Opening the Software Engineering Toolbox for the Assessment of Trustworthy AI

Mohit Kumar Ahuja[1] and **Mohamed-Bachir Belaid**[1] and **Pierre Bernab**[1] and **Mathieu Collet**[1] and **Arnaud Gotlieb**[1] and **Chhagan Lal**[1] and **Dusica Marijan**[1] and **Sagar Sen**[1] and **Aizaz Sharif**[1] and **Helge Spieker**[1]

Abstract. Trustworthiness is a central requirement for the acceptance and success of human-centered artificial intelligence (AI). To deem an AI system as trustworthy, it is crucial to assess its behaviour and characteristics against a gold standard of Trustworthy AI, consisting of guidelines, requirements, or only expectations. While AI systems are highly complex, their implementations are still based on software. The software engineering community has a long-established toolbox for the assessment of software systems, especially in the context of software testing. In this paper, we argue for the application of software engineering and testing practices for the assessment of trustworthy AI. We make the connection between the seven key requirements as defined by the European Commission's AI high-level expert group and established procedures from software engineering and raise questions for future work.

1 INTRODUCTION

Artificial Intelligence (AI) has increasing relevance for many aspects of the current and future everyday life. Many of these aspects interfere directly with the personal space of humans, their perception, actions, and, more generally, their data, both online and offline. Due to this close integration, it is therefore crucial to develop the AI systems in a human-centered fashion such that they are trustworthy and can be accepted by providers, who develop and deploy the AI systems, users, who operate the AI systems, regulatory bodies, who oversee the usage and effects of the AI systems, and affected humans, who are act in cooperation with or next to the AI systems or who's data is subject to processing via the AI systems.

To define the extent and more specific definition of a trustworthy AI, a high-level expert group (AI-HLEG) that was set up by the European Commission, identified guidelines and requirements for an AI system that need to be sufficiently fulfilled to be regarded as trustworthy [12]. On the highest level, an AI system is deemed trustworthy if it behaves according to four ethical principles: respect for human autonomy, prevention of harm, fairness, and explicability [12, p. 12]; on a more technical level, requirements have been formulated that are supposed "to be continuously evaluated and addressed throughout the AI system's life cycle" [12, p. 15].

[1] Simula Research Laboratory, Dept. of Validation Intelligence for Autonomous Software Systems, Oslo, Norway, {mohit, bachir, pierbernabe, mathieu, arnaud, chhagan, dusica, sagar, aizaz, helge}@simula.no Funding: This work has received funding from the European Union under grant agreement no. 825619 (AI4EU), the EU landmark project to develop a European AI on-demand platform and ecosystem. Copyright ©2020 for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

Having a definition of trustworthy formulates a goal for the development of AI systems. The second step is to evaluate if a system fulfills the definition sufficiently and can be deemed trustworthy. This evaluation should be transparent and accessible to understand its results, robust and reproducible, and both automated and generic as much as possible to allow a low barrier for application to new AI systems. Since there is no single trustworthiness criterion or even metric, a single evaluation technique is not sufficient or maybe even possible. The trustworthiness assessment has to consist of multiple techniques, each appropriate for some of the requirements of trustworthy AI and each robust and mature enough to be reliable.

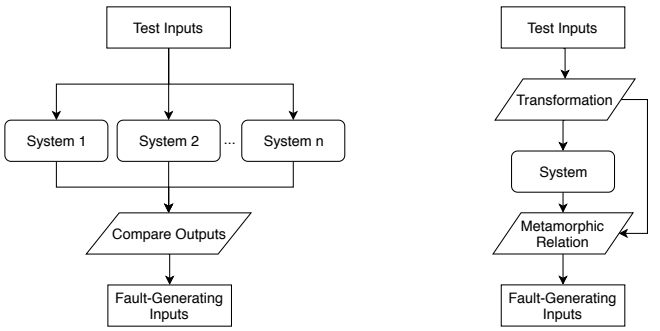
Tools and techniques for the assessment of trustworthy AI can be taken from the established methods of software engineering research and especially the subarea of software testing. For 50 years, these communities have proposed methods for the realization and assessment of large-scale, complex software systems. While the criteria for trustworthy AI do cover more than technical aspects, the AI system itself is still mostly a software system. Even though their are differences in their engineering, many of the software engineering principles apply to them or are transferable [1, 5]. Recently, motivated through the recent breakthroughs of AI and especially deep learning, the software engineering community has increased the attention on machine learning, both as a tool within software engineering and an area for the application of software engineering principles.

