

Keynote speech

From Catching Cheaters to Helping Students Become Better Programmers

Badri Adhikari

University of Missouri–St. Louis

USA

adhikarib@umsl.edu

ABSTRACT

While AI has many productive uses, many students now use it to skip the thinking. One instinctive response from institutions and teachers is AI detection, but detection works poorly on code. This session introduces a different approach: instead of assessing just the finished code students submit, make the coding process itself visible, giving instructors and students something concrete to talk about. Details such as the history of successful and failed runs, paste events, and active coding time shift the instructor-student conversation from "Did you cheat?" to "How did you learn?" For example, code execution history can distinguish a student who debugged their way to a solution from one whose code simply worked on the first run. Furthermore, encouraging students to reflect on their own coding process builds reflective habits that outlast a single lab assignment. The session will close with how colleges in Nepal have already adopted free tools that support such practices in their programming labs and assignments.

Index Terms— *Academic integrity, Computer science education, AI detection, Introductory programming (CS1), Formative assessment, Metacognition and reflection, Programming assignments, Learning analytics.*

Speaker's short biography:



Dr. Adhikari is an Associate Professor in the Computer Science Department at the University of Missouri–St. Louis (UMSL). His work focuses on artificial intelligence, educational technology, and the responsible use of AI in learning. His research has been supported by the U.S. National Science Foundation. At UMSL, he teaches Deep Learning, Artificial Intelligence, Data Visualization, and Interpretable Machine Learning.