Large Scale Clone Detection, Analysis, and Benchmarking: An Evolutionary Perspective (Keynote)

Chanchal K. Roy
Department of Computer Science, University of Saskatchewan, Canada
chanchal.roy@usask.ca

***Abstract*—Copying a code fragment and then reusing it by pasting and adapting (e.g., adding/modifying/deleting statements) is a common practice in software development, which results in a significant amount of duplicated code in software systems. Developers consider cloning as one of the principled reengineering approaches and often intentionally practice cloning for a variety of reasons such as faster development, avoiding risk by reusing stable old code, or for time pressure. On the other hand, duplicated code poses a number of threats to the maintenance of software systems such as clones are the #1 "bad smell" in Flower's refactoring list and several recent studies including studies with industrial systems show that although for many cases clones are not really harmful, and even could be useful for some cases, they could be also detrimental to software maintenance. For example, reusing a fragment containing unknown bugs may result in bugs propagation, or any changes in requirements involving a cloned fragment may lead to changes to all the similar fragments to it, multiplying the work to be done. Furthermore, inconsistent changes to the cloned fragments during any updating processes may lead to severe unexpected behaviour. Software clones are thus considered to be one of the major contributors to the high software maintenance cost, which could be up to 80% of total software development cost. The era of Big Data has introduced new applications for clone detection. For example, clone detection has been used to find similar mobile applications, to intelligently tag code snippets, to identify code examples, and so on from large inter-project repositories. The dual role of clones in software development and maintenance, along with these many emerging new applications of clone detection, has led to a great many clone detection tools and analysis frameworks. In this keynote talk, I will review the cloning literature to date, in particular, I will talk about our recent work on large scale clone detection, and the challenges in evaluating such clone detectors and how we have overcome them at least in part with our BigCloneBench and Mutation framework. I will then talk about the recent advances in clone analysis and management along with a vision for a comprehensive clone management system.**

Short Biography of the Speaker

Chanchal K. Roy is an associate professor of Software Engineering/Computer Science at the University of Saskatchewan, Canada. While he has been working on a broad range of topics in Computer Science, his chief research interest is Software Engineering. In particular, he is interested in software maintenance and evolution, including clone detection, Big Data analytics, recommendation systems, empirical software engineering, crowdsourcing and mining software repositories. He is the co-lead of the Big Data Analytics group of an NSERC Canada First Research Excellence Fund (CFREF) on Food security and a co-investigator of another CFREF grant on water security where he mostly focuses on establishing Big Data cloud frameworks, Big Data analytics techniques, and high-speed processing pipelines towards analyzing, modelling, visualizing and exploring petabytes of heterogeneous types of crop phenotype and water related data.

He has written more than 140 publications that have been cited more than 3600 times. His Google Scholar h-index is 25 and his i10-index index is 64. His contributions to the software maintenance community, and particularly to the cloning community, have been highly influential. He has recently been awarded with two Most Influential Paper (MIP) awards: he has been recognized with a ten-year MIP award at SANER 2018 for their WCRE 2008 paper (with Jim Cordy) on analysis of function clones in open source software, he has also been awarded another MIP award at ICPC 2018 for their ICPC 2008 paper on NiCad clone detection tool. He has been a vision keynote speaker at the joint conferences of WCRE 2014 and CSMR 2014 (now called SANER) on software clones, and a keynote speaker at IEEE R10HTC recently.

He served or has been serving in the program committee of major software engineering conferences (e.g., ICSE, ICSME, SANER, MSR, ICPC and SCAM). He served as the Finance Chair for ICPC 2011, Tool Co-chairs for ICSM 2012 and WCRE 2012, Tool Chair for SCAM 2012, Poster Co-chair for ICPC 2012, Program Co-chair for IWSC 2012, Finance Chair for ICSM 2013, and General Chairs for ICPC 2014 and IWSC 2015. He has been serving as the program co-chair for ICPC 2018 and General Chair for SCAM 2019. He has been a regular reviewer of the major journals of the area including IEEE transactions in software engineering, Empirical Software Engineering, Journal of Systems and Software, Journal of Software: Evolution and Process, Science of Computer Programming, Information and Software Technology and so on.

Acknowledgment

I would like to thank the program co-chairs and steering committee members of IWSC 2018 for inviting me for the keynote talk. This research is supported by the Natural Sciences and Engineering Research Council of Canada (NSERC), and by a Canada First Research Excellence Fund (CFREF) grant coordinated by the Global Institute for Food Security (GIFS) at University of Saskatchewan.