Through the remainder of this short paper, we argue to open the software engineering toolbox with its wide range of methods for the realization and assessment of trustworthy AI systems. Following the structure of the key requirements for trustworthy AI [12], we link existing techniques with the goals for the fulfillment of these requirements. It is important to note, that even though there are already many methods available, the research on trustworthy AI is by far not complete. Our current toolbox, however, provides a strong starting position but needs adjustments and further experiences to be adapted for the specific characteristics of modern AI.

2 TRUSTWORTHY AI

This section discusses an overview of approaches related to the key requirements for Trustworthy AI from the HLEG's Ethics Guidelines from software engineering and adjacent subfields. We aim to analyse how to map system engineering onto the requirements, and to show examples for techniques, case studies, that have already been explored. At the same time allows a discussion of existing techniques to identify where future research is required or areas where the software engineering toolbox might be insufficient to properly address aspects of a given requirement.

CEUR-WS.org/Vol-2659/ahuja.pdf



(a) Differential Testing (b) Metamorphic Testing

Figure 1. Schematic Overview of two testing techniques for Deep Learning Systems

2.1 Human agency and oversight

The first of the requirements is the necessity for the AI to support human autonomy and the option for the human to inspect and influence the AI's actions. Human agency directly affects the collaboration between AI and human and to support this interaction, it is important to take appropriate design measures, such as ergonomic and accessible user interfaces (UI) and an excellent user experience (UX). Human oversight requires the inspection of the AI decision making, either by having interpretable models or having access to design decision documents, source code, or data, depending on the level of expertise of the inspecting party.

It is also one of the main challenges in AI to find a perfect balance between enhancing human agency and preserving a degree of responsibility [11]. Some "black box" AI techniques prevent the human from understanding the embraced process and thus prevent him from the control. We believe that software engineering and testing frameworks can contribute in achieving a better degree of human understanding and control of AI techniques. Software testing techniques are often based around the goal to design the simplest test cases to determine a system's quality. Having these tests for AI systems will improve the ability to understand the AI behaviour and its deviations from it. While there is already work on applying and adopting current testing techniques on AI [29], future work is required to ease their capabilities and expressiveness for human oversight.

2.2 Technical Robustness and safety

The technical robustness of AI systems is central to their reliability. While performing well in their main performance metrics, e.g. the classification accuracy, additional safety, and robustness metrics and the resilience to attacks often remain open challenges [20]. Of particular relevance are adversarial inputs which are specially crafted to attack an AI system, for example, to cause misclassification or to extract internal information about training data.

These challenges have recently been identified by several software testing techniques and have been adopted towards the testing of AI systems, especially for deep learning. To highlight two techniques that have been successfully applied towards the testing of deep learning, we briefly discuss *differential* and *metamorphic* testing (see Figure 1). In differential testing, a system is evaluated by comparing its behaviour against a set of reference implementations for the same task. For the same inputs, it is expected that all systems provide similar outputs and if a system diverges it is an indicator of faulty behaviour. The advantage of differential testing is that the specific test oracles for the inputs, i.e. the precise expected outputs, are not required which allows easier setup of the test cases, especially when defining the test oracles is too costly or complex. DeepXplore [21] first explored differential testing for deep learning. The paper proposes a controlled way to generate test inputs, similar to adversarial examples, that are likely to identify diverging behaviours and showed promising results on multiple datasets and models.

