Software and Systems Modeling (2023) 22:1391–1392
https://doi.org/10.1007/s10270-023-01126-0

EDITORIAL

Large language models as an “operating” system for software and systems modeling

Benoit Combemale[1] · Jeff Gray[2] · Bernhard Rumpe[3]

Published online: 16 September 2023

Large Language Models (LLM), such as the variants of ChatGPT or BERT, are currently under intensive development and enhancement, but it has become clear that they will contribute a significant change to the way text and images are created in the future. It is therefore not surprising that the challenges, risks and opportunities of generative AI are discussed intensively in various scientific communities as well as in industry, government, education, and society, in general. There are of course technological aspects to be understood, but the impact on society and the industrial world has been significantly changed by the advantages and consequences of an expanded focus of generative AI, as supported by LLMs.

Although there is much to learn from the emergence of this technology, it is becoming more evident how the usage of LLMs may impact the way we develop software and software intensive systems. We assume that many SoSyM readers have made their own experiments to understand the capabilities and limitations of the currently available LLMs. Similar to any other complex system, it is challenging to understand analytically what the system does and why it produced a specific output. In particular, the traditional computer science formal and precise analytical methods often do not apply to the use of LLMs. Thus, we may need to rely on experimentation to gain initial insights into the capabilities and limitations of the potential for using LLMs in our own work. For a deeper validation of these insights and the hypotheses that will be built in the near future, it may make sense to (among other techniques) **apply psychological methods to understand AI systems**. Psychology has understood how to analyze brain capabilities and how to deal with uncertainty based on statistical evidence. Such statistical evidence may not be enough when a general AI system is used in safety critical applications (e.g., autonomous driving), but it may be helpful when trying to understand the usefulness of the conversation with such an AI, and how the AI is generally thinking. It may also be helpful to complement the analytical approaches to cope with uncertainty in the modeling activities.

In this editorial, we pose another hypothesis on the future use of LLMs. The hypothesis is:

1. Variations of LLMs will emerge, with some very specialized to a distinct task or a specific domain, which focus on a particular underlying knowledge that is constrained by technical or societal concerns.
2. However, it will not be the case that there will be a unique LLM developed and trained for each purpose. Instead, there will be individual post-training of an underlying LLM that allows configuration of an AI model in such a way that it can be adapted easily to the individual functions that need assistance.
3. There will only be a few fully trained core LLMs and they will be provided in an open form. They will act like a kind of “operating system” for AI-systems where “application”-specific extensions are created.

The first two parts of this hypothesis are not very far-fetched, because we can already see first approaches in this direction. This would allow for the training of an LLM for a specific modeling language that is tied to a particular target infrastructure for a distinct domain. This could conceivably be accomplished with relatively little training data and support very helpful assistance when developing, evolving or adapting a software system. This concept would allow the developer to think, work and handle all concerns at the modeling level while completely deferring the implementation to an automated generator. Thus, traditional software

✉ Bernhard Rumpe
bernhard.rumpe@sosym.org

Benoit Combemale
benoit.combemale@sosym.org

Jeff Gray
jeff.gray@sosym.org

1 University of Rennes, Rennes, France

2 University of Alabama, Tuscaloosa, AL, USA

3 RWTH Aachen University, Aachen, Germany

models would become the intermediate step between informal explanations and a detailed implementation that is not human-understandable, but can be analyzed by humans based on the abstract model-based description.

It may be that a **general-purpose LLM** serves as a kind of "operating system" that is only customized with individual knowledge for a variety of different activities. This will help humans in various domains and developers in their particular tasks. It may be necessary for researchers to understand what kinds of modeling languages the LLM should embody and how much needs to be known about the semantic foundations. The standard approach from tool developers that leads to over-generalization and rather semantic-agnostic modeling concepts could then be extended by very specific techniques that either have their roots in the domain, the sponsoring company, or even the project itself.

We look forward to seeing LLMs that are applied to modeling tasks by using specific, semantically sound modeling languages and supporting environments that provide efficient developer assistance as discussed in a previous editorial [CGR23b] (available at https://www.sosym.org/editorials/).

[CGR23b] B. Combemale, J. Gray, and B. Rumpe. ChatGPT in software modeling In: Journal Software and Systems Modeling (SoSyM), Springer Berlin/Heidelberg, ISSN 1619–1366. Volume 22(3), Pages 777–779, 2023.

1 Content of this Issue

1. MODELS 2021 Special Section
Guest editors: Shiva Nejati and Daniel Varro

Funding Open Access funding enabled and organized by Projekt DEAL.