AI Engineering: Realizing the Potential of AI

Jan Bosch, Software Center and Chalmers University of Technology

Helena Holmström Olsson, Software Center and Malmö University

Björn Brinne, Peltarion

Ivica Crnkovic, Chalmers Artificial Intelligence Research Center and Chalmers University of Technology

Digital Object Identifier 10.1109/MS.2022.3199621
Date of current version: 24 October 2022

OVER THE LAST decade, and increasingly so in the last few years, interest in artificial intelligence (AI) has grown exponentially not only in industry and research but also in all areas of modern society. Many claim that AI is revolutionary and will change everything, and already now, AI technologies are rapidly changing not only system performance and capabilities but also the ways in which systems are developed and how they evolve over time. The successful deployment of machine learning/deep learning (ML/DL) models requires engineering solutions around them to ensure production quality and industry-strength deployment, that is, we need AI engineering.

The Challenges of Deploying AI Projects

Although some view AI as an evolution of traditional computer science (see also the "point/counterpoint" section of this special issue), as editors of the special issue, we view the development of ML-powered systems to be different from other software development in that the resulting program and its ability to solve a given task depend very much on the data and the examples it is trained on. This means that when you initiate a project, it is nigh impossible to say whether a satisfactory solution can be found within given resource limits. To make matters worse, the skill sets needed in this discovery or the R&D phase are not the same as for software development, but instead, R&D is usually done by data scientists who, when they have created a working prototype, need to hand it over to the engineers who will build the production system. Oftentimes, there are different coding practices, different frameworks allowed, and so on, and even though in principle it should be possible to replicate the results of the prototype with a new implementation, it is often quite difficult to do that—frameworks are not 100% deterministic, new bugs are introduced in the redevelopment, and so on.

However, when an organization is able to overcome these obstacles and others that are presented in the AI engineering literature, great things can be accomplished. There are, of course, the incredible results of state-of-the-art AI research like AlphaFold or recent Internet hits like GPT-3, DALL·E, and similar generative models as well as high-profile AI applications like autonomous transportation or AI-powered robotics. But also, in more mundane applications, ML is making a lot of progress: personalization of consumer services; less waste and better predictability in industry; more efficient tools in administrative tasks; more accurate search applications; fast and accurate medical diagnostics support; and much more. In all of these latter applications, it is of utmost importance to get the engineering sorted in an efficient and reliable manner. The applications may not always be as fanciful, but this is where society and industry will get true value from the use of AI, and hence, it is of absolute importance to create practices and tooling that support it.

When looking at the state of practice and the extent to which AI-enabled systems are deployed in practice, still only a smaller number of companies are really mastering the technology and using it for continuous development and improvement of their products. Despite large initiatives and investments, the vast majority of companies are in the starting phases of AI deployment and have not yet managed to go beyond the initial prototyping and experimentation stages. In industry, most of AI today is related to ML and DL, which is AI based on automated analysis of, and generalization from, data that are either collected or generated. If looking at state-of-the-art research on topics related to AI, the primary focus is on obtaining efficient computation or understanding the AI modeling principles (such as explainable AI), and there are many promising prototypes using AI technologies.

Unfortunately, our research[1,2,3] shows that the transition from the prototyping and experimentation stages to the production-quality deployment of ML/DL models proves to be a significant challenge for many companies. Though not recognized by everyone, the engineering challenges surrounding ML/DL model deployment are exceptional.

Typically, the first problem companies face is a shortage in AI skills and expertise. And even if companies are equipped with skilled AI experts, these alone are not sufficient for building highly complex, software-intensive, and AI-enabled systems that scale in domains that might be subject to safety-critical regulations. Instead, there is a need for interdisciplinary teams that include AI expertise as well as data science, domain knowledge, and in particular, software engineering (SE) expertise.[4,5]

Beyond the problem related to skills and expertise, companies face a number of additional challenges. Throughout the development process of AI-enabled systems, the main challenge is not to develop the best model or algorithm but to build a process that provides support for the entire lifecycle of a system[1,6] from a business idea; the collection and management of data; software development in which both code and data dependency must be under management control and in which a large amount of code is glue code;[7] product deployment and operation; and its continuous evolution. The need for specific

support of SE for AI, that is, support for managing the entire lifecycle from idea to industry-strength operation and evolution, has been identified in the SE community[8,9] as well as in the AI industry.[1,10]

Overview of the Special Issue Articles

The aim of this special issue is to provide a venue for sharing practical experiences and research results on the new challenges that are emerging in SE and that AI/data science engineers and software engineers are facing in the development of AI-enabled software systems (that is, systems that include AI components and functions). The special issue includes five articles and a "point/counterpoint" discussion. Next, we give a brief overview of the content of the special issue.

