POLITECNICO
MILANO 1863

Computer Science and Engineering

Software Engineering 2

Academic year 2021-2022

Research Project

Software Quality Metrics for AI-based software systems:

a multivocal literature review

Author: Federica Suriano
Student ID code: 953085
Reference Professor: Damian Andrew Tamburri

November 2, 2022

Contents

List of Tables

List of Figures

Abstract

Nowadays, software systems based on artificial intelligence are widely used both at the academic and research level, as well as in the industrial field. This study aims to identify the quality metrics that such a system should meet and the future challenges that AI and ML experts will face for the universal determination of high-quality AI software systems. By conducting a multivocal literature review we concluded that the non-functional requirements of typical software systems are necessary but not sufficient to comprehensively determine the quality of an AI-based software system. In fact, there are other factors to consider such as the ethics of AI. The research also reveals that very few AI software has been properly evaluated from all the qualitative standpoints a software should conform to. A joint effort by experts in software engineering and artificial intelligence will be required to determine the methods and tools necessary for the analysis of these quality requirements.

1 Introduction

The purpose of this research project is to present the state of the art on software engineering for AI-based systems by producing a multivocal literature review (MLR). A MLR is a form of systematic literature review (SLR) [1] that includes grey literature in addition to published formal literature. The choice to use this type of approach derives from the fact that, grey literature can give enormous advantages in some areas of software engineering (SE), as within it evidence is often based on experience [2].

The goal of this review is to define the concept of AI-based systems in a software engineering context and identify the most common quality issues in them based on the results of an empirical research.

1.1 Related work

Many other studies on AI-based systems in software engineering have been carried out by different research groups. Those considered to be of greatest interest for the purposes of this research - because they led to the birth of the research itself aimed at filling some gaps found in the literature - will be cited below.

In Reference [3] the authors analyzed the main software quality problems in AI systems, based on the experience of their three research groups. They identified the key points and discussed possible solutions that might be adopted to solve the emerged issues. They stated one of the biggest shortcomings in AI is that the developers aren't trained enough to develop these kinds of systems. This is a theme also taken up in Reference [4], according to which the solution is to fill the gap between the approaches used by software engineering developers to produce code and approaches suitable for AI-based systems. This is necessary to avoid several issues related not only to the low quality of the AI code, but also to its future maintenance.

Satoshi Masuda, Kohichi Ono, Toshiaki Yasue and Nobuhiro Hosokawa carried out a survey to discover and evaluate techniques to improve the software qualities of ML applications. Problems reported in ML applications arise because training data determines the logic of ML applications and results to unknown data cannot be verified in terms of correctness [5]. Hence, new software engineering approaches are needed to solve the problems.
The survey is carried out by classifying the research results in some reference targets. From the scarce number of papers found, it emerged that the topics regarding the quality of the software in this type of applications are still poorly considered.

In Reference [6], to collect and analyze the state of the art of SE knowledge for AI-based systems, a systematic mapping study was conducted. Also in this case, the need to update the current quality standards relating to AI-based systems was identified.

All these studies mentioned above, analyze the problem of the software quality of AI and ML applications, but none of these do carry out an empirical research based on the data obtained through a MLR. For this reason, we want to give our contribution to the analysis of this emerging problem by analyzing the results produced by the research carried out on both formal published literature and grey literature.

1.2 Structure of the article

The purpose of this introduction is to show why this study is relevant and therefore why it was conducted.
In the next section *(Section 2 - Background information)* some historical background information will be discussed to clarify the research context.
In *Section 3 - Research materials and methods* we will analyze how this study was conducted by specifying the methods used for the research, the data preparation approach and the selected results. We will delve into the research findings, paying particular attention to their meaning.
Finally, in *Section 4 - Conclusion* we will discuss the main results and observations obtained from this research.

2 Background information

The concept of AI was born during a summer research project held in Darthmouth in 1955 whose main proposers were John McCarthy, Marvin L. Minsky, Nathaniel Rochester and Claude E. Shannon. They had a quite ambitious purpose for that time: "The study is to proceed on the basis of the conjecture that every aspect of learning or any other feature of intelligence can in principle be so precisely described that a machine can be made to simulate it" [7]. In fact, the goal of AI is to design rational agents, i.e. agents capable of acting rationally to achieve objectives even with incomplete information and insufficient time.

The discipline of artificial intelligence can be combined with that of software engineering, which focuses on the development of reliable and good quality programs [4].

For what concerns this research, it is necessary to know what is meant when we talk about AI-based systems. They are software systems that have at least one artificial intelligence component inside them, thanks to which they can perform certain functions [6].

Nowadays, AI is becoming more and more relevant in software development in order to exploit the potential and advantages of both these disciplines. However, many of the AI-based software applications are developed by developers without adequate training on quality practices or processes that a software should comply with [3].