Metamorphic testing also alleviates the problem of defining precise test oracles. Here, new test cases are generated with the help of metamorphic relations. These relations allow to describe a property of the behaviour, e.g. the output, when a change in the input is made. For example, for an AI-based HR system to rank resumes of applicants, adding relevant keywords should improve the ranking, even though there is no precise definition of the final expected ranking. In the context of testing AI, metamorphic testing has been applied for testing of autonomous driving systems [30], image classifiers [10], or ranking algorithms [19].

2.3 Privacy and data governance

Privacy protection of individuals who contribute with their personal data towards development of AI is of paramount importance in human-centered AI. Any party that curates datasets needs to ensure that the data does not provide means of re-identifying individuals while, at the same time, being effective at predicting patterns of business/societal value. Secure data-intensive systems storing personal data typically contain identifying, quasi-identifying, non-identifying and sensitive attributes about individuals.

Software tools such as ARX [22] can anonymize and perform re-identification risk analysis on large datasets to quantify the risk of prosecutor, journalist, and marketer attacks before the data is used in AI. ARX can be used to anonymize data based on criteria [8] such as k-anonymization (personal attributes are suppressed or generalized until each row is identical with at least k-1 other rows), l-diversity (entails reducing granularity of data), and t-closeness (a refined reduction of granularity by maintaining an underlying data distribution). However, in specific cases, quasi-identifying attributes such as the birth date of an individual are required to train AI models. Therefore, controlled fuzzification of quasi-identifying attributes [27] can minimize the risk of re-identification while maintaining underlying patterns of interest in the data. For instance, in cervical cancer screening, attributes such as birth date or screening exam date can be perturbed within certain bounds. This is primarily due to the fact that the human papillomavirus has an average latency period of 3 months. Therefore, database commands can fuzzy all dates to the 15th of a month (middle), and move months by ± 2 months without affecting disease progression patterns and increasing risk of re-identification.

2.4 Transparency

The transparency of an AI system is closely related to its interpretability and explainability, but also to the documentation of its purpose and how it has been designed. An approach for transparency documentation is the concept of *model cards* [18], which aims to provide accessible overview of a model for people of different expertise, including all of developers, testers, and technical end-users, similar to a package insert in pharmaceutical products.

Lower level measures for transparency can be achieved via strict traceability and static analysis [26] to allow the documentation of system behaviour, e.g. in autonomous vehicles [3] in combination with requirements engineering [7]. These techniques allow higher transparency of the AI during development, evaluation, and certification tasks, where they serve mostly technical needs for the development and integration of the AI component.

2.5 Diversity, non-discrimination and fairness

Adequate diversity in data to train AI systems is necessary to avoid discrimination and maintain fairness in human-centered AI. History has taught us that bias in using personal data has harmed several generations of ethnic minorities. Lundy Braun [4] reports the implications of biased data in spirometers that measure a person's lung function after a forced exhale. The predicted values of a lung's forced vital capacity (in litres of air exhaled) for black people for over a century been lower than white people. One of the reasons was that the data was collected from black men working in cotton fields where lint from cotton severely damaged lung function. This has resulted in black people receiving very little help from medical insurance companies for several generations. Even today, race and not socio-economic factor is used as a parameter to predict lung capacity in spirometers used worldwide. This unfortunate trend continues in AI systems where a recent study [15] shows that millions of black people are victims of biased decision making in health care systems.

Data needs to be carefully curated for training AI systems such that variation in human attributes such as different ethnic groups, genders, ages, weights, heights, geographical areas, and medical histories are taking into account for unbiased decision making. However, discovering if a data set satisfies all possible combinations of attributes is often computationally intractable. Combinatorial interaction testing (CIT) of software has been very effective in finding over 95% of all faults in a wide range of software systems using a very small set of tests covering all 2-wise/pairwise combinations of features [14]. CIT has been extended to verify if data in a large relational database contains all pairwise interactions between attribute values of interest [24]. Verifying the presence of all pairwise interactions in human attribute values in data set can clarify limitations or guarantee adequate diversity in human-centered AI systems.

The importance of fairness in software received attention as a dedicated topic in software engineering research [28] with close connections for the assessment via software testing methodology [6].

