Intelligent Software Engineering: Synergy between AI and Software Engineering

Tao Xie
University of Illinois at Urbana-Champaign
taoxie@illinois.edu

ABSTRACT

There has been a long history of applying AI technologies to address software engineering problems especially on tool automation. On the other hand, given the increasing importance and popularity of AI software, recent research efforts have been on exploring software engineering solutions to improve the productivity of developing AI software and the dependability of AI software. The emerging field of intelligent software engineering is to focus on two aspects: (1) instilling intelligence in solutions for software engineering problems; (2) providing software engineering solutions for intelligent software. This extended abstract shares perspectives on these two aspects of intelligent software engineering.

CCS CONCEPTS

• Computing methodologies → Artificial intelligence; **• Software and its engineering**;

KEYWORDS

Intelligent software engineering, artificial intelligence, software dependability

ACM Reference Format:
Tao Xie. 2018. Intelligent Software Engineering: Synergy between AI and Software Engineering. In *ISEC '18: Innovations in Software Engineering Conference, February 9–11, 2018, Hyderabad, India.* ACM, New York, NY, USA, Article 4, 1 page. https://doi.org/10.1145/3172871.3172891

1 INTRODUCTION

The emerging field of intelligent software engineering has two aspects of denotation: *intelligent* [software engineering] and [*intelligent software*] engineering. *Intelligent* [software engineering] (i.e., intelligence in software engineering) concerns about instilling intelligence in solutions for software engineering problems, with (but not limited to) AI technologies. [*Intelligent software*] engineering (i.e., software engineering for intelligent software) concerns about providing software engineering solutions for intelligent software. We next elaborate these two aspects with examples.

Intelligence in Software Engineering. There has been a long history of applying AI technologies to address software engineering problems [1], often with some common goals of making tool automation for software engineering more effective. In our perspective, we advocate the research community to step back (from just simply applying AI technologies) and explore instilling intelligence in software engineering solutions, with AI technologies and other technologies (such as program analysis). Example open research questions are how to define or determine levels of intelligence in software engineering solutions, how to bring high levels of intelligence in software engineering solutions, and how to synergically integrate machine intelligence and human intelligence (e.g., domain knowledge or insight) to effectively tackle challenging software engineering problems. In addition, we advocate the research community to investigate whether assumptions made by the proposed intelligent solutions are valid in software engineering practices, e.g., the representativeness of the training data used for machine learning is sufficient for real practices [7].

Software Engineering for Intelligent Software. Insufficient confidence on the safety and control of intelligent software limits the deployment scope of intelligent software in the real world [2]. In addition, security of AI software has been an increasing concern, motivating recent active research on adversarial machine learning [4, 5]. To assure dependability of intelligent software, software testing techniques have been commonly used in practice, yet where test oracles are a well-known challenge [3, 6].

ACKNOWLEDGMENT

This work was supported in part by National Science Foundation under grants no. CCF-1409423, CNS-1513939, CNS-1564274, and the Natural Science Foundation of China Grant No 61529201.

REFERENCES

[1] Mark Harman. 2012. The Role of Artificial Intelligence in Software Engineering. In *Proc. International Workshop on Realizing AI Synergies in Software Engineering (RAISE).* 1–6.

[2] National Science and Technology Council (NSTC). 2016. *Preparing for the Future of Artificial Intelligence.* Technical Report.

[3] Siwakorn Srisakaokul, Zhengkai Wu, Angello Astorga, Oreoluwa Alebiosu, and Tao Xie. 2018. Multiple-Implementation Testing of Supervised Learning Software. In *Proc. AAAI-18 Workshop on Engineering Dependable and Secure Machine Learning Systems (EDSMLS).*

[4] Wei Yang, Deguang Kong, Tao Xie, and Carl A. Gunter. 2017. Malware Detection in Adversarial Settings: Exploiting Feature Evolutions and Confusions in Android Apps. In *Proc. Annual Computer Security Applications Conference (ACSAC).* 288–302.

[5] Wei Yang and Tao Xie. 2018. Telemade: A Testing Framework for Learning-Based Malware Detection Systems. In *Proc. AAAI-18 Workshop on Engineering Dependable and Secure Machine Learning Systems (EDSMLS).*

[6] Wujie Zheng, Hao Ma, Michael R. Lyu, Tao Xie, and Irwin King. 2011. Mining Test Oracles of Web Search Engines. In *Proc. IEEE/ACM International Conference on Automated Software Engineering (ASE).* 408–411.

[7] Zexuan Zhong, Jiaqi Guo, Wei Yang, Tao Xie, Jian-Guang Lou, Ting Liu, and Dongmei Zhang. 2018. Generating Regular Expressions from Natural Language Specifications: Are We There Yet?. In *Proc. Workshop on NLP for Software Engineering (NL4SE).*

ISEC '18, February 9–11, 2018, Hyderabad, India

ACM ISBN 978-1-4503-6398-3/18/02.
https://doi.org/10.1145/3172871.3172891