3 Research materials and methods

A Multivocal Literature Review on AI-based software systems has been conducted to investigate the following research questions:

RQ1 What are the key quality attribute for AI-based systems?

RQ2 What approaches are used to verify that quality attributes in AI-based systems are met?

RQ3 What are the main challenges faced in developing AI-based software systems?

3.1 Data preparation approach

According to Geraldo Torres et al. there are three main reasons for introducing grey literature (GL) when performing a literature review [8]: lack of academic research on the topic, evidence in the GL, emerging research on the research topic.

Furthermore, conducting an MLR and therefore including the grey literature within the review, has several benefits [2]. First of all, most software practitioners do not publish on academic forums and therefore their voices, which is of significant importance, would be almost nil if we did not take into account grey literature as well as academic literature in this study. This allows to analyze the state of the art of the research topic from a more practical point of view. The approach used in white literature is, on the other hand, usually more theoretical, as it is produced by research done in Academia and which does not arise from a practical challenge in the industry.

If on the one hand, the grey literature allows us to give a voice to those who actually deal with a certain research topic, on the other hand it could also happen that the results found are irrelevant and do not bring any novelty to the existing literature. For this reason, in the following steps we will deal with selecting only the results that we consider relevant for the purposes of this research and we will define inclusion and exclusion criteria to be followed before drawing up any possible conclusions.

The guidelines proposed by Vahid Garousi et al. [9] will be followed. Once an MLR is planned, it shall be conducted according to five phases as reported in Figure 1.

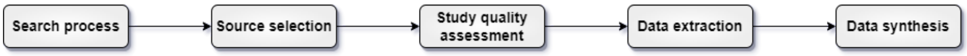

Figure 1: Five phases of conducting an MLR

3.2 Search strategy

A search query aimed at generating the most specific possible results for the questions to which this search wants to give an answer has been formulated.

Search query Q:
("software engineering for artificial intelligence" OR "software engineering for machine learning")
OR
(("software engineering" OR "software requirement*" OR "requirements engineering" OR "software design" OR "software architecture" OR "software testing" OR "software maintenance" OR "software quality")
AND
("AI-enabled system*" OR "AI-based system*" OR "AI-infused system*" OR "AI software" OR "artificial intelligence enabled system*" OR "artificial intelligence-based system*" OR "artificial intelligence-infused system*" OR "artificial intelligence software" OR "ML-enabled system*" OR "ML-based system*" OR "ML-infused system*" OR "ML software" OR "machine learning-enabled system*" OR "machine learning-based system*" OR "machine learning-infused system*" OR "machine learning software"))
AND

("quality attribute*" OR "quality requirement*" OR "quality metric*" OR "non functional requirement*")

For what concerns scientific literature, the search string has been applied to the scholarly search engines ACM Digital Library and SCOPUS.

Table 1 shows the number of results obtained with each search engine.

Search engine	*N results*
ACM Digital Library	31
SCOPUS	107

Table 1: Number of results - scientific literature

The Google scholar search engine was used to find the grey literature of interest. If the same search query Q had been applied to Google Scholar, this would have returned about 17000 results, so the choice was to simplify the query itself and this led us to have a much smaller number of results obtained and that are more consistent with the research topic itself. In particular, the search query used to obtain the grey literature is:

Search query Q1:
("software engineering for artificial intelligence" OR "SE4AI" OR "software engineering for machine learning" OR "SE4ML" OR "AI-based software system")
AND
("quality attribute" OR "quality requirement" OR "quality metric" OR "non functional requirement")

Table 2 shows the number of results obtained with Google Scholar search engine.

Search engine	*N results*
Google Scholar	37

Table 2: Number of results - grey literature

In both cases the search query results obtained refer to September 2022. Gray and scientific literature was merged into a single sample for review and the duplicates found were discarded. The final number of results to be submitted to the selection criteria is 160. The following rules were applied to the results:

1. literature published before 2015 is not taken into account;
2. only literature written in English is considered;
3. only literature fully accessible through the Politecnico di Milano credentials is considered.

The remaining results were subjected to an initial screening to eliminate those deemed not relevant for the purposes of this research. As regards the gray literature, the results deemed relevant will be reviewed and subjected to quality controls through inclusion and exclusion criteria as suggested by Vahid Garousi et al. [9].

Criteria	**Exclusion**
Authority of the producer	The publishing organization is not reputable
Methodology	The methodology is not reliable or crystal clear or the research is not supported by authoritative documents
Objectivity	The research is not objective or there is a hidden interest on the part of the authors
Date	The item has not a clearly stated date
Novelty	It doesn't enrich or add something unique to the research
Impact	It doesn't have a sufficient impact to the research

Table 3: Quality Criteria for Grey Literature

Figure 2 shows the selection procedure used to select the literature of interest.