The article by Rodríguez-Mancini et al. addresses the challenge that we already raised earlier in this discussion: many AI-enabled systems experience significant amounts of glue code and what the authors refer to as *tech stack sprawl*. This can lead to high levels of technical debt in the system, which, in time, tends to cause high maintenance and evolution cost. As a solution, the authors propose a fractal approach to modeling data processing pipelines where each data processing unit is a composition of lower level data processing elements. In addition, the authors present a three-step process to develop fractal data processing pipelines.

The article by Sen et al. addresses the problem of data quality that, in our experience, virtually all AI-enabled systems struggle with. The authors specifically focus on the domain of industrial Internet of Things (IoT) systems, covering the entire scope from edge to cloud. The authors use ML/DL models both to identify erroneous data in unsupervised data pipelines as well as to repair erroneous data to ensure the proper operation of systems. As most data in industrial IoT systems tend to follow a periodic pattern, the authors make use of time series models that are first trained on normal data and then used to detect and repair erroneous data.

The article by Vaidhyanathan et al. focuses on the process aspects of developing ML-enabled systems. The authors recognize that although practitioners (as well as the academic research community) have been working on identifying and solving challenges related to, for example, the development, deployment, and testing of ML-enabled systems, there is little work on methodology and how to benefit from agile practices in the development of ML-enabled systems. To address this gap, and based on the experiences from a small, medium-sized enterprise developing a computer vision-based solution, the authors present the Agile4MLS process. As shown in the article, this process enables agility; fosters collaboration; helps practitioners manage uncertainties; and increases the release frequency in the development of ML-enabled systems.

The article by Sagodi et al. recognizes how engineering AI-enabled systems goes beyond merely building algorithms. The authors emphasize how building industrial solutions, including AI components, requires algorithms to be embedded into mature and complex products and how this poses novel challenges for software engineers. In the article, the authors illuminate how interdisciplinary collaboration; definition and agreement on AI added value; and expectation management are vital for successful AI engineering. Based on use cases and significant domain experience, the authors provide an overview of challenges and solutions in engineering AI-enabled systems in the context of manufacturing, and they highlight the need for an integrated interplay among AI, data, and the specifics of the domain.

The article by Nili et al. addresses the lack of guidance for managers on how to manage uncertainties associated with the deployment of AI technologies. The authors draw on their experiences in the public sector to identify challenges that managers typically face; to describe patterns related to what AI deployment uncertainties are; and to provide strategies for how managers can tackle these challenges. The authors recognize that although contextual and cultural particularities might make some recommendations more or less effective, there are indeed recommendations that are relevant for all. As part of the actionable insights the article provides, the authors emphasize how the successful deployment of AI technologies involves not only the technologies but also the transformation of people, processes, and data-related regulations that support new information-driven business models.

We are grateful for and excited about the "point/counterpoint" discussion provided by Mary Shaw and Liming Zhu. They discuss whether AI engineering should be viewed as a significant novel and radically different field or if it is an evolution of SE as we have studied it for decades. Although we as guest editors take the position that AI engineering brings with it a set of novel engineering challenges that were not present in traditional SE, it is important to discuss where SE can provide solutions also to AI-enabled systems.

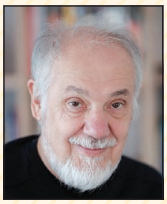
As a short reflection on the future of AI, as guest editors, we see two interesting trends in the companies that

ABOUT THE AUTHORS

JAN BOSCH is the director of Software Center (www.software-center.se), a strategic collaboration between 16 international companies, including Ericsson, Volvo, Siemens, and Robert Bosch GmbH, and five universities with the mission of accelerating the digital transformation of the European software-intensive systems industry, and a professor at Chalmers University of Technology, Gothenburg 41296, Sweden. Contact him at jan.bosch@chalmers.se.

HELENA HOLMSTRÖM OLSSON is a principal investigator/senior researcher at Software Center, and a professor at Malmö University, Malmö 20506, Sweden. Contact her at helena.holmstrom.olsson@mau.se.

BJÖRN BRINNE is the chief artificial intelligence officer at Peltarion, Stockholm 11160, Sweden. Contact him at bbrinne@gmail.com.



