

Guest editorial: special issue on realizing AI synergies in software engineering

Rachel Harrison · Tim Menzies

Received: 13 October 2014 / Accepted: 17 November 2014 / Published online: 14 January 2015
© Springer Science+Business Media New York 2015

Software engineering is now expected to solve a plethora of increasingly complex questions that are dynamic, automated, adaptive, or must execute on a very large scale.

In theory, AI technologies have the potential to support the development of increasingly complex SE systems. Conversely, in theory, SE might also play a role in alleviating development costs and the development effort associated with AI tools.

This special multi-issue presents some of the latest innovative synergistic AI and SE techniques and practices and was stimulated (in part) by a 2-day workshop entitled “Realizing AI Synergies in SE (RAISE’13)” which was held at ICSE’13 and was sponsored by the NSF. The workshop is part of an on-going series, usually held at the IEEE International Conference on Software Engineering. This first part includes papers covering topics related to defects and maintainability.

In “AutoODC: Automated Generation of Orthogonal Defect Classifications”, LiGuo Huang, Vincent Ng, Issac Persing, Mingrui Chen, Zeheng Li, Ruili Geng and Jeff Tian present a way to automate the classification of software defects by regarding it as a supervised text classification problem. The authors use the orthogonal defect classification (ODC) framework from IBM for their work and propose a novel annotation framework which uses experts’ opinions to help classify defects. The automated approach, called AutoODC, was trained using machine learning algorithms and the

R. Harrison (✉)
Department of Computing and Communication Technologies, Oxford Brookes University,
Oxford OX33 1HX, UK
e-mail: rachel.harrison@brookes.ac.uk

T. Menzies
Department of Computer Science, North Carolina State University, Raleigh, NC, USA
e-mail: tim@menzies.us; tim.menzies@gmail.com

result was evaluated using two defect reports, one from an industrial system and one from an open source system, with very high overall accuracy.

Spreadsheets are a part of everyday life for workers in the business sector, and faults in spreadsheets can cause colossal damage to those businesses that depend on them. The paper “On the Empirical Evaluation of Similarity Coefficients for Spreadsheets Fault Localization”, by Birgit Hofer, Alexandre Perez, Rui Abreu and Franz Wotawa discusses the application of spectrum-based fault localization techniques to the spreadsheet domain. The authors also discuss the impact of the proportion of correct cells on a diagnostic report. They suggest that spectrum-based fault localization performs well in the spreadsheet domain when compared to model-based diagnosis.

Continuing with the theme of software defects, the paper “Automatic, High Accuracy Prediction of Reopened Bugs”, by Xin Xia, David Lo, Emad Shihab, Xinyu Wang and Bo Zhou discusses how to predict which bugs will be reopened and proposes an automatic predictor for such bugs. The authors use three open source projects to validate the predictor and show that it outperforms prior work.

Software development companies have to spend a great deal of effort on maintenance and so any techniques for reducing maintenance effort are welcome. In “SMPLearner: Learning to Predict Software Maintainability”, Wei Zhang, LiGuo Huang and Vincent Ng present an automated approach to maintainability prediction. Their approach uses the reported maintenance effort obtained from the code change history and a set of code metrics. The approach was evaluated using observations from eight large scale open source software systems. The authors compare their approach with a 4-metric model and learning-based maintainability predictors and show that their approach outperforms the other approaches.

Please enjoy these contributions, and look for the rest of the special issue coming soon.