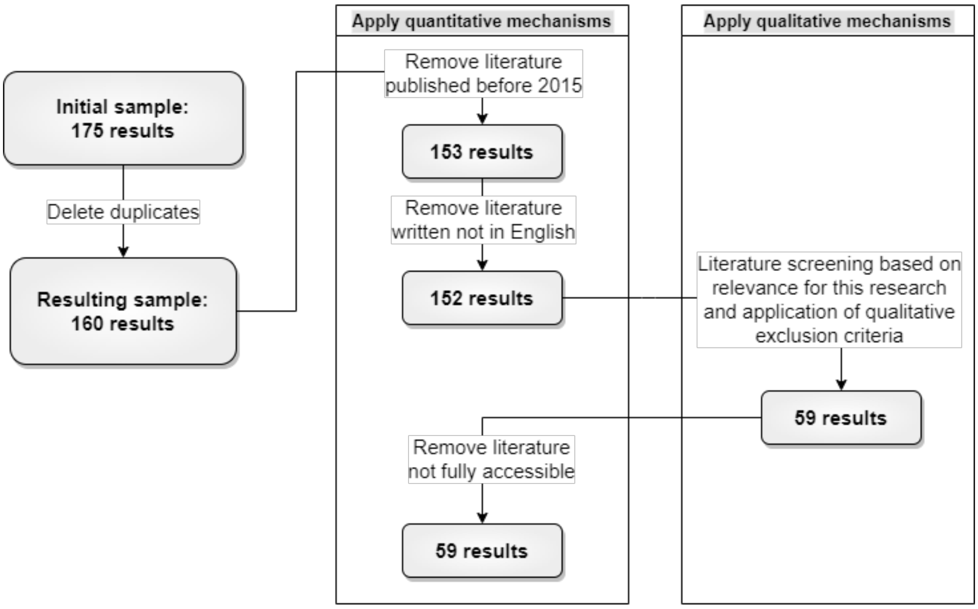
Apply quantitative mechanisms
Apply qualitative mechanisms
Initial sample: 175 results
Delete duplicates
Resulting sample: 160 results
Remove literature published before 2015
153 results
Remove literature written not in English
152 results
Literature screening based on relevance for this research and application of qualitative exclusion criteria
59 results
Remove literature not fully accessible
59 results

Figure 2: Selection process of the literature of interest

Starting from 175 results, in total 59 items passed all the steps of the selection criteria for this research. The resulting literature was read and analyzed in detail.

3.3 Data analysis

After selecting the relevant items, a quantitative analysis was conducted to detect how much the research topic was of interest to the scientific community over the years. Subsequently, all the papers obtained were qualitatively analyzed in order to answer the questions posed above.

3.4 Sample selection result

In Figure 3 we can see an increasing trend in the number of publications relating to each year. This shows an ever-growing interest in this topic, especially in recent years when there is the need to know with certainty how to identify and verify the quality parameters of software systems that have at least one AI component inside them.

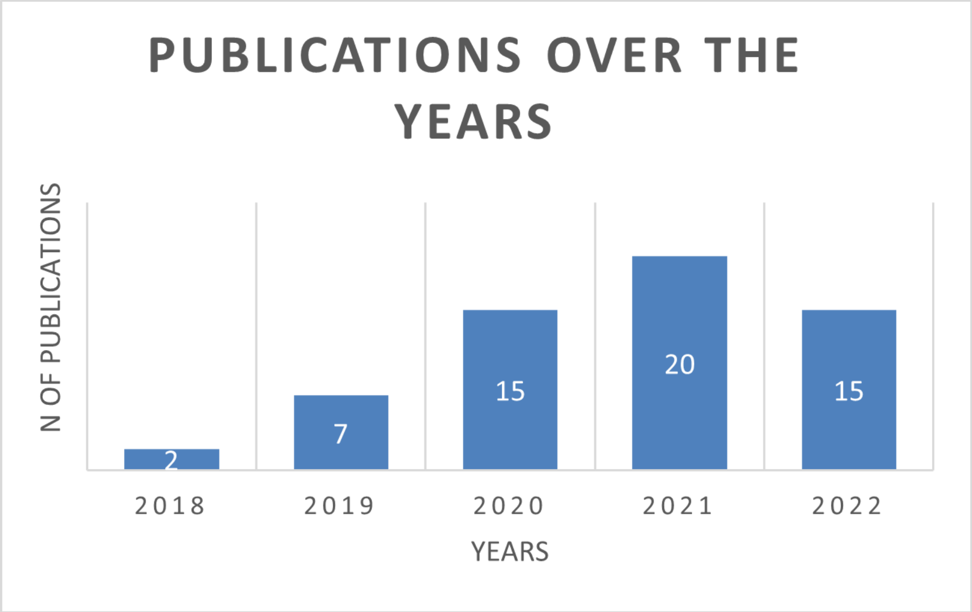

Figure 3: Publications over the years

The Figure 4 shows the percentage of each type of study present in the set of results analyzed. We can see that most of the literature derives from conference paper publications, but a good part of our analysis sample consists of articles that are taken from scientific journals, magazines or written as a result of workshops.

