An overview of key trustworthiness attributes and KPIs for trusted ML-based systems engineering

Juliette Mattioli, Henri Sohier, Agnès Delaborde, Kahina Amokrane, Afef Awadid, Zakaria Chihani, Souhaiel Khalfaoui, Gabriel Pedroza

An overview of key trustworthiness attributes and KPIs for trusted ML-based systems engineering

Juliette MATTIOLI[1], **Henri SOHIER**[2], **Agnès DELABORDE**[2, 3], **Kahina AMOKRANE-FERKA**[2]
Afef AWADID[2], **Zakaria CHIHANI**[4] **Souhaiel KHALFAOUI**[2, 5] **Gabriel PEDROZA**[4]

[1] Thales, [2] IRT SystemX, [3] Laboratoire National de métrologie et d'Essais, [4] CEA List, [5] Valéo
France

Abstract

When deployed, machine-learning (ML) adoption depends on its ability to actually deliver the expected service safely, and to meet user expectations in terms of quality and continuity of service. For instance, the users expect that the technology will not do something it is not supposed to do, *e.g.,* performing actions without informing users. Thus, the use of Artificial Intelligence (AI) in safety-critical systems such as in avionics, mobility, defense, and healthcare requires proving their trustworthiness through out its overall lifecycle (from design to deployment). Based on surveys on quality measures, characteristics and sub-characteristics of AI systems, the Confiance.ai program (www.confiance.ai) aims to identify the relevant trustworthiness attributes and their associated Key Performance Indicators (KPI) or their associated methods for assessing the induced level of trust.

Motivation for ML trustworthiness assessment

Trustworthiness is tightly related to accountability: accountability can be considered as a factor of trust or as an alternative to trust [57]. Then, in [4], *dependability* is used to represent the overall quality measure of a system based on four sub-attributes including *security*, *safety*, *reliability*, and *maintainability*. Thereafter, security and dependability became key attributes for computer-based system trust [8]. In 2019, the U.S. National Artificial Intelligence Research and Development Strategic Plan [54] emphasized that: *"standard metrics are needed to define quantifiable measures in order to characterize AI technologies"*. More recently, [65] noted that "*significant work is needed to establish what appropriate metrics should be to assess system performance across attributes for responsible AI and across profiles for particular applications/contexts.*".

The Assessment List for Trustworthy AI [1] considers 7 pillars of trustworthiness: 1) human agency and autonomy, 2) technical robustness and safety, 3) privacy and data governance, 4) transparency, 5) diversity, non discrimination and fairness, 6) societal and environmental well-being, 7) accountability. The European Commission has proposed a set of rules for AI, the AI Act [19], regulating the technology. Such proposals, which are still at the consultation stage, would apply to AI systems developed or deployed in the EU and require companies to take measures to ensure their products are safe and comply with ethical principles based on a risk analysis.

In the aeronautic domain, EASA [14] proposes a model of trustworthiness based on: the characterization of the Machine Learning (ML) application (high-level function/task, concept of operations, functional analysis, classification of the ML application), safety assessment, information security management, and ethics-based assessment (which includes the 7 pillars of the ALTAI).

The Fraunhofer [52] offered an analysis of the standard [39, Under development] on management system for AI, stating compliance to the standard can contribute to ensuring AI trustworthiness since it encompasses the pillars of the ALTAI, provided that a third-party verification has been performed and along with an adapted quality management system.

In the same period, the NIST produced an analysis of the components of trust [68] and highlighted several top level aspects for the design of a trustworthiness model, that should encompass the user experience, the perceived technical trustworthiness, the pertinence of each trustworthiness characteristic in the user's specific context of use, *etc.*

Moreover, ETSI set-up in 2019 an Industry Specification Group on Securing AI (ISG SAI) from attack to resilience [17] providing existing and potential mitigation against threats for AI-based systems.

However, as a property of a ML-based system, such trustworthy concept is complex and determined by considering many characteristics as well as its application in particular contexts. This implies that the relative importance of each attribute can fluctuate depending on the circumstances wherein such system is operating [7]. While most active academic research on trustworthy ML has focused on the algorithm properties, its holistic modeling has received very little attention given the lack of literature.

The present work aims to characterize key trustworthy attributes and their associated assessment methods and metrics that can impact trustworthiness of ML-based systems [10]. In this paper, we highlight how trustworthiness characterization and assessment are positioned within ML engineering process. Then, we present a ML trustworthiness meta-model to capture relevant information and different inter-relations needed to assess the level of trust. In the meantime, we fo-

