Association for Computing Machinery

Advancing Computing as a Science & Profession

NEWS RELEASE

CONTACT: Jim Ormond
212-626-0505
ormond@hq.acm.org

TOP CONFERENCE UNDERSCORES HOW SOFTWARE ENGINEERING POWERS THE DIGITAL AGE

ESEC/FSE Features Outstanding Research and Best Practices in the Field of Software Engineering

New York, NY, August 14– ACM, the Association for Computing Machinery, and the ACM Special Interest Group on Software Engineering (SIGSOFT) will present ESEC/FSE 2019, the 27th ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering, from August 26-30 in Tallinn, Estonia. ESEC/FSE brings together researchers, practitioners, and educators to exchange the latest research results and trends, as well as their practical application in all areas of software engineering.

Software engineering is interwoven throughout almost all aspects of computing and is the engine of many of the fastest-growing areas in technology—including artificial intelligence, mobile apps, and blockchain technologies. ESEC/FSE encompasses the most important developments in the field and includes talks on recent innovations in software engineering; industry best practices; software engineering of the future; workshops, tool demonstrations; and a Student Research Competition.

“By combining the ESEC and FSE conferences into one event, we have formed the world’s most important forum on the state of software engineering research and practice,” says General Co-chair Marlon Dumas, University of Tartu. “At ESEC/FSE researchers and practitioners are the first to interact with the latest innovations in the field through the conference’s highly competitive research papers program as well as outstanding applied research in the industry track. Importantly they also actively engage with their peers by participating in ESEC/FSE’s workshops and tool demonstrations.”

“Several trends are reshaping the software industry, including the growth of cloud computing; big data collection and analytics; artificial intelligence; and cybersecurity,” added General Co-chair Dietmar Pfahl, University of Tartu. “We’ve put together a comprehensive program that also welcomes multidisciplinary work at the interface between software engineering and other topics such as artificial intelligence, human-computer interaction, programming languages and systems engineering.”

ESEC/FSE 2019 HIGHLIGHTS:

Visit here for the full ESEC/FSE 2019 program.

Keynote Addresses

"Living with Feature Interactions"

Joanne Atlee, University of Waterloo

This talk will explore challenges in feature interactions and their resolutions. Resolution strategies can tackle large classes of interactions, but are imperfect and incomplete, leading to research opportunities in software architecture, composition semantics, and verification.

"Safety and Robustness for Deep Learning with Provable Guarantees"

Marta Kwiatkowska, University of Oxford

This lecture will describe progress with developing automated verification and testing techniques for deep neural networks to ensure safety and robustness of their decisions with respect to input perturbations.

"Insights from Open Source Software Supply Chains"

Audris Mockus, University of Tennessee

This talk describes the efforts of Mockus and colleagues to develop an understanding of how the periphery of the Open Source Software (OSS) ecosystem is interconnected through technical dependencies, code sharing, and knowledge flows.

ACM SIGSOFT Distinguished Paper Award

"Empirical Review of Java Program Repair Tools: A Large-Scale Experiment on 2 141 Bugs and 23 551 Repair Attempts"

Thomas Durieux, University of Lisbon; Fernanda Madeiral, Federal University of Uberlândia; Matias Martinez, Université Polytechnique Hauts-de-France; and Rui Abreu, University of Lisbon

In this paper, the authors present a large-scale experiment using 11 Java test-suite-based repair tools and 5 benchmarks of bugs. Their goal is to have a better understanding of the current state of automatic program repair tools on a large diversity of benchmarks.

"Generating Automated and Online Test Oracles for Simulink Models with Continuous and Uncertain Behaviors"

Claudio Menghi, Shiva Nejati, Khouloud Gaaloul, and Lionel Briand, University of Luxembourg

We propose an automated approach to translate cyber physical systems (CPS) requirements specified in a logic-based language into test oracles specified in Simulink—a widely-used development and simulation language for CPS.

"The Importance of Accounting for Real-World Labelling When Predicting Software Vulnerabilities"

Matthieu Jimenez, University of Luxembourg; Renaud Rwemalika, University of Luxembourg; Mike Papadakis, University of Luxembourg; Federica Sarro, University College London; Yves Le Traon, University of Luxembourg; and Mark Harman, University College London
This study investigates the effectiveness of three previously proposed vulnerability prediction approaches, in two settings with and without the unrealistic labelling assumption. The results reveal that the unrealistic labelling assumption can profoundly mis-lead the scientific conclusions drawn; suggesting highly effective and deployable prediction results vanish when we fully account for realistically available labelling in the experimental methodology.