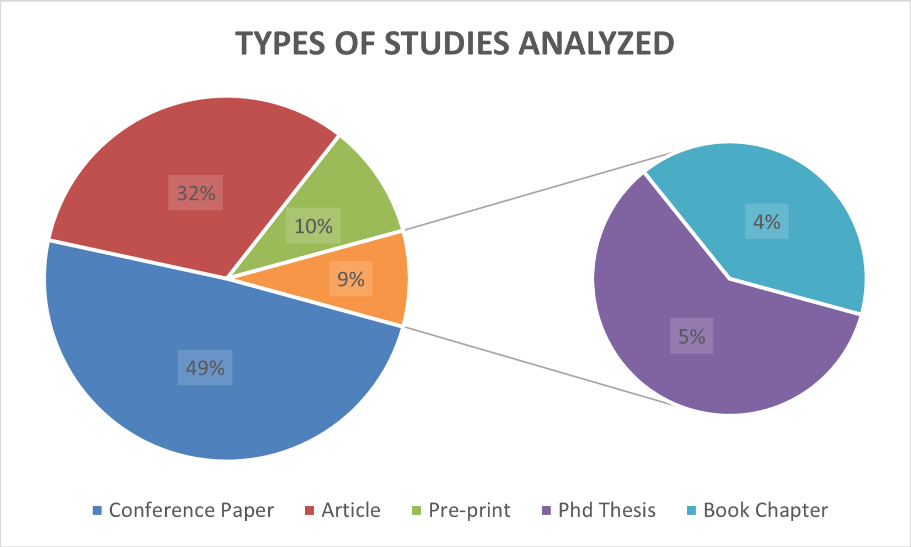

Figure 4: Types Of Publications

For ease of reading the research questions will be reported below again:

RQ1 What are the key quality attribute for AI-based systems?

RQ2 What approaches are used to verify that quality attributes in AI-based systems are met?

RQ3 What are the main challenges faced in developing AI-based software systems?

Guidelines for quality assurance of machine learning-based artificial intelligence come from Japan, thanks to Satoshi Masuda et al. and can be explored at Reference [10]. They extract five aspects of quality assessment for ML-based systems. These are **data integrity**, **model robustness**, overall **system quality** referring to parameters such as system performance, criticality and frequency of incidents, system controllability in case of incidents, explainability and insurability. **Agility of the development process** and **customer expectations** that relate to quality assurance for various stakeholders, who may not be familiar with developing ML-based systems, are also taken into account. To answer to **RQ1**, the analysis of the results revealed that the main qualitative attributes of an AI-based software system include not only all the attributes contained in a typical software system such as security, availability, scalability and more, but also **ethics**. In fact, with the advent of AI it has become a hot topic also debated outside the scientific community. It has therefore become necessary to write ethical guidelines for trustworthy AI [11]. The work was carried out by the High Level Expert Group on AI (AI HLEG), an independent expert group set up by the European Commission (EC) in June 2018. These guidelines follow four ethical principles: respect for humans, prevention of damage, fairness and explicability.

Despite the great work accomplished by Satoshi Masuda et al., the authors admit that a great deal of work is still needed by ML experts to precisely define the criteria for measuring the metrics and thresholds of the qualitative parameters. In order to answer the **RQ2**, we can conclude that, from what emerged from the analysis of the results, there are still **no well-defined and universal metrics** to verify the qualitative parameters of an AI-based software system.

Engineers have to adopt two types of development style when dealing with ML-based software systems: the inductive one for the core ML models - as for traditional software, engineers have a deep understanding of development due to their experience - and the deductive one for the entire ML-based system, given the **lack of knowledge of the ML systems** due to the characteristics of system itself. In fact, ML-based software systems are generally quite complex, non-linear and automatically generated.