2.6 Societal and environmental well-being

Human-centered AI systems need to benefit society and not cause harm. It is necessary to see an AI system as not merely a software system but as a socio-technical system where the interaction between people the system is used to evaluate its benefit. Learning from epidemiology, we can evaluate an AI system as if it were an intervention on the public. For instance, in [25], the authors visualize the paths a patient takes after different screening exams for cervical cancer. Similarly, there is a need to understand how decisions made by the AI system affect the decisions made by people and the paths they take in life. Are people making healthier life choices, environmentally conscious, or giving a helping hand in society after an AI intervention? Evidence-based software engineering [13] inspired by epidemiology and clinical studies presents numerous approaches to evaluate the impact of AI on people. These approaches include randomized controlled trials, observational studies, and focus group discussions to name a few. All these approaches however require careful data collection after a target audience has been exposed to an AI system.

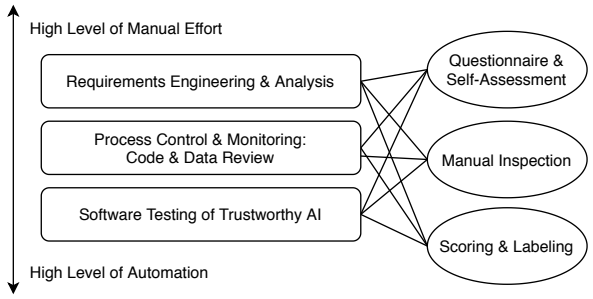


Figure 2. Concepts from the Software Engineering toolbox for the preparation, monitoring and evaluation of trustworthiness in AI projects.

2.7 Accountability

Access to personal data used in AI systems should be controlled by its owner in human-centered AI. The owner can give consent of use and take away access to personal data whenever he/she wants to. This implies that the AI system would need to be re-trained with or without a specific person's data. The proof of this operation should be made known to the owner to ensure accountability. The *blockchain* has the potential to facilitate the accountability of such transactions between a data owner and an AI system. The blockchain is a *distributed ledger* which was initially designed to record financial transactions. Numerous models of using the blockchain and smart contracts have now been proposed for data access control [9] and AI [23]. Tal Rapke [16] suggests that people own and access their health and life record on a decentralized blockchain that does not rely on a central storage facility. This will liberate organizations from the liability of storing personal data. The data will reside on the latest secure technology and using verifiable cryptography and owners of the data will be empowered to decide who they share their data with.

3 THE SOFTWARE ENGINEERING TOOLBOX

The discussion of the key requirements on trustworthy AI [12] shows that there are many challenges to be addressed, but also a set of methods available that can embraced and extended. As a general approach towards these challenges, we propose to adopt three main considerations (see Figure 2): First, since the expectations on trustworthy AI cannot be presented as a strict set of guidelines and rules only, it is recommended to understand their impact on the AI that is developed. Performing thorough requirements analysis helps to gather these requirements in a systematic way [2] and to formalize the requirements' impact on the AI including final acceptance criteria and whether they can be assessed automatically or require manual intervention. One method to guide the requirements analysis at this point could be to formulate checklists for each of the requirements, e.g. similar to this proposition for fairness [17].

Second, the realization of a trustworthy AI should be continuously accompanied by regular monitoring instruments. The goal of this monitoring is to ensure the awareness of trustworthiness measures during development. These monitoring instruments can include dedicated questions to consider during code and data reviews, as well as retrospective meetings.

Third, automated testing should be used to allow automated, repeated, and comparable assessment of trustworthiness. Where possible, testable acceptance criteria should be defined or test process that can quantify the behaviour of the AI system. For example, the technical robustness of an AI systems against adversarial inputs can be assessed through automatic techniques.

Finally, in all cases does the qualitative and quantitative summary of the results, e.g. via a score or a badge to attest the quality of an AI system, provides valuable information to the different stakeholder groups, e.g. the providers, their customers, or the users. A common scoring scheme, similar to the maturity levels in engineering projects, could allow for comparability and accessibility of the results.