"Assessing the Quality of the Steps to Reproduce in Bug Reports"
Oscar Chaparro, University of Texas, Dallas; Carlos Bernal-Cárdenas, College of William and Mary; Jing Lu, Apple; Kevin Moran, College of William and Mary; Andrian Marcus, University of Texas, Dallas; Massimiliano Di Penta, University of Sannio; Denys Poshyvanyk, College of William & Mary; and Vincent Ng, University of Texas, Dallas
A major problem with user-written bug reports, indicated by developers and documented by researchers, is the (lack of high) quality of the reported steps to reproduce the bugs. Low-quality steps to reproduce lead to excessive manual effort spent on bug triage and resolution. This paper proposes Euler, an approach that automatically identifies and assesses the quality of the steps to reproduce in a bug report, providing feedback to the reporters, which they can use to improve the bug report.

"A Framework for Writing Trigger-Action Todo Comments in Executable Format"
Pengyu Nie, Rishabh Rai, Junyi Jessy Li, Sarfraz Khurshid, and Raymond J. Mooney, University of Texas, Austin
Natural language elements are frequently used to communicate among developers and to describe tasks that need to be performed when specific conditions hold on artifacts related to the code repository. As projects evolve, development processes change, and development teams reorganize, these comments, because of their informal nature, frequently become irrelevant or forgotten. The authors present a framework, dubbed TrigIt, to specify trigger-action todo comments in executable format.

"A Statistics-based Performance Testing Methodology for Cloud Applications"
Sen He, University of Texas, San Antonio; Glenna Manns, University of Virginia; John Saunders, University of Virginia; Wei Wang, University of Texas, San Antonio; Lori Pollock, University of Delaware; Mary Lou Soffa, University of Virginia
In this paper, the authors present a novel cloud performance testing methodology called PT4Cloud. By employing non-parametric statistical approaches of likelihood theory and the bootstrap method, PT4Cloud provides reliable stop conditions to obtain highly accurate performance distributions with confidence bands.

Industry Track Papers (partial list)
"When Deep Learning Met Code Search"
Jose Cambronero, MIT; Hongyu Li, Facebook; Seohyun Kim, Facebook; Koushik Sen, University of California, Berkeley; and Satish Chandra, Facebook Research

Many proposals for using deep neural networks for code search using natural language incorporate the idea of *embedding* code and natural language queries into real vectors and then using vector distance to approximate semantic correlation between code and the query. In this paper, the authors assembled implementations of state-of-the-art techniques to run on a common platform, training and evaluation corpora. To explore the design space in network complexity, they also introduced a new design point that is a *minimal* supervision extension to an existing unsupervised technique.

"WhoDo: Automating Reviewer Suggestions at Scale"

Sumit Asthana, Rahul Kumar, Ranjita Bhagwan, Chetan Bansal, Christian Bird, Chandra Maddila, Sonu Mehta, B. Ashok, Microsoft Research

Today's software development is distributed and involves continuous changes for new features—and yet, the development cycle has to be fast and agile. The authors deploy WhoDO, a reviewer suggestions algorithm. They also evaluate its effect but also incorporate load balancing as part of it to address one of its major shortcomings—of recommending experienced developers very frequently.

"Code Coverage at Google"

Marko Ivanković and Goran Petrović, Google; René Just, University of Washington; and Gordon Fraser, University of Passau

Code coverage is a measure of the degree to which a test suite exercises a software system. Although coverage is well established in software engineering research, deployment in industry is often inhibited by the perceived usefulness and the computational costs of analyzing coverage at scale. At Google, coverage information is computed for one billion lines of code daily, for seven programming languages. This paper describes Google's code coverage infrastructure and how the computed code coverage information is visualized and used.

About ACM SIGSOFT

ACM SIGSOFT focuses on issues related to all aspects of software development and maintenance. Areas of special interest include: requirements, specification and design, software architecture, validation, verification, debugging, software safety, software processes, software management, measurement, user interfaces, configuration management, software engineering environments, and CASE tools. SIGSOFT sponsors or co-sponsors many conferences and events.

About ACM

ACM, the Association for Computing Machinery, is the world's largest educational and scientific computing society, uniting computing educators, researchers and professionals to inspire dialogue, share resources and address the field's challenges. ACM strengthens the computing profession's collective voice through strong leadership, promotion of the highest standards, and recognition of technical excellence. ACM supports the professional growth of its members by providing opportunities for life-long learning, career development, and professional networking.

# # #