Research Inventy: International Journal of Engineering And Science
Vol.13, Issue 7 (July 2023), PP 97-101
Issn (e): 2278-4721, Issn (p):2319-6483, www.researchinventy.com

Software Development for AI-Powered Systems: An Overview

Rati Ranjan Sahoo [1], Sunil Kumar Panigrahi [2], Chittaranjan Sahoo[3]
[1,2,3] *Dept. of CSE, Einstein Academy of Technology and Management, Bhubaneswar*

Abstract
Systems that use artificial intelligence (AI) have at least one AI component to enable certain functionality, such as autonomous driving, speech and image recognition, and image processing. Advances in AI are leading to the widespread adoption of AI-based systems in society. But there isn't much synthesised information available about Software Engineering (SE) techniques for creating, managing, and sustaining AI-based systems. A thorough mapping study was carried out in order to gather and examine the most recent knowledge regarding SE for AI-based systems. Between January 2010 and March 2020, 248 studies were published. In the field of emergent research, more than two thirds of papers on SE for AI-based systems have been published since 2018. Reliability and safety are two of the most researched aspects of AI-based systems. Several SE techniques for AI-based systems were found, and we categorized them using the SWEBOK areas. Research pertaining to Software quality and testing are highly visible, but software maintenance seems to be overlooked. Problems with data are the most frequent difficulties. Researchers can immediately comprehend the state-of-the-art and determine which of our findings are valuable. Subjects in need of additional study; practitioners need educated on the strategies and difficulties that SE presents for AI-based systems and to close the curriculum gap between AI and SE, educators.
Keywords: *AI-based systems, systematic mapping study, SE4AI*

I. INTRODUCTION

In the last decade, increased computer processing power, larger datasets, and better algorithms have enabled advances in Artificial Intelligence (AI) [11]. Indeed, AI has evolved towards a new wave, which Deng calls "the rising wave of Deep Learning" (DL) 1 . DL has become feasible, leading to Machine Learning (ML) becoming integral to many widely used software services and applications [6]. For instance, AI has brought a number of important applications, such as image- and speech-recognition and autonomous, vehicle navigation, to near-human levels of performance [11]. The new wave of AI has hit the software industry with the proliferation of AI-based systems integrating AI capabilities based on advances in ML and DL [6]. AI-based systems are software systems which include AI components. These systems learn by analyzing their environment and taking actions, aiming at having an intelligent behaviour. As defined by the expert group on AI of the European Commission, "AI-based systems can be purely software-based, acting in the virtual world (e.g. voice assistants, image analysis software, search engines, speech and face recognition systems) or AI can be embedded in hardware devices (e.g. advanced robots, autonomous cars, drones or Internet of Things applications)"2 . Building, operating, and maintaining AI-based systems is different from developing and maintaining traditional software systems. In AI-based systems, rules and system behaviour are inferred from training data, rather than written down as program code . AI-based systems require interdisciplinary collaborative teams of data scientists and software engineers [6]. The quality attributes for which we need to design and analyze are different . The evolution of AI-based systems requires focusing on large and changing datasets, robust and evolutionary infrastructure, ethics and equity requirements engineering. Without acknowledging these differences, we may end up creating poor AI-based systems with technical debt . In this context, there is a need to explore Software Engineering (SE) practices to develop, maintain and evolve AI-based systems. This paper aims to characterize SE practices for AI-based systems in the new wave of AI, i.e., Software Engineering for Artificial Intelligence (SE4AI). The motivation of this work is to synthesize the current SE knowledge pertinent to AI-based systems for: researchers to quickly understand the state of the art and learn which topics need more research; practitioners to learn about the approaches and challenges that SE entails when applied to AI-based systems; and educators to bridge the gap among SE and AI in their curricula. Bearing this goal in mind, we have conducted a Systematic Mapping Study (SMS) considering literature from January 2010 to March 2020. The reason to focus on the last decade is that this new wave of AI started in 2010, with industrial applications of DL for large-scale speech recognition, computer vision and machine translation.