4 CONCLUSION

The realization of trustworthy AI systems is one of the big challenges for the success of ethical and human-centered AI. This has been acknowledged by both politics [12] and academia. For the implementation of trustworthiness principles, we argue for the adoption of methods and technologies from software engineering. Software engineering has a long-standing tradition on the principled construction of complex systems and has already much of the fundamental work available, as shown throughout this paper.

Still, further work is necessary to cover all the requirements on trustworthy AI and to provide the tools and guidelines necessary for the widespread realization of trustworthy AI. Are the current software engineering tools sufficient to assess AI systems? Or do we need to develop dedicated tools? How can we converge on a set of acceptance criteria for trustworthiness? How many of the requirements can effectively be assessed in a mostly automated way? What are the challenges for assessing trustworthy AI by non-specialists or external users? How can we present the results in an accessible way?

The software engineering community has already taken up the challenge of software engineering for AI/ML, but often with a focus on the general system engineering, maintenance requirements, and general validation. However, as the requirements discussed in this paper showed, the engineering efforts need to cast a wider need and address more concerns in the context of trustworthy AI. This will be an interdisciplinary challenge and the software engineering toolbox can be of central relevance during its development.

REFERENCES

[1] Saleema Amershi, Andrew Begel, Christian Bird, Robert DeLine, Harald Gall, Ece Kamar, Nachiappan Nagappan, Besmira Nushi, and Thomas Zimmermann, 'Software Engineering for Machine Learning: A Case Study', in *International Conference on Software Engineering: Software Engineering in Practice*, (2019).

[2] Hrvoje Belani, Marin Vukovic, and Željka Car, 'Requirements Engineering Challenges in Building AI-Based Complex Systems', in *International Requirements Engineering Conference Workshops*, (2019).

[3] Markus Borg, Cristofer Englund, and Boris Duran, 'Traceability and deep learning-safety-critical systems with traces ending in deep neural networks', *Proc. of the Grand Challenges of Traceability: The Next Ten Years*, (2017).

[4] Lundy Braun, *Breathing race into the machine: The surprising career of the spirometer from plantation to genetics*, U of Minnesota, 2014.

[5] Eric Breck, Shanqing Cai, Eric Nielsen, Michael Salib, and D. Sculley, 'The ML test score: A rubric for ML production readiness and technical debt reduction', in *IEEE International Conference on Big Data*, (2017).

[6] Yuriy Brun and Alexandra Meliou, 'Software fairness', in *ACM ESEC/FSE*, p. 754759, (2018).

[7] L. M. Cysneiros, M. Raffi, and J. C. Sampaio do Prado Leite, 'Software transparency as a key requirement for self-driving cars', in *International Requirements Engineering Conference (RE)*, (2018).

[8] Anil Prakash Dangi and Ravindar Mogili, 'Privacy preservation measure using t-closeness with combined l-diversity and k-anonymity', *International Journal of Advanced Research in Computer Science and Electronics Engineering (IJARCSEE)*, **1**(8), (2012).

[9] George Drosatos and Eleni Kaldoudi, 'Blockchain applications in the biomedical domain: a scoping review', *Computational and structural biotechnology journal*, (2019).

[10] A. Dwarakanath, M. Ahuja, S. Sikand, RM Rao, RP Bose, N. Dubash, and S. Podder, 'Identifying implementation bugs in machine learning based image classifiers using metamorphic testing', in *ACM International Symposium on Software Testing and Analysis (ISSTA)*, (2018).

[11] L. Floridi, J. Cowls, M. Beltrametti, R. Chatila, P. Chazerand, V. Dignum, C. Luetge, R. Madelin, U. Pagallo, F. Rossi, et al., 'Ai4peoplean ethical framework for a good ai society: Opportunities, risks, principles, and recommendations', *Minds and Machines*, **28**(4), (2018).

[12] High-Level Expert Group on Artificial Intelligence, 'Ethics Guidelines for Trustworthy AI', Technical report, European Commission, (2019).

