Scoping Software Engineering for AI: The TSE Perspective

IN recent years, important advances in Artificial Intelligence (AI), and, in particular, in Machine Learning (ML), including Deep Learning (DL) and Large Language Models (LLMs), have caused a substantial increase of submissions to all Software Engineering (SE) venues (conferences and journals) related to SE with and for AI. They are commonly referred to as AI for SE and SE for AI.

On the one hand, AI techniques have been used to provide better solutions to problems with which software engineering researchers have struggled for a long time (e.g., code completion, fault localization, program repair, and test case generation), as well as solve problems for which automated solutions did not exist in the past, or were very limited, e.g., automated bug reproduction, code review, or the generation of complete, non-trivial program elements. Contributions along these lines are commonly described as "AI for Software Engineering" and are welcome at IEEE TRANSACTIONS ON SOFTWARE ENGINEERING (TSE). The questions of what constitutes novelty and significance of such papers are interesting and complex, and we will address them in a future editorial.

On the other hand, certain AI artifacts, e.g., ML models, can be seen as software components forming part of a more complex software system. Thus, the engineering of ML components *might be considered* to be of a core interest to SE. In fact, many top SE venues, including IEEE TSE, have been publishing a broad range of contributions on testing, verifying, repairing, understanding, and optimizing ML components, under the broad umbrella of "software engineering (SE) for AI".

That said, a matter of growing concern is the number of papers submitted under the category of SE for AI that provide weak arguments motivating the work by connecting it to SE problems, and ultimately bear little or no relation to SE.

Going forward, IEEE TSE would like to take a more nuanced approach with respect to reviewing these "SE for AI" papers. Specifically, we observe that submitted manuscripts frequently go into the depths of core AI techniques to improve them in various ways. The question that editors, reviewers, and authors themselves therefore often ask is whether some of the submitted manuscripts are a good fit for SE venues such as IEEE TSE, or would be a better fit for more AI- or ML-specialized venues instead.

The question of scoping SE with respect to other fields in computing is not new. A similar case can be made, for example, for network protocols, database management systems, or operating systems. There are intersections and common interests with the Networks, Databases, and OS research communities respectively, all of which ultimately deal with engineering software artifacts. Yet, certain types of contributions are better reviewed and more appreciated in the communities that deal with the specificities of networks, databases or operating systems.

The IEEE TSE call for contributions [1] states:

> *"IEEE Transactions on Software Engineering is interested in well-defined theoretical results and empirical studies that potentially impact the construction, analysis, or management of software. The scope of this Transactions ranges from the mechanisms through the development of principles to the application of those principles to specific environments."*

That is, the main focus of the submission and the research questions it addresses should be on software engineering, i.e., software engineering cannot be a tangential aspect of the paper.

However, the above call leaves for interpretation what an SE contribution means in the growing area of SE for AI, and we describe our views on this below.

Let us consider aspects of contributions that make submissions more relevant to IEEE TSE.

1) The work's contribution **concerns a software system as a whole, or a subsystem, and not simply its AI or ML component**. For example, on the one side, we will consider manuscripts proposing approaches for testing an ML-intensive software subsystem, like the perception subsystem of a self-driving car, to be highly relevant to IEEE TSE. Conversely, focusing solely on evaluating or improving the inference capability of the ML pedestrian recognition model is a contribution that is more appropriate for an ML-specific or a computer vision venue.
2) The manuscript **considers software engineering artifacts** (e.g., requirements, design, architecture) beyond an ML component and its standard constituents, such as its hyperparameters, or datasets for training, testing and evaluation. For example, repairing an ML model while considering its requirements specification, or in the context of the software architecture that incorporates the component under repair, would be considered very relevant. Yet, the repair process that eliminates one incorrectly classified input would be significantly less relevant to IEEE TSE and would be more appropriate for an AI- or ML-specific venue.
3) The proposed contribution is targeting a **novel context for an SE task** (such as requirements elicitation or

program repair) where ML models are prominent. Such a paper would then need to argue that (1) it is tackling a problem that is relevant to the ML community and is not adequately solved by such community; and (2) the existing state-of-the-art SE approaches do not straightforwardly solve the ML problem either. Examples of novel contexts for SE tasks include inferring specifications from ML models, or repairing and modularizing them. Note the difference between a task and a technique. SE uses a large toolbox of techniques from a range of fields, from static analysis to model checking, to support different tasks such as debugging, program comprehension, and software design. Simply applying a technique from the SE toolbox to an ML problem (e.g., the direct use of a statistical model checking technique to analyze an ML model) most likely does not advance the SE state-of-the-art. On the other hand, using statistical model checking for an SE task, such as adapting an ML model by repairing it instead of doing so via standard model retraining, directly contributes to SE.

