

Guest editorial: special issue on realising artificial intelligence synergies in software engineering

Rachel Harrison¹ · Ayşe Başar Bener² ·
Çetin Meriçli³ · Burak Turhan⁴ 

Received: 4 August 2017 / Accepted: 7 August 2017 / Published online: 21 August 2017
© Springer Science+Business Media, LLC 2017

The aim of realising artificial intelligence synergies in software engineering (RAISE) workshop series is to bring together academics and industrial practitioners to exchange and discuss the latest innovative synergistic artificial intelligence (AI) and software engineering (SE) techniques and practices. Mutually beneficial characteristics have appeared in the past few decades and are still evolving due to new challenges and technological advances. Hence, the question that motivates and drives the RAISE Workshop series is: Are SE and AI researchers ignoring important insights from AI and SE?

To pursue this question, RAISE 2015, the fourth in the series, explored not only the application of AI techniques to SE problems but also the application of SE techniques to AI problems. During the workshop, Prof. John Mylopoulos, University of Toronto, provided an excellent example of such exploration in his keynote titled “Knowledge Representation for Requirements Engineering, and Requirements Engineering for Intelligent Systems”. In addition, seven contributions by the workshop

✉ Burak Turhan
burak.turhan@brunel.ac.uk

Rachel Harrison
rachel.harrison@brookes.ac.uk

Ayşe Başar Bener
ayse.bener@ryerson.ca

Çetin Meriçli
cetin@cmu.edu

- 1 Oxford Brookes University, Oxford, UK
- 2 Ryerson University, Toronto, Canada
- 3 Carnegie Mellon University, Pittsburgh, PA, USA
- 4 Brunel University London, London, UK

participants were thoroughly discussed during the workshop. This special section results from an open call for contributions following on these themes. All submissions were peer-reviewed by experts in the field. Finally, two papers were selected for inclusion in this special section.

We would like to extend our sincere thanks to the authors for their contributions, reviewers for their invaluable assistance, and the Editor-in-Chief for making this special issue possible. We hope that you will enjoy reading these interesting contributions!

NLCI: A Natural Language Command Interpreter. Mathias Landhuer, Sebastian Weigelt, Walter F. Tichy. The method presented in this paper populates domain ontologies from given APIs and generates executable code, which calls the ontologised APIs, to complete action commands given in natural language. Evaluations in home automation, e.g., “Turn on the heaters in the living room and increase the volume of the radio”, and 3D animation domains yield promising results for future applications with 67% recall and 78% precision on 50 benchmark scripts. Authors find that the key to success for full and reliable automated code generation is in having descriptive names in API design.

From requirements to source code: a model-driven engineering approach for RESTful web services. Christoforos Zolotas, Themistoklis Diamantopoulos, Kyriakos C. Chatzidimitriou, Andreas L. Symeonidis. The model-driven engineering engine proposed in this paper generates REST-compliant and fully functional web services from textual functional requirements and visual storyboards. After three user study evaluations, authors report roughly 80% (22%) reduction in total effort excluding (including) the learning curve for using the engine. Most importantly, the tool is available for you to try at <http://s-case.github.io>.