The complexity of this type of system leads to a new challenge to be faced - and this answers **RQ3** - that is: making the systems as **transparent** as possible, with the creation of models through the support of **tools and methods**, which however are often missing. In particular, from what emerges from the analysis of the results, the main challenges that those who develop an AI/ML-based software system must face concern, not surprisingly, the areas of **testing**, **quality of AI software** and **data management** [12].
Another problem has also emerged that should not be overlooked: often, those who manage a project that includes this particular, but now widely used type of software system, have the difficult task of making decisions based on experience gained on software systems that do not include AI or ML component. One of the challenges to be faced is training people capable of combining different areas of information technology, but above all of giving them the means to do so, defining **methods to verify the qualitative parameters** of these systems.
This is hard work to complete: in fact, given the granularity of AI systems, which can vary a lot, it is very difficult to find methods to standardize the related quality parameters. The difficulty stems from the unique nature of ML, which is that system behavior is derived from training data, not from logical design by human engineers. This leads to intrinsically imperfect black-box implementations that invalidate many of the principles and techniques that exist in traditional software engineering.

4 Conclusion

In this study, we performed a multivocal literature review in order to analyze the literature related to software engineering that supports the development of AI/ML based software systems. A total of 59 documents were extracted and analyzed.

This study presents the key quality attributes of an AI-based software system and explains why it is so difficult to test and validate a system that embodies such complexity. To the non-functional requirements of a typical software system, there are other attributes such as the ethics of AI that are crucial for an AI-based software system to be considered qualitatively of a high level. The challenges faced by experts in the AI and ML sector are still many, including that of precisely defining all the parameters that contribute to the evaluation of an AI-based software system in a qualitative way and to the creation of methods and tools that can help to verify that the necessary requirements are met.

There still appears to be a gap between SE experts and AI experts. It is therefore necessary to make this distance between the two areas less and less wide in order to be able to train professional figures capable of supporting with their knowledge the whole process of creating, maintaining and verifying the requirements of such a complex system and capable of working with a variety and very large amount of data, which is inevitable when working on systems that have an ML component inside them. To adequately address the challenges raised in this paper and enable high-quality AI-based systems, the **exchange of knowledge** and ideas between the **SE** and the **AI community** is first and foremost necessary.

References

[1] Barbara Kitchenham et al. "Systematic literature reviews in software engineering – A systematic literature review". In: *Information and Software Technology* 51.1 (2009). Special Section - Most Cited Articles in 2002 and Regular Research Papers, pp. 7–15. ISSN: 0950-5849. DOI: `https://doi.org/10.1016/j.infsof.2008.09.009`. URL: `https://www.sciencedirect.com/science/article/pii/S0950584908001390`.

[2] Vahid Garousi, Michael Felderer, and Mika V. Mäntylä. "The Need for Multivocal Literature Reviews in Software Engineering: Complementing Systematic Literature Reviews with Grey Literature". In: EASE '16. New York, NY, USA: Association for Computing Machinery, 2016. ISBN: 9781450336918. DOI: `10.1145/2915970.2916008`. URL: `https://doi.org/10.1145/2915970.2916008`.

[3] Valentina Lenarduzzi et al. "Software Quality for AI: Where we are now?" In: Aug. 2020.

[4] Prince Jain. "Interaction between Software Engineering and Artificial Intelligence-A Review". In: *International Journal on Computer Science and Engineering* 3 (Dec. 2011).

[5] Satoshi Masuda et al. "A Survey of Software Quality for Machine Learning Applications". In: *2018 IEEE International Conference on Software Testing, Verification and Validation Workshops (ICSTW)*. 2018, pp. 279–284. DOI: `10.1109/ICSTW.2018.00061`.

[6] Silverio Martınez-Fernández et al. "Software engineering for AI-based systems: a survey". In: *ACM Transactions on Software Engineering and Methodology (TOSEM)* 31.2 (2022), pp. 1–59.

[7] J. McCarthy et al. *A PROPOSAL FOR THE DARTMOUTH SUMMER RESEARCH PROJECT ON ARTIFICIAL INTELLIGENCE.* http://www-formal.stanford.edu/jmc/history/dartmouth/dartmouth.html. 1955. URL: `http://www-formal.stanford.edu/jmc/history/dartmouth/dartmouth.html`.

[8] Geraldo Torres G. Neto et al. "Multivocal literature reviews in software engineering: Preliminary findings from a tertiary study". In: *2019 ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM).* 2019, pp. 1–6. DOI: 10.1109/ESEM.2019.8870142.

[9] Vahid Garousi, Michael Felderer, and Mika V. Mäntylä. "Guidelines for including grey literature and conducting multivocal literature reviews in software engineering". In: *Information and Software Technology* 106 (2019), pp. 101–121. ISSN: 0950-5849. DOI: https://doi.org/10.1016/j.infsof.2018.09.006. URL: https://www.sciencedirect.com/science/article/pii/S0950584918301939.

[10] Gaku Fujii et al. "Guidelines for quality assurance of machine learning-based artificial intelligence". In: *International Journal of Software Engineering and Knowledge Engineering* 30.11n12 (2020), pp. 1589–1606.