4) The work studies **human, social, socio-technical, and organizational aspects** (e.g., as developer, user, stakeholder characteristics and experiences) **in the development of AI-intensive software systems**. An example of a relevant contribution is an exploration of the collaboration challenges occurring in the interaction between data scientists and software engineers across their respective teams or within a multi-disciplinary team (e.g., during requirements engineering, or as part of exploration and discovery in agile development).

To summarize, the boundaries between SE and AI are, and certainly will be, blurred. Yet, establishing core interests of each research community will help ensure that future manuscripts submitted to IEEE TSE will, on the one hand, receive high-quality and timely reviews, and, on the other hand, be appreciated by our readers[1].

In describing our current vision on SE for AI, which may shift in the future given how dynamic the field is, we do so not in the spirit of settling the discussion. Rather, we would like to start a community-wide conversation, which may, or may not, converge toward a single vision. It is possible that other SE venues would have a wider or a narrower scope.

We look forward to your contributions!

Sebastian Uchitel
Universidad de Buenos Aires
Argentina
e-mail: suchitel@dc.uba.ar

Marsha Chechik
Department of Computer Science
University of Toronto
Canada

Massimiliano Di Penta
Department of Engineering
University of Sannio
Italy

Bram Adams
Queen's University
Canada

Nazareno Aguirre
Departamento de Computación
Universidad Nacional de Río Cuarto
Argentina

Gabriele Bavota
Faculty of Informatics
Universita della Svizzera
Switzerland

Domenico Bianculli
University of Luxembourg
Luxembourg

Kelly Blincoe
University of Auckland
New Zealand

Ana Cavalcanti
University of York
United Kingdom of Great Britain and Northern Ireland

Yvonne Dittrich
IT University of Copenhagen
Denmark

Filomena Ferrucci
Dipartimento di Informatica
Università di Salerno, Italy

Rashina Hoda
Faculty of Information Technology
Monash University
Melbourne, Australia

LiGuo Huang
Department of Computer Science and Engineering
Southern Methodist University
Dallas, TX 75205 USA

David Lo
Singapore Management University
Singapore

Michael R. Lyu
Department of Computer Science and Engineering
The Chinese University of Hong Kong
Hong Kong

Lei Ma
Department of Electrical and Computer Engineering
University of Alberta
Canada

[1]Please note that this editorial position supersedes any claims of scope of submitted manuscripts based on prior publication venues of related work.

Jonathan I. Maletic
Kent State University
OH, USA

Leonardo Mariani
University of Milano - Bicocca
Milan, Italy

Collin McMillan
Department of Computer Science
University of Notre Dame
USA

Tim Menzies
Department of Computer Science
North Carolina State University
USA

Martin Monperrus
KTH Royal Institute of Technology
Sweden

Ana Moreno
Universidad Politécnica de Madrid
Madrid, Spain

Nachiappan Nagappan
Meta Platforms Inc.
USA

Liliana Pasquale
University College Dublin
Ireland

Patrizio Pelliccione
Department of Computer Science
Gran Sasso Science Institute (GSSI)
Italy

Michael Pradel
University of Stuttgart
Stuttgart, Germany

Rahul Purandare
University of Nebraska–Lincoln
USA

Sukyoung Ryu
Department of Computer Science
KAIST
Republic of Korea

Mehrdad Sabetzadeh
University of Ottawa
Canada

Alexander Serebrenik
Eindhoven University of Technology
The Netherlands

Jun Sun
Singapore Management University
Singapore

Kla Tantithamthavorn
Department of Information Technology
Monash University
Australia

Christoph Treude
Singapore Management University
Singapore

Manuel Wimmer
Johannes Kepler Universitat Linz
Austria

Yingfei Xiong
School of Computer Science
Peking University
China

Tao Yue
Beihang University
China

Andy Zaidman
Delft University of Technology
The Netherlands

Tao Zhang
Faculty of Information Technology
Macau University of Science and Technology
Macao

Hao Zhong
Department of Computer Science and Engineering
Shanghai Jiao Tong University
China

Acknowledgment

The authors thank Matthew Dwyer, Sebastian Elbaum and Mark Harman for their feedback on earlier versions of this editorial.

Reference

[1] "Scope of IEEE Transactions on Software Engineering," Accessed: Oct. 14, 2024. [Online]. Available: https://www.computer.org/csdl/journal/ts/about/14409?title=About&periodical=IEEE