IVICA CRNKOVIC was the director of the Chalmers Artificial Intelligence Research (CHAIR) Center, and a professor of software engineering at Chalmers University of Technology, Gothenburg 41296, Sweden, until his unexpected and sad passing earlier in 2022.

we work with, that is, from centralized to decentralized and from offline to online training. Traditional AI approaches tend to be centralized in that all data are brought to a central location for training the models, and the resultant model is then distributed for operation. Although this traditional model has resulted in wonderful innovations, it is also very expensive in terms of infrastructure; complicates working with sensitive data; and results in a one-size-fits-all ML/DL model.

As we work a lot with the embedded systems industry, where companies typically have hundreds, thousands, if not millions of devices in the field, we see a clear interest in using these devices in federated setups as the computational resources in these devices are not always fully engaged. Also, for suitable use cases, we see experiments with online training models, such as reinforcement learning, bringing the training from offline to online. The combination of federated and reinforcement learning addresses some of the key challenges associated with centralized and offline approaches, and hence, we believe that AI engineering will need to focus on supporting these new approaches to employing AI as well.

Finally, in the process of proposing and defining the special issue, our dear friend and colleague, Ivica Crnkovic, passed away unexpectedly. As close colleagues, and in this context, part of the guest editor team, we are all still coming to terms with this incredible, very sad loss for us as Ivica's friends and close colleagues as well for the SE community at large. At his funeral, where we had the privilege to be present, the priest described Ivica as a person "larger than life." This is an apt description for all of us who had the opportunity to work with him. His kind, friendly nature combined with a strong research direction and ambition made him a treasure. Ivica was instrumental in shaping the AI engineering community, and this special issue is part of his legacy.

References

1. A. Arpteg, B. Brinne, L. Crnkovic-Friis, and J. Bosch, "Software engineering challenges of deep learning," in Proc. *44th EUROMICRO Conf. Softw. Eng. Adv. Appl.*, Prague, Czech Republic, 2018, pp. 50–59, doi: 10.1109/SEAA.2018.00018.
2. L. E. Lwakatare, A. Raj, J. Bosch, H. H. Olsson, and I. Crnkovic, "A taxonomy of software engineering challenges for machine learning systems: An empirical investigation," in *Proc. Int. Conf. Agile Softw. Dev.*, Cham: Springer-Verlag, May 2019, pp. 227–243.
3. A. Raj, J. Bosch, H. H. Olsson, A. Arpteg, and B. Brinne, "Data management challenges for deep learning," in *Proc. EUROMICRO*

Conf. Softw. Eng. Adv. Appl., Kallithea, Greece, 2019, pp. 140–147, doi: 10.1109/SEAA.2019.00030.
4. L. Bernardi, M. Themistoklis, and P. Estevez, "150 successful machine learning models: 6 Lessons learned at Booking.com," in *Proc. 25th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, ACM, 2019, pp. 1743–1751, doi: 10.1145/3292500.3330744.
5. J. Bosch, H. H. Olsson, and I. Crnkovic, "It takes three to tango: Requirement, outcome/data, and AI driven development," in *Proc. Int. Workshop Softw.-Intensive Business,* Helsinki, Finland, Dec. 2018, pp. 186–263.
6. D. Wang et al., "Human-AI collaboration in data science: Exploring data scientists' perceptions of automated AI," *Proc. ACM Hum.-Comput. Interact.*, vol. 3, no. CSCW, pp. 1–4, Nov. 2019, doi: 10.1145/3359313.
7. D. Sculley et al., "Hidden technical debt in machine learning systems," in *Proc.* Adv. *Neural Inf. Process. Syst.* (NIPS), C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, Eds., Curran Associates, 2015, pp. 2503–2511.
8. T. Menzies, "The five laws of SE for AI," *IEEE Softw.*, vol. 37, no. 1, pp. 81–85, Jan./Feb. 2020, doi: 10.1109/MS.2019.2954841.
9. I. Ozkaya, "What is really different in engineering AI-enabled systems?" *IEEE Softw.*, vol. 37, no. 4, pp. 3–6, Jul./Aug. 2020, doi: 10.1109/MS.2020.2993662.
10. M. Kim, T. Zimmermann, R. DeLine, and A. Begel, "Data scientists in software teams: State of the art and challenges," *IEEE Trans. Softw. Eng.*, vol. 44, no. 11, pp. 1024–1038, Nov. 2018, doi: 10.1109/TSE.2017.2754374.

Digital Object Identifier 10.1109/MS.2022.3210314