[11] H. Kuwajima and F. Ishikawa. "Adapting square for quality assessment of artificial intelligence systems". In: *Proceedings - 2019 IEEE 30th International Symposium on Software Reliability Engineering Workshops, ISSREW 2019.* Cited By :8. 2019, pp. 13–18.

[12] Elizamary Nascimento et al. "Software engineering for artificial intelligence and machine learning software: A systematic literature review". In: *arXiv preprint arXiv:2011.03751* (2020).

[13] Anh Nguyen-Duc and Pekka Abrahamsson. "Continuous Experimentation on Artificial Intelligence Software: A Research Agenda". In: *Proceedings of the 28th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering.* ESEC/FSE 2020. Virtual Event, USA: Association for Computing Machinery, 2020, pp. 1513–1516. ISBN: 9781450370431. DOI: 10.1145/3368089.3417039. URL: https://doi.org/10.1145/3368089.3417039.

[14] Saleema Amershi et al. "Software Engineering for Machine Learning: A Case Study". In: *Proceedings of the 41st International Conference on Software Engineering: Software Engineering in Practice.* ICSE-SEIP '19. Montreal, Quebec, Canada: IEEE Press, 2019, pp. 291–300. DOI: 10.1109/ICSE-SEIP.2019.00042. URL: https://doi.org/10.1109/ICSE-SEIP.2019.00042.

[15] Joymallya Chakraborty et al. "Fairway: A Way to Build Fair ML Software". In: *Proceedings of the 28th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering.* ESEC/FSE 2020. Virtual Event, USA: Association for Computing Machinery, 2020, pp. 654–665. ISBN: 9781450370431. DOI: 10.1145/3368089.3409697. URL: https://doi.org/10.1145/3368089.3409697.

[16] Soumyadip Bandyopadhyay, Rohan Mukherjee, and Santonu Sarkar. "A Report on the First Workshop on Software Engineering for Artificial Intelligence (SE4AI 2020)". In: *Proceedings of the 13th Innovations in Software Engineering Conference on Formerly Known as India Software Engineering Conference.* ISEC 2020. Jabalpur, India: Association for Computing Machinery, 2020. ISBN: 9781450375948. DOI: 10.1145/3385032.3385055. URL: https://doi.org/10.1145/3385032.3385055.

[17] Adrian Colyer. "Putting Machine Learning into Production Systems: Data Validation and Software Engineering for Machine Learning". In: *Queue* 17.4 (Aug. 2019), pp. 17–18. ISSN: 1542-7730. DOI: 10.1145/3358955.3365847. URL: https://doi.org/10.1145/3358955.3365847.

[18] Anh Nguyen-Duc et al. "A Multiple Case Study of Artificial Intelligent System Development in Industry". In: *Proceedings of the Evaluation and Assessment in Software Engineering.* EASE '20. Trondheim, Norway: Association for Computing Machinery, 2020, pp. 1–10. ISBN: 9781450377317. DOI: 10.1145/3383219.3383220. URL: https://doi.org/10.1145/3383219.3383220.

[19] Panagiotis Kourouklidis et al. "Towards a Low-Code Solution for Monitoring Machine Learning Model Performance". In: *Proceedings of the 23rd ACM/IEEE International Conference on Model Driven Engineering Languages and Systems: Companion Proceedings.* MODELS '20. Virtual Event, Canada: Association for Computing Machinery, 2020. ISBN: 9781450381352. DOI: 10.1145/3417990.3420196. URL: https://doi.org/10.1145/3417990.3420196.

[20] Sabine Moisan. "Intelligent Monitoring of Software Components". In: *Proceedings of the First International Workshop on Realizing AI Synergies in Software Engineering.* RAISE '12. Zurich, Switzerland: IEEE Press, 2012, pp. 17–21. ISBN: 9781467317535.

[21] Anush Sankaran et al. "DARVIZ: Deep Abstract Representation, Visualization, and Verification of Deep Learning Models". In: *Proceedings of the 39th International Conference on Software Engineering: New Ideas and Emerging Results Track.* ICSE-NIER '17. Buenos Aires, Argentina: IEEE Press, 2017, pp. 47–50. ISBN: 9781538626757. DOI: 10.1109/ICSE-NIER.2017.13. URL: https://doi.org/10.1109/ICSE-NIER.2017.13.