II. BACKGROUND

AI4SE. Recently, Perkusich et al. [15] referred to AI4SE as intelligent SE and defined it as a portfolio of SE techniques, which "explore data (from digital artifacts or domain experts) for knowledge discovery, reasoning, learning, planning, natural language processing, perception or supporting decision-making". AI4SE has developed driven by the rapid increase in size and complexity of software systems and, in consequence, of SE tasks. Wherever software engineers came to their cognitive limits, automatable methods were the subject of research. While searching for solutions, the SE community observed that a number of SE tasks can be formulated as data analysis (learning) tasks and thus can be supported, for example, with ML algorithms. SE4AI. First applications of SE to AI were limited to simply implementing AI algorithms as standalone programs, such as the aforementioned Mark 1 perception. As AI-based software systems grew in size and complexity and as its practical and commercial application increased, more advanced SE methods were required. The breakthrough took place when AI components became a part of established software systems, such as expert systems, or driving control. It quickly became clear that, because of the specific nature of AI (e.g., dependency on learning data), traditional SE methods were not suitable anymore (e.g., leading to technical debt [17]). This called for revision of classical, and development of new, SE paradigms and methods.

Related work on SE4AI

Several secondary studies in the broad area of SE4AI have been published so far (see Table 1). Masuda et al. conducted a review to identify techniques for the evaluation and improvement of the software quality of ML applications. They analyzed 101 papers and concluded that the field is still in an early state, especially for quality attributes other than functional correctness and safety. Washizaki et al. conducted a multivocal review to identify architecture and design patterns for ML systems. From 35 resources (both white and grey literature), they extracted 33 unique patterns and mapped them to the different ML phases. They discovered that, for many phases, only very few or even no patterns have been conceptualized so far. Serban and Visser performed both a case study and a systematic literature review on the topic of software architecture for machine learning. They reviewed 42 studies and performed 10 semi-structured interviews with practitioners from 10 different organisations. In their paper, the authors report 20 challenges and potential solutions. On a similar topic, John et al. , performed a systematic review of both scientific (13 studies) and grey literature (6 studies) on the topic of deployment of ML systems. They report a total of 27 challenges and 52 practices. Lorenzoni et al. performed a systematic literature review on the topic of development of machine learning systems. They analysed 33 studies between 2010 and 2020 and classified 10 issues and 13 solutions into seven SE practices. A number of reviews have been conducted in the area of software testing. Borg et al. [23] performed a review of verification and validation techniques for deep neural networks (DNNs) in the automotive industry. From 64 papers, they extracted challenges and verified them with workshops and finally a questionnaire survey with 49 practitioners. They conclude, among other challenges, that a considerable gap exists between safety standards and nature of contemporary ML-based systems. Another study was published by Ben Braiek and Khomh [. In their categories: realism of test input data (5 papers), adequacy of test criteria (12 papers), identification of behavioural boundaries (2 papers), scenario specification and design (3 papers), oracle (13 papers), faults and debugging (8 papers), regression testing (5 papers), online monitoring and validation (8 papers), cost of testing (10 papers), integration of ML models (2 papers), and data quality assessment (2 papers). Similarly, Zhang et al. Surveyed the literature on ML testing and selected 138 papers. From these, they summarized the tested quality attributes (e.g. correctness or fairness), the tested components (e.g. the data or the learning program), workflow aspects (e.g. test generation or evaluation), and application contexts (e.g. autonomous driving or machine translation) review of testing practices for ML programs, they selected a total of 37 primary studies and extracted challenges, solutions, and gaps. The primary studies were assigned to the categories of detect errors in data (five papers), in ML models (19 papers), and in the training program (13 papers). Riccio et al. Extracted testing challenges from 70 primary studies and propose the following.

III. RESEARCH METHODOLOGY

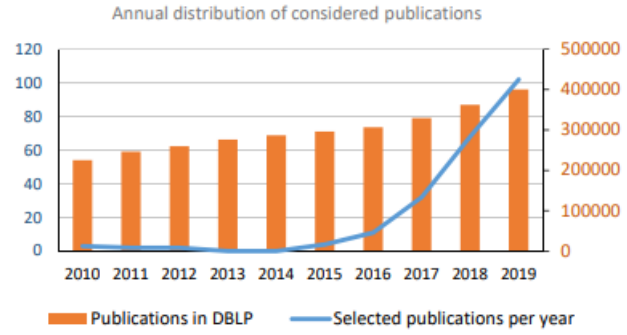
This SMS has been developed following the guidelines for performing SMSs from Petersen et al. [19]. Additionally, the guidelines for systematic literature reviews provided by Kitchenham and Charters [10] have also been used when complementary. This is because the method for searching for the primary studies, and taking a decision about their inclusion or exclusion is very similar between a systematic literature review and an SMS. This SMS has five main steps [11, 18] described in the following subsections.