[13] Barbara A Kitchenham, Tore Dyba, and Magne Jorgensen, 'Evidence-based software engineering', in *International Conference on Software Engineering (ICSE)*, pp. 273–281, (2004).

[14] D Richard Kuhn and Michael J Reilly, 'An investigation of the applicability of design of experiments to software testing', in *27th Annual NASA Goddard/IEEE Software Engineering Workshop*, (2002).

[15] Heidi Ledford, 'Millions of black people affected by racial bias in health-care algorithms.', *Nature*, **574**(7780), (2019).

[16] Laure A Linn and Martha B Koo, 'Blockchain for health data and its potential use in health it and health care related research', in *ONC/NIST Use of Blockchain for Healthcare and Research Workshop*, (2016).

[17] Michael A Madaio, Luke Stark, Jennifer Wortman Vaughan, and Hanna Wallach, 'Co-Designing Checklists to Understand Organizational Challenges and Opportunities around Fairness in AI', in *CHI*, (2020).

[18] Margaret Mitchell, Simone Wu, Andrew Zaldivar, Parker Barnes, Lucy Vasserman, Ben Hutchinson, Elena Spitzer, Inioluwa Deborah Raji, and Timnit Gebru, 'Model cards for model reporting', in *Conference on Fairness, Accountability, and Transparency - FAT**, (2019).

[19] Christian Murphy, Gail Kaiser, Lifeng Hu, and Leon Wu, 'Properties of Machine Learning Applications for Use in Metamorphic Testing', *20th International Conference on Software Engineering and Knowledge Engineering (SEKE)*, (2008).

[20] Nicolas Papernot, Patrick McDaniel, Arunesh Sinha, and Michael P. Wellman, 'SoK: Security and Privacy in Machine Learning', in *IEEE European Symposium on Security and Privacy (EuroS&P)*, (2018).

[21] Kexin Pei, Yinzhi Cao, Junfeng Yang, and Suman Jana, 'Deepxplore: Automated whitebox testing of deep learning systems', in *Symposium on Operating Systems Principles*, (2017).

[22] Fabian Prasser and Florian Kohlmayer, 'Putting statistical disclosure control into practice: The arx data anonymization tool', in *Medical Data Privacy Handbook*, (2015).

[23] Khaled Salah, M Habib Ur Rehman, Nishara Nizamuddin, and Ala Al-Fuqaha, 'Blockchain for ai: Review and open research challenges', *IEEE Access*, **7**, (2019).

[24] Sagar Sen, Dusica Marijan, Carlo Ieva, Astrid Grime, and Atle Sander, 'Modeling and verifying combinatorial interactions to test data intensive systems: Experience at the norwegian customs directorate', *IEEE Transactions on Reliability*, **66**(1), (2016).

[25] Sagar Sen, Manoel Horta Ribeiro, Racquel C De Melo Minardi, Wagner Meira, and Mari Nygård, 'Portinari: a data exploration tool to personalize cervical cancer screening', in *International Conference on Software Engineering: Software Engineering in Society (ICSE-SEIS)*, (2017).

[26] Caterina Urban, 'Static analysis of data science software', in *International Static Analysis Symposium*, pp. 17–23, (2019).

[27] Giske Ursin, Sagar Sen, Jean-Marie Mottu, and Mari Nygård, 'Protecting privacy in large datasetsfirst we assess the risk; then we fuzzy the data', *Cancer Epidemiology and Prevention Biomarkers*, **26**(8), (2017).

[28] S. Verma and J. Rubin, 'Fairness definitions explained', in *IEEE/ACM International Workshop on Software Fairness (FairWare)*, (2018).

[29] Jie M. Zhang, Mark Harman, Lei Ma, and Yang Liu, 'Machine Learning Testing: Survey, Landscapes and Horizons', *IEEE Transactions on Software Engineering*, (2020).

[30] Mengshi Zhang, Yuqun Zhang, Lingming Zhang, Cong Liu, and Sarfraz Khurshid, 'DeepRoad: GAN-based metamorphic testing and input validation framework for autonomous driving systems', in *International Conference on Automated Software Engineering (ASE)*, (2018).