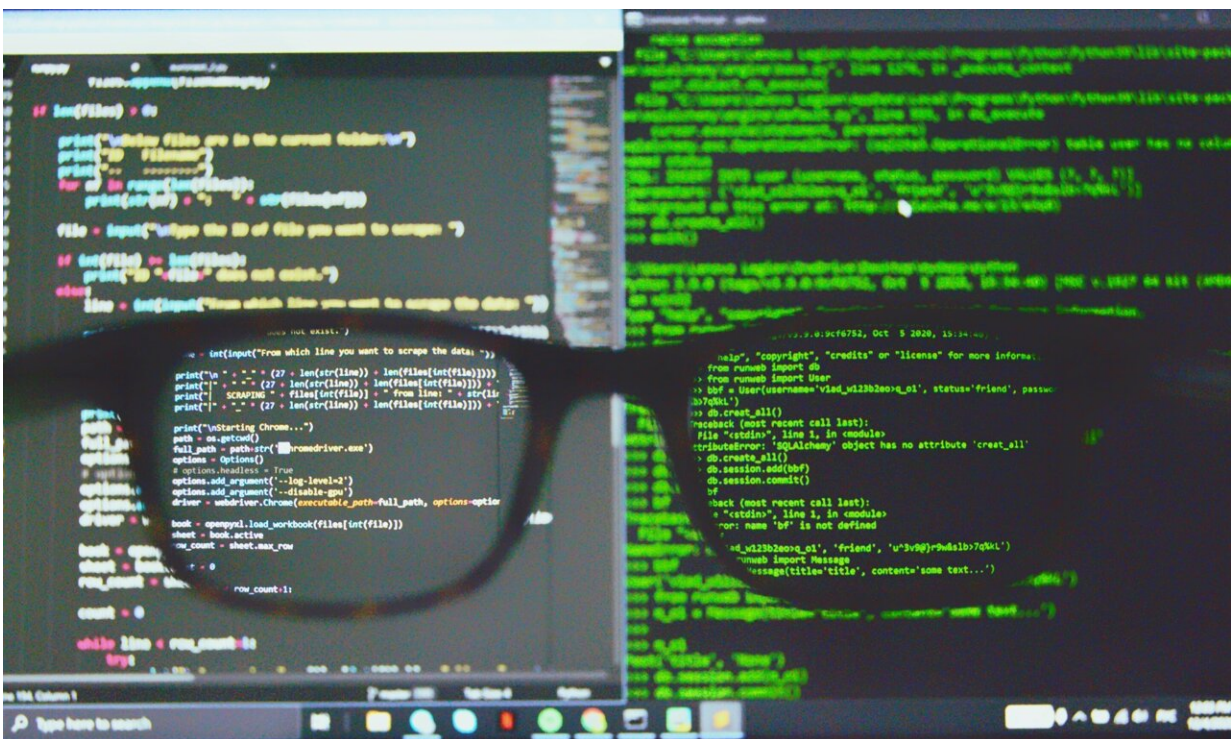# Computer scientists develop open-source tool for dramatically speeding up the programming language Python

August 28 2023

Credit: Unsplash/CC0 Public Domain

A team of computer scientists at the University of Massachusetts Amherst, led by Emery Berger, recently unveiled a prize-winning Python profiler called Scalene. Programs written with Python are notoriously

slow—up to 60,000 times slower than code written in other programming languages—and Scalene works to efficiently identify exactly where Python is lagging, allowing programmers to troubleshoot and streamline their code for higher performance.

There are many different [programming languages](#)—C++, Fortran and Java are some of the more well-known ones—but, in recent years, one language has become nearly ubiquitous: Python.

"Python is a 'batteries-included' language," says Berger, who is a professor of computer science in the Manning College of Information and Computer Sciences at UMass Amherst, "and it has become very popular in the age of data science and [machine learning](#) because it is so user-friendly." The [language](#) comes with libraries of easy-to-use tools and has an intuitive and readable syntax, allowing users to quickly begin writing Python code.

"But Python is crazy inefficient," says Berger. "It easily runs between 100 to 1,000 times slower than other languages, and some tasks might take 60,000 times as long in Python."

Programmers have long known this, and to help fight Python's inefficiency, they can use tools called "profilers." Profilers run programs and then pinpoint which parts are slow and why.

Unfortunately, existing profilers do surprisingly little to help Python programmers. At best, they indicate that a region of code is slow, and leave it to the programmer to figure out what, if anything, can be done.

Berger's team, which included UMass computer science graduate students Sam Stern and Juan Altmayer Pizzorno, built Scalene to be the first profiler that not only precisely identifies inefficiencies in Python code, but also uses AI to suggest how the code can be improved.

"Scalene first teases out where your program is wasting time," Berger says. It focuses on three key areas—the CPU, GPU and memory usage—that are responsible for the majority of Python's sluggish speed.

Once Scalene has identified where Python is having trouble keeping up, it then uses AI—leveraging the same technology underpinning ChatGPT—to suggest ways to optimize individual lines, or even groupings of code. "This is an actionable dashboard," says Berger. "It's not just a speedometer telling you how fast or slow your car is going, it tells you if you could be going faster, why your speed is affected and what you can do to get up to maximum speed."

"Computers are no longer getting faster," says Berger. "Future improvements in speed will come less from better hardware and more from faster, more efficient programming."

Scalene is already in wide use and has been downloaded more than 750,000 times since its public unveiling on GitHub. A paper describing this work appeared at this year's USENIX Conference on Operating System Design and Implementation, where it won a Best Paper Award.

More information: GitHub: github.com/plasma-umass/scalene

Provided by University of Massachusetts Amherst