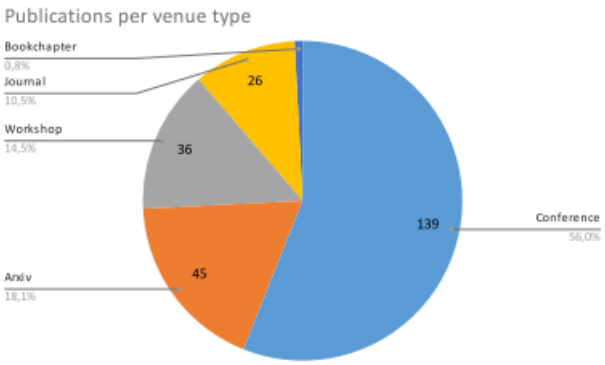
The aim of this SMS consists of identifying the existing research of SE for AI-based systems by systematically selecting and reviewing published literature, structuring this field of interest in a broader, quantified manner. We define the goal of this SMS formally using GQM [17]: Analyze SE approaches for the purpose of structure and classification with respect to proposed solutions and existing challenges from the viewpoint of both SE researchers and practitioners in the context of AI-based systems in the new wave of AI. To further refine this objective, we formulated four Research Questions (RQs) to be answered by our primary

studies: RQ1. How is SE research for AI-based systems characterized? RQ2. What are the characteristics of AI-based systems (used terms, scope, and quality goals)? RQ3. Which SE approaches for AI-based systems have been reported in the scientific literature? RQ4. What are the existing challenges associated with SE for AI-based systems?

Data extraction and mapping process

Each of the 248 primary studies was assigned to a single researcher for extraction based on the predefined data extraction form. Extractors were in frequent asynchronous contact to discuss potential inconsistencies. The weekly project meeting was also used for synchronization on this matter. In these meetings, we discussed the most persistent conflicts and shared our way of working as well as emerging findings to ensure cohesion of the team and to minimize subjectivity. Additionally, the data collection form includes the name of the reviewer and space for additional notes. This enabled us also to keep track of this process. After completing the extraction, we started data analysis and synthesis. In general, we performed both quantitative and qualitative analyses to classify the extracted data. During this process, additional extraction inconsistencies and mistakes were discovered and easily resolved in direct communication with the original extractors. Synthesis per RQ was performed by groups of at least two researchers, who often re-read (parts of) the original papers for this and kept the group in the loop. Final results were presented to the rest of the team and feedback was incorporated. Regarding the required mapping and analysis to answer the RQs, we performed the following activities in each RQ: For the analysis of RQ1, we used the assessed rigor and relevance quality [3] and performed frequency analysis to additionally determine bibliographical data such as the annual publication trend, venue types, authors' affiliations, geography distribution, and the empirical research type of the primary studies. To answer RQ2, we counted the number of occurrences of various terms related to AI used to characterize the study objects. We then used inductive coding to distil dimensions (Scope, Application Domain, and Technologies of AI) to describe the study objects, coded all primary studies, and reported frequencies. For the key quality attribute goals of AI-based systems, this also included a harmonization of the used terms as well as axial coding to cluster the identified quality properties.





Data validity

Data validity threats can be identified in the steps 4 and 5 of our SMS: keywording, and data extraction and mapping process. During the process of data extraction, subjective bias may lead to the misclassification of data or an inconsistent interpretation of the extracted data by the researchers. To mitigate these risks, we piloted the data extraction form, conducted weekly meetings with all the researchers, and discussed potential issues related to data extraction. All found issues were discussed among all researchers and decisions were documented to ensure that all researchers followed consistent data extraction and synthesis criteria. Nonetheless, apart from the three studies used for piloting, each paper was extracted by a single researcher. While many extractions were

fairly objective (e.g., a paper either explicitly described threats to validity or not, a paper used a specific term for AI-based system or a quality attribute goal, etc.), others left more room for interpretation. However, we argue that complete agreement is neither attainable for a sufficiently complex extraction process with multiple researchers nor is it strictly necessary, since we are very confident in the general tendencies and take-aways based on the extracted and synthesized data. Furthermore, as explained in Section 3, we performed qualitative analysis through an existing conceptual framework (SWEBOK). We iterated on initial classifications among all researchers in our weekly meetings, leading to some proposals to update this framework. Lastly, we need to mention that we adopted an inductive approach to the coding of properties. During the data extraction and mapping process, we e.g. extracted quality attribute goals and then grouped similar terms into unique codes. Including such terms explicitly in the search string may have produced slightly different results. Overall, we are confident that snowballing led to valid general tendencies in our sample, even though we do not claim completeness.

