Software engineering
research

Bugs and artificial intelligence: what is happening in the field of software engineering?

Software plays a huge role behind the scenes of our lives, but there can be disastrous consequences when it malfunctions. **Dr Masud Rahman**, a software engineering researcher at **Dalhousie University** in Canada, is studying how to fix software bugs, as well as confronting the challenges and opportunities that artificial intelligence (AI) is bringing to this field.

Dr Masud Rahman

RAISE Lab, Faculty of Computer Science, Dalhousie University, Canada

Field of research
Software engineering

Research project
'Mining, Intelligence and Automation in Tackling Machine-Learning Bugs' – developing new tools and methods for tackling deep learning bugs

Funder
Natural Sciences and Engineering Research Council of Canada (NSERC)

Website
raise.cs.dal.ca

doi: 10.33424/FUTURUM632

PROFILE

Talk like a …
software engineering researcher

Artificial intelligence (AI) — the ability of computers to perform tasks previously done by humans, such as translating languages, analysing data and making recommendations

Bugs — a flaw or error in a computer program that causes the machine to behave incorrectly

Debugging — the process of removing errors so that a computer program runs as expected

Deep learning — a method of teaching AI models to process complex data so that they can mimic the human brain

Large language models (LLMs) — AI models that have been trained on large amounts of text to understand human language

Software developer — a person developing software by writing programming code

Software engineer — a person designing and developing software following the best engineering practices

Although software bugs are tiny errors in computer code, the impact of the failures they cause can be catastrophic. Software bugs cost the global economy trillions of dollars every year and have been the cause of significant information leaks, air and spacecraft crashes, power outages, company bankruptcies and medical disasters. Being able to find and fix these errors is a massive part of the work of software engineers and is crucial for creating safe and dependable software. But why is it so difficult?

Maintaining and fixing software
Unlike other consumer products, software is unique in that once it has been produced and is in use by customers, engineers still spend time changing it. Think of an app that you use regularly, and you will probably realise how certain features have been changed or altered. For example, in July 2025 alone, TikTok's updates ranged from launching 'Bulletin Boards' to releasing its own font, TikTok Sans. Such updates are part of

what is called the maintenance phase of software and take up around 50% of software developers' time. "Software might go through changes due to a number of circumstances," explains Dr Masud Rahman of Dalhousie University. "It might encounter a failure that needs to be addressed immediately, it might miss very important features that need to be incorporated, or it might be changed to run on an operating system for which it was not originally designed."

If a user (often a regular customer) notices a bug in the software – something that does not seem to be working properly – they might submit a complaint explaining it. Developers then try and recreate the bug on their own machine. "Reproduction is useful in two ways," says Masud. "Firstly, it confirms the presence of the reported bug in the software. Secondly, it provides complementary information about the bug (e.g., symptoms, root causes), which might not be present in the bug report."

However, recreating a bug is not straightforward. "Software users (a.k.a., bug reporters) often fail to provide the necessary details in their bug reports due to a lack of experience or security/privacy concerns," explains Masud. "In some cases, recreating the same reported scenario on a developer's machine is either infeasible or prohibitively costly."

How is AI changing software engineering?

Up until now, software has always been written by human developers. This has meant that if there was a bug hidden within a piece of code, it was relatively easy for developers to find and fix it, as they were often the ones who wrote the code in the first place! However, in recent years, more companies are beginning to use artificial intelligence (AI) to write their software.

While AI models are bringing huge benefits to the world of software engineering, they come with a new wave of challenges too. "AI (a.k.a., Deep learning)

> **Unlike traditional bugs, deep learning bugs are often unpredictable, data-driven, multifaceted and highly complex.**

models learn to mimic human capabilities, such as writing an essay or a computer program, by looking at tons of human-generated artifacts (e.g., code, texts) and can sometimes be unpredictable. Thus, if something goes wrong within the AI model writing software, whose working process is not well known to the developers, it is difficult for them to detect and correct," says Masud.

As well as being difficult to find, studies have found that only 3% of deep learning bugs are reproducible by developers. "Unlike traditional bugs, deep learning bugs are often unpredictable, data-driven, multifaceted and highly complex," says Masud.

While many tools have been designed over the last few decades to help find traditional software bugs, these methods are not capable of tackling errors in deep learning systems.

The need for new methods

In his software engineering research lab, Masud and his team are trying to detect and reproduce more of these deep learning bugs and find new ways of fixing them. To do this, they use information from the bug report, the faulty version of the code and past changes made to this code, information from programming sites such as Stack Overflow, and help from large language models (LLMs) such as ChatGPT, Llama and Mistral. "We also look at hundreds of software bugs that others have reported, investigated and corrected for an in-depth understanding," says Masud.

The team has already had multiple successes with its work and developed several new tools. "These include Bugsplainer, which explains faulty software code in plain language, and DEFault, which detects and diagnoses faults in deep learning models," says Masud. These tools have attracted new industry partners and funding, and one of Masud's tools was even featured on the **Stack Overflow** blog.

Where is Masud's work going next?

Although there are challenges associated with deep learning bugs, AI brings exciting opportunities too. "LLMs have an impressive understanding of software code and text, indicating their strong potential to support software debugging, including deep learning bugs," explains Masud.

As developments in software engineering go on, researchers like Masud have a lot of work to do to leverage the potential of AI models and help create reliable and sustainable software for years to come.

About *software engineering research*

At the intersection of computer science and engineering lies software engineering, an advancing field which involves designing, developing, testing and maintaining software applications. Software covers a huge variety of tools, including web browsers like Google Chrome and Mozilla Firefox, operating systems such as Windows and Android, productivity software such as Microsoft Office, and graphic design tools such as Adobe Photoshop. Software engineering is a massive field with wide-reaching applications which will only continue to grow as technology and computers become even more involved in different aspects of our lives.