[22] Alex Serban et al. "Adoption and Effects of Software Engineering Best Practices in Machine Learning". In: *Proceedings of the 14th ACM / IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM).* ESEM '20. Bari, Italy: Association for Computing Machinery, 2020. ISBN: 9781450375801. DOI: 10.1145/

3382494.3410681. URL: https://doi.org/10.1145/3382494.3410681.

[23] Hadhemi Jebnoun et al. “The Scent of Deep Learning Code: An Empirical Study”. In: *Proceedings of the 17th International Conference on Mining Software Repositories*. MSR ’20. Seoul, Republic of Korea: Association for Computing Machinery, 2020, pp. 420–430. ISBN: 9781450375177. DOI: 10.1145/3379597.3387479. URL: https://doi.org/10.1145/3379597.3387479.

[24] Grace A Lewis, Ipek Ozkaya, and Xiwei Xu. “Software Architecture Challenges for ML Systems”. In: *2021 IEEE International Conference on Software Maintenance and Evolution (ICSME)*. IEEE. 2021, pp. 634–638.

[25] Ipek Ozkaya. “What is really different in engineering AI-enabled systems?” In: *IEEE Software* 37.4 (2020), pp. 3–6.

[26] Giordano d’Aloisio. “Quality-Driven Machine Learning-based Data Science Pipeline Realization: a software engineering approach”. In: *2022 IEEE/ACM 44th International Conference on Software Engineering: Companion Proceedings (ICSE-Companion)*. IEEE. 2022, pp. 291–293.

[27] Khlood Ahmad et al. “What’s up with Requirements Engineering for Artificial Intelligence Systems?” In: *2021 IEEE 29th International Requirements Engineering Conference (RE)*. IEEE. 2021, pp. 1–12.

[28] Görkem Giray. “A software engineering perspective on engineering machine learning systems: State of the art and challenges”. In: *Journal of Systems and Software* 180 (2021), p. 111031.

[29] Lucy Ellen Lwakatare et al. “Large-scale machine learning systems in real-world industrial settings: A review of challenges and solutions”. In: *Information and software technology* 127 (2020), p. 106368.

[30] Nicholas Maltbie. “Integrating Explainability in Deep Learning Application Development: A Categorization and Case Study”. PhD thesis. University of Cincinnati, 2021.

[31] Huaming Chen and M Ali Babar. “Security for Machine Learning-based Software Systems: a survey of threats, practices and challenges”. In: *arXiv preprint arXiv:2201.04736* (2022).

[32] Sangeeta Dey and Seok-Won Lee. “Multilayered review of safety approaches for machine learning-based systems in the days of AI”. In: *Journal of Systems and Software* 176 (2021), p. 110941.

[33] Junming Cao et al. “Characterizing Performance Bugs in Deep Learning Systems”. In: *arXiv preprint arXiv:2112.01771* (2021).

[34] AC Serban. “Designing Robust Autonomous Systems”. PhD thesis. [Sl]:[Sn], 2022.

[35] Mona Nashaat et al. “M-Lean: An end-to-end development framework for predictive models in B2B scenarios”. In: *Information and Software Technology* 113 (2019), pp. 131–145.

[36] Baraa Zieni. “Software Requirements Engineering for Transparency”. PhD thesis. University of Leicester, 2021.

[37] Houssem Ben Braiek and Foutse Khomh. “On testing machine learning programs”. In: *Journal of Systems and Software* 164 (2020), p. 110542.

[38] Jie M Zhang et al. “Machine learning testing: Survey, landscapes and horizons”. In: *IEEE Transactions on Software Engineering* (2020).

[39] Md Saidur Rahman et al. “Machine Learning Application Development: Practitioners’ Insights”. In: *arXiv preprint arXiv:2112.15277* (2021).

[40] V. Kharchenko, H. Fesenko, and O. Illiashenko. “Quality Models for Artificial Intelligence Systems: Characteristic-Based Approach, Development and Application”. In: *Sensors* 22.13 (2022).

[41] J. Siebert et al. “Construction of a quality model for machine learning systems”. In: *Software Quality Journal* 30.2 (2022). Cited By :3, pp. 307–335.

[42] B. Gezici and A. K. Tarhan. “Systematic literature review on software quality for AI-based software”. In: *Empirical Software Engineering* 27.3 (2022).

[43] Abdullah and J. Singh. “Applications of AI: Software Verifiability Point of View”. In: *2022 2nd International Conference on Advance Computing and Innovative Technologies in Engineering, ICACITE 2022*. 2022, pp. 1128–1133.

[44] D. Friesel and O. Spinczyk. “Black-Box Models for Non-Functional Properties of AI Software Systems”. In: *Proceedings - 1st International Conference on AI Engineering - Software Engineering for AI, CAIN 2022*. 2022, pp. 170–180.

[45] B. Van Oort et al. “’Project smells’ - Experiences in Analysing the Software Quality of ML Projects with mllint”. In: *Proceedings - International Conference on Software Engineering*. 2022, pp. 211–220.