IV. CONCLUSION

In this paper, we surveyed the literature for software engineering for artificial intelligence (SE4AI) in the context of the new wave of AI. In the last ten years, the number of papers published in the area of SE4AI has strongly increased. There were almost no papers up to 2015 while afterwards, we saw a strong increase to 102 in 2019. The share of more than 18% on arXiv shows the "hotness" of the topic, but also emphasizes that literature reviews need to take arXiv into account. Furthermore, most articles are from a purely academic context, but 20% of publications with only industry authors show the importance for practice. When we look at the countries of the authors, the United States play in a separate league altogether, while China, Germany, and Japan are the strongest of the remaining countries. The empirical studies in our sample seem to form a healthy mix of case studies, experiments, and benchmarks. The latter play a larger role than in other fields of SE, which can be explained by the data-driven nature of the methods that often lend themselves to being benchmarked. In these studies, we see overall many realistic problems, data sets, and applications in practice. The involvement of practitioners improves some quality characteristics of the studies, like the realism of the case studies, and more significantly, their scale. We have also found, however, that authors often ignore discussing threats to validity. The terminology in the primary studies is all but homogeneous. This makes it often difficult to judge the scope of the contributions. We therefore propose to include taxonomy in each SE4AI paper that clarifies the level of AI that the contribution is associated with. Furthermore, we suggest using the term AI component if the article is about a part of a system that uses AI. An AI-based system is a system consisting of various software and potentially other components with at least one AI component. Most of our primary studies are about AI-based systems or AI components. The most mentioned application domain for AI-based systems is automotive, while almost half of the contributions are not addressing any specific application domain. In terms of methods, almost all contributions use ML techniques, with DL as the largest explicitly mentioned technique.

REFERENCES

[1]. Raja Ben Abdessalem, Shiva Nejati, Lionel C. Briand, and Thomas Stifter. 2018. Testing vision-based control systems using learnable evolutionary algorithms. In Proceedings of the 40th International Conference on Software Engineering. ACM, New York, NY, USA, 1016–1026. https://doi.org/10. 1145/3180155.3180160

[2]. Morayo Adedjouma, Gabriel Pedroza, and Boutheina Bannour. 2018. Representative Safety Assessment of Autonomous Vehicle for Public Transportation. In 2018 IEEE 21st International Symposium on Real-Time Distributed Computing (ISORC). IEEE, 124–129. https://doi.org/10.1109/ ISORC.2018.00025

[3]. Aniya Aggarwal, Pranay Lohia, Seema Nagar, Kuntal Dey, and Diptikalyan Saha. 2019. Black box fairness testing of machine learning models. In Proceedings of the 2019 27th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering. ACM, 625–635. https://doi.org/10.1145/3338906.3338937

[4]. Rama Akkiraju, Vibha Sinha, Anbang Xu, Jalal Mahmud, Pritam Gundecha, Zhe Liu, Xiaotong Liu, and John Schumacher. 2018. Characterizing machine learning process: A maturity framework. arXiv (2018).

[5]. Mohannad Alahdab and Gül Çalıklı. 2019. Empirical Analysis of Hidden Technical Debt Patterns in Machine Learning Software. In Product-Focused Software Process Improvement. Springer International Publishing, 195–202. https://doi.org/10.1007/978-3-030-35333-9_14

[6]. Saleema Amershi, Andrew Begel, Christian Bird, Robert DeLine, Harald Gall, Ece Kamar, Nachiappan Nagappan, Besmira Nushi, and Thomas Zimmermann. 2019. Software Engineering for Machine Learning: A Case Study. In 2019 IEEE/ACM 41st International Conference on Software Engineering: Software Engineering in Practice (ICSE-SEIP). IEEE, 291–300. https://doi.org/10.1109/ICSE-SEIP.2019.00042

[7]. Dario Amodei, Chris Olah, Jacob Steinhardt, Paul Christiano, John Schulman, and Dan Mané. 2016. Concrete Problems in AI Safety. arXiv 277, 2003 (2016), 1–29. arXiv:1606.06565 http://arxiv.org/abs/1606.06565 4

[8]. Apostolos Ampatzoglou, Stamatia Bibi, Paris Avgeriou, Marijn Verbeek, and Alexander Chatzigeorgiou. 2019. Identifying, categorizing and mitigating threats to validity in software engineering secondary studies. Information and Software Technology 106 (2019), 201–230.