While working in software engineering involves a significant amount of programming and fixing errors in code, there is much more to it than this. "Software engineers spend a lot of time working alongside product managers, customers, end users and other developers, so gaining soft skills such as communication and team-working is super important," says Masud.

How will AI change software engineering?

"Over the last few decades, software systems have been developed by human developers who wrote programming code themselves to solve many business problems," says Masud. "However, with the rise of AI models capable of writing programming code (e.g., ChatGPT, Claude, Gemini), software engineering is expected to evolve significantly. That is, most of the code in future software systems will be written by AI models under human oversight."

"AI developers (a.k.a., AI agents) have different strengths and weaknesses, which will shape in part how the future software is developed and maintained," adds Masud. "Thus, the next generation of software engineering researchers will study the challenges faced by AI developers and design novel ways and methods to combine AI capabilities with human strengths or specialities to build better software. Incorporating human values – ethics, fairness, universal human rights, equity, diversity and inclusion – into the AI-written software systems will be another challenge."

Pathway from school to *software engineering*

During high school, build a strong foundation in mathematics, computer science and physics.

Visit websites such as **w3schools.com** to learn the basics of coding in your spare time.

At university, study a bachelor's degree in software engineering or computer science. "Foster critical thinking, problem solving, design abstraction, teamwork and soft skills," says Masud. After graduating, you will need to pursue a PhD to work in research like Masud.

Watch this video on computer science at Dalhousie University: **youtube.com/watch?v=VYVfZVW1nls** . The Faculty of Computer Science also has three student societies (**dal.ca/faculty/computerscience/about/student_life.html**) – one for graduate students, one for undergraduate students, and one for female students in technology degrees.

During summer breaks when you are at university, get hands-on experience by doing an internship at a software company and working alongside professional developers. "Most of the major tech giants offer summer student internships – have a look at Google Careers (**google.com/about/careers/applications/buildyourfuture/internships**), Amazon Careers (**amazon.jobs/content/en/career-programs/university/internships-for-students**), and Meta Careers (**metacareers.com/careerprograms/students**)," says Masud.

Masud recommends exploring the SuperNOVA programmes (**supernova.dal.ca**), which advance STEM education for young people. He also recommends HereWeCode (**herewecode.ca**), a computer science outreach programme at Dalhousie University.

Explore careers in
software engineering

Read this interview about what it is like to work as a software engineer: **hbr.org/2021/07/career-crush-what-is-it-like-to-be-a-software-engineer**

"Software Engineering is supported by two major professional societies – IEEE Software (**computer.org/csdl/magazine/so**) and ACM SIGSOFT (**www2.sigsoft.org**)," explains Masud. "They do a great job connecting software engineering practitioners, large companies and academic researchers from all over the world. They also offer student membership, which is a great opportunity for students to network and prepare for the job market."

Have a read of the Stack Overflow blog, where Masud's work was featured, to learn more about the world of programming and computer science: **stackoverflow.blog**

"Knowledge about algorithms, data structures, programming, software design and architecture, and AI will be crucial to working in software engineering. Although AI models will write the majority of the code, humans will need the expertise to review and enhance them. Knowledge about other computer science branches – operating systems, database systems, networking, cybersecurity, and human-computer interaction – will also be extremely valuable for a comprehensive understanding of software systems," says Masud.

According to Indeed, the average annual salary for a software engineer in Canada is CA$100,000.

Meet
Masud

I was introduced to personal computers back in 1997 when I was a high school student at Jhenidah Cadet College, Bangladesh, and I have been fascinated by these machines ever since. My father bought me my first computer in 2004 when I started my bachelor's programme in computer science at Khulna University, Bangladesh.

My software engineering journey started in 2009 when I joined a software company. As a freshman software developer, I not only learnt valuable problem-solving and professional skills but also identified the opportunities for research and innovations in software engineering. This first-hand experience as a professional software developer significantly shaped my career choice as a software engineering researcher. In 2012, I started my graduate studies and research in computer science and software engineering at the University of Saskatchewan, Canada.

My proudest career achievements so far are receiving two gold medals: the Chancellor's Gold Medal in 2010, which is the highest academic award for an undergraduate student in Bangladesh, and the Governor General's Gold Medal in 2020, which is the highest academic award for a Canadian PhD student.

To date, I have co-authored 58 peer-reviewed articles, with 14 top-tier publications and a TCSE Distinguished paper award. I have been awarded substantial funding over the last 5 years by various agencies, which reflects the value and relevance of my work. I am also proud to have attracted many brilliant graduate students, academic collaborators and industry partners (e.g., Mozilla Firefox, MetaBob) to work with.

An academic career is a long journey with many ups and downs. In other words, it's a marathon, not a sprint. I have aimed for a long, impactful academic career where I can challenge myself with exciting, new challenges every day and grow with a lasting legacy.

To quote computer scientist Andrew Ng, "AI is the new electricity". I want to use this 'new electricity' to build innovative, useful things for the software engineering community and beyond. I believe we are witnessing the biggest transformative period of digital innovation and computation after the World Wide Web. I am super-excited about the future and what it holds!

Masud's *top tips*

1. Dream big. To quote A. P. J. Abdul Kalam (aerospace scientist and former president of India), "Dream is not that which you see while sleeping, it is something that does not let you sleep." Big dreams bring motivation, confidence, responsibilities, accountability and resilience.
2. Failure is inevitable. However, a person is not defined by failure, but by how they overcome it. Each life will experience a fair share of disappointments and frustrations, but the dreams and ambitions should overpower them!
3. If a task cannot be completed in its entirety, break it into small parts and distribute them across multiple working days.