[46] N. Nahar et al. “Collaboration Challenges in Building ML-Enabled Systems: Communication, Documentation, Engineering, and Process”. In: *Proceedings - International Conference on Software Engineering*. Vol. 2022-May. 2022, pp. 413–425.

[47] A. Khan et al. “Handling Non-Fuctional Requirements in IoT-based Machine Learning Systems”. In: *7th International Conference on Digital Arts, Media and Technology, DAMT 2022 and 5th ECTI Northern Section Conference on Electrical, Electronics, Computer and Telecommunications Engineering, NCON 2022*. 2022, pp. 477–479.

[48] L. Myllyaho et al. "Systematic literature review of validation methods for AI systems". In: *Journal of Systems and Software* 181 (2021). Cited By :4.

[49] H. Villamizar, T. Escovedo, and M. Kalinowski. "Requirements Engineering for Machine Learning: A Systematic Mapping Study". In: *Proceedings - 2021 47th Euromicro Conference on Software Engineering and Advanced Applications, SEAA 2021*. Cited By :3. 2021, pp. 29–36.

[50] N. Yoshioka et al. "Landscape of Requirements Engineering for Machine Learning-based AI Systems". In: *Proceedings - Asia-Pacific Software Engineering Conference, APSEC*. 2021, pp. 5–8.

[51] K. Ahmad. "Human-centric Requirements Engineering for Artificial Intelligence Software Systems". In: *Proceedings of the IEEE International Conference on Requirements Engineering*. 2021, pp. 468–473.

[52] K. M. Habibullah and J. Horkoff. "Non-functional Requirements for Machine Learning: Understanding Current Use and Challenges in Industry". In: *Proceedings of the IEEE International Conference on Requirements Engineering*. Cited By :5. 2021, pp. 13–23.

[53] M. Felderer and R. Ramler. *Quality Assurance for AI-Based Systems: Overview and Challenges (Introduction to Interactive Session)*. Vol. 404. Lecture Notes in Business Information Processing. Cited By :5. 2021, pp. 33–42.

[54] M. Borg. *The AIQ Meta-Testbed: Pragmatically Bridging Academic AI Testing and Industrial Q Needs*. Vol. 404. Lecture Notes in Business Information Processing. Cited By :7. 2021, pp. 66–77.

[55] F. Wotawa. "On the use of available testing methods for verification validation of ai-based software and systems". In: *CEUR Workshop Proceedings*. Vol. 2808. Cited By :1. 2021.

[56] K. Nakamichi et al. "Requirements-driven method to determine quality characteristics and measurements for machine learning software and its evaluation". In: *Proceedings of the IEEE International Conference on Requirements Engineering*. Vol. 2020-August. Cited By :8. 2020, pp. 260–270.

[57] F. Ishikawa and Y. Matsuno. "Evidence-driven Requirements Engineering for Uncertainty of Machine Learning-based Systems". In: *Proceedings of the IEEE International Conference on Requirements Engineering*. Vol. 2020-August. Cited By :6. 2020, pp. 346–351.

[58] H. Kuwajima, H. Yasuoka, and T. Nakae. "Engineering problems in machine learning systems". In: *Machine Learning* 109.5 (2020). Cited By :18, pp. 1103–1126.

[59] P. Santhanam. *Quality Management of Machine Learning Systems.* Vol. 1272. Communications in Computer and Information Science. Cited By :2. 2020, pp. 1–13.

[60] J. Siebert et al. *Towards guidelines for assessing qualities of machine learning systems.* Vol. 1266 CCIS. Communications in Computer and Information Science. Cited By :10. 2020, pp. 17–31.

[61] B. Kostova, S. Gürses, and A. Wegmann. "On the interplay between requirements, engineering, and artificial intelligence". In: *CEUR Workshop Proceedings.* Vol. 2584. Cited By :2. 2020.

[62] A. Vogelsang and M. Borg. "Requirements engineering for machine learning: Perspectives from data scientists". In: *Proceedings - 2019 IEEE 27th International Requirements Engineering Conference Workshops, REW 2019.* Cited By :57. 2019, pp. 245–251.

[63] J. Gao et al. "Invited paper: What is ai software testing? and Why". In: *Proceedings - 13th IEEE International Conference on Service-Oriented System Engineering, SOSE 2019, 10th International Workshop on Joint Cloud Computing, JCC 2019 and 2019 IEEE International Workshop on Cloud Computing in Robotic Systems, CCRS 2019.* Cited By :12. 2019, pp. 27–36.

[64] M. G. Gramajo, L. Ballejos, and M. Ale. "Software Requirements Engineering through Machine Learning Techniques: A Literature Review". In: *2018 IEEE Biennial Congress of Argentina, ARGENCON 2018.* 2019.