[9]. Adina Aniculaesei, Jörg Grieser, Andreas Rausch, Karina Rehfeldt, and Tim Warnecke. 2018. Towards a holistic software systems engineering approach for dependable autonomous systems. In Proceedings of the 1st International Workshop on Software Engineering for AI in Autonomous Systems. ACM, 23–30. https://doi.org/10.1145/3194085.3194091

[10]. Adina Aniculaesei, Jorg Grieser, Andreas Rausch, Karina Rehfeldt, and Tim Warnecke. 2019. Graceful Degradation of Decision and Control Responsibility for Autonomous Systems based on Dependability Cages. 5th International Symposium on Future Active

Safety Technology toward Zero Accidents (FAST-zero '19) September (2019), 1–6.
[11]. Gary Anthes. 2017. Artificial intelligence poised to ride a new wave. Commun. ACM 60, 7 (2017), 19–21.
[12]. M. Arnold, R. K.E. Bellamy, M. Hind, S. Houde, S. Mehta, A. Mojsilović, R. Nair, K. Natesan Ramamurthy, D. Reimer, A. Olteanu, D. Piorkowski, J. Tsay, and K. R. Varshney. 2018. FactSheets: Increasing trust in ai services through supplier's declarations of conformity. arXiv (2018). arXiv:1808.07261
[13]. Anders Arpteg, Bjorn Brinne, Luka Crnkovic-Friis, and Jan Bosch. 2018. Software Engineering Challenges of Deep Learning. In 2018 44th Euromicro Conference on Software Engineering and Advanced Applications (SEAA). IEEE, 50–59. https://doi.org/10.1109/SEAA.2018.00018 arXiv:1810.12034
[14]. Peter Bailis, Kunle Olukotun, Christopher Ré, and Matei Zaharia. 2017. Infrastructure for usable machine learning: The stanford DAWN project. arXiv (2017). arXiv:1705.07538
[15]. Alec Banks and Rob Ashmore. 2019. Requirements assurance in machine learning. CEUR Workshop Proceedings 2301 (2019).
[16]. Somil Bansal and Claire J. Tomlin. 2018. Control and Safety of Autonomous Vehicles with Learning-Enabled Components. In Safe, Autonomous and Intelligent Vehicles. Springer International Publishing, 57–75. https://doi.org/10.1007/978-3-319-97301-2_4
[17]. V. Basili, G. Caldiera, and H. D. Rombach. 1994. The Goal Question Metric Approach. In Encyclopedia of Software Engineering, Vol. 2. John Wiley & Sons, 528—532.
[18]. Denis Baylor, Eric Breck, Heng-Tze Cheng, Noah Fiedel, Chuan Yu Foo, Zakaria Haque, Salem Haykal, Mustafa Ispir, Vihan Jain, Levent Koc, Chiu Yuen Koo, Lukasz Lew, Clemens Mewald, Akshay Naresh Modi, Neoklis Polyzotis, Sukriti Ramesh, Sudip Roy, Steven Euijong Whang, Martin Wicke, Jarek Wilkiewicz, Xin Zhang, and Martin Zinkevich. 2017. TFX: A TensorFlow-Based Production-Scale Machine Learning Platform. In Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, 1387–1395. https: //doi.org/10.1145/3097983.3098021
[19]. Woubshet Behutiye, Pertti Karhapää, Lidia López, Xavier Burgués, Silverio Martínez-Fernández, Anna Maria Vollmer, Pilar Rodríguez, Xavier Franch, and Markku Oivo. 2020. Management of quality requirements in agile and rapid software development: A systematic mapping study. Information and software technology 123 (2020), 106225.
[20]. Hrvoje Belani, Marin Vukovic, and Zeljka Car. 2019. Requirements Engineering Challenges in Building AI-Based Complex Systems. In 2019 IEEE 27th International Requirements Engineering Conference Workshops (REW). IEEE, 252–255. https://doi.org/10.1109/REW.2019.00051
[21]. Lucas Bernardi, Themistoklis Mavridis, and Pablo Estevez. 2019. 150 Successful Machine Learning Models. In Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. ACM, 1743–1751. https://doi.org/10.1145/3292500.3330744
[22]. Jan Aike Bolte, Andreas Bär, Daniel Lipinski, and Tim Fingscheidt. 2019. Towards corner case detection for autonomous driving. arXiv Iv (2019).
[23]. Markus Borg, Cristofer Englund, Krzysztof Wnuk, Boris Duran, Christoffer Levandowski, Shenjian Gao, Yanwen Tan, Henrik Kaijser, Henrik Lönn, and Jonas Törnqvist. 2018. Safely entering the deep: A review of verification and validation for machine learning and a challenge elicitation in the automotive industry. arXiv preprint arXiv:1812.05389 (2018).