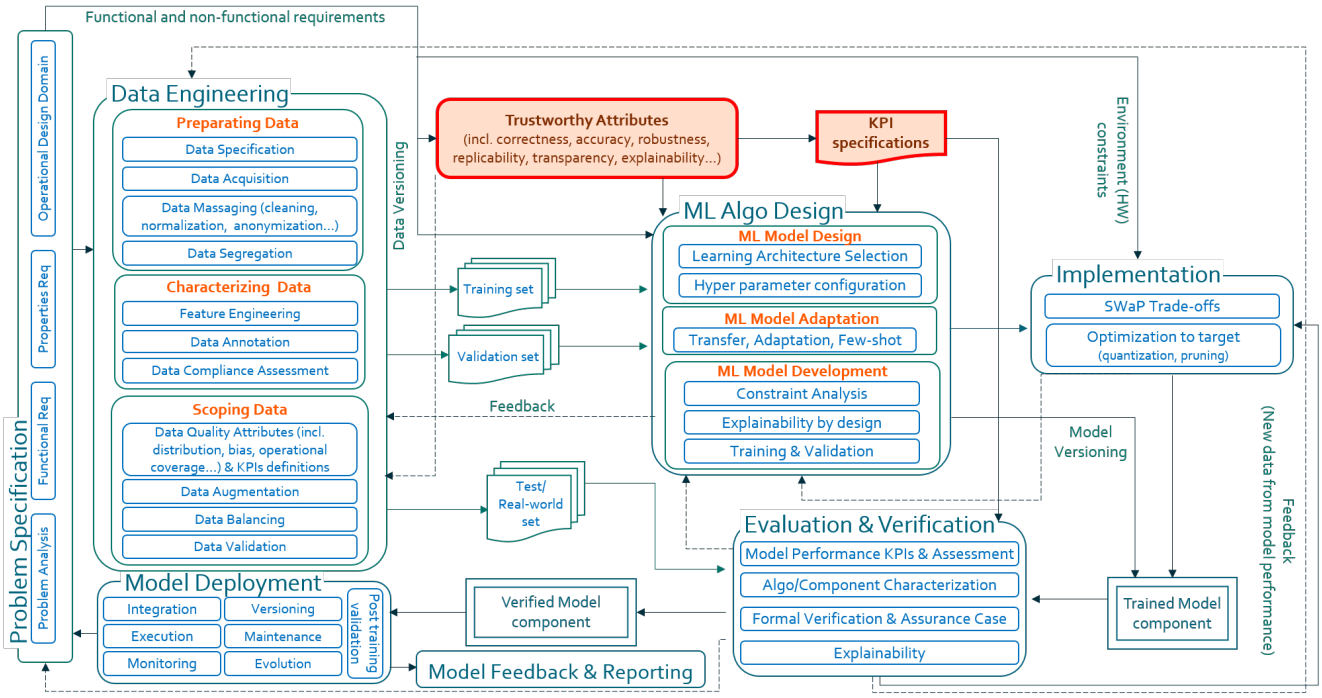

Figure 1: Trustworthiness assessment within the ML algorithm engineering process [49]

cus on 6 key trustworthy attributes namely data quality, dependability, operability, robustness, human centred quality including explainability / interpretability, and human oversight, providing references illustrated with some KPIs.

ML Algorithm Engineering

ML algorithm engineering [9, 67, 69] is a field revisiting algorithm engineering practices and processes [13, 64] through classic considerations on specification, traceability and validation [6, 56]; processing data requires new processes with new best practices [77], as highlighted by ML Model Operationalization Management (MLOps) approaches. ML system must present new assessments of trustworthiness through security, safety, robustness, explainability *etc.* To capture such issues, Confiance.ai program (www.confiance.ai) defines a ML pipeline (see fig. 1) emphasizing requirements-driven, safety-driven and ML-driven development. Main sub-tasks are encapsulated as a series of steps such as [49]: Problem specification; data engineering; ML Algorithm design; implementation; evaluation and verification; model deployment to provide trustworthy evidences on top of ML assurance [53, 66]. Each step has to be evaluated through KPIs and/or assessment methods.

From a strict metrological point of view, trustworthiness may not be measured as it is not a physical property that can be compared to a reference quantity of the same kind. Trustworthiness does not have units. More generally speaking, “to measure” refers to assign an element of a scale to an object in order to quantify an attribute of it. Thus, a trustworthiness metric is defined as [10] an “*objective, mathematical measure of a ML-based component/system that is sensitive to differences in safety-critical characteristics. It provides a quantitative measure of an attribute which the body of solution exhibits*”.

A new ML trustworthiness meta-model

A trustworthy software is defined [72] by a combination of overlapping properties: reliability, safety, security, privacy, availability and usability. For a ML-based system, this translates and extends to accuracy, robustness, fairness, accountability, transparency, explainability and ethics. [12] also considers auditability. To capture the type of considered information and the different inter-relations needed to assess ML trustworthiness, we proposed a meta-model with concepts in different abstraction levels (see fig. 2). The red part describes the way the tree of attributes is built. It highlights the abstract concepts central to trustworthiness assessment. An attribute which aggregates other attributes is called a macro-attribute (e.g. robustness, explainability, *etc.*). It is assessed with an aggregation method. An atomic attribute (leaf attribute) is assessed with a clear and actionable observable which can take different forms (metric, ”expected proof”).

The green part of fig. 2 is the meta-model fragment with concrete concepts. These concepts represent the different possible subjects and relations between them. For example, the product is developed following processes as technical processes (through which the product must go: design definition, implementation, operation, ...), agreement processes (with external organizations : acquisition, supply), and management processes (supporting the development of the product: quality management, risk management, *etc.*). Risk and quality management ensures the compliance with the specification which includes the different expected trustworthiness attributes. Processes are applied with tools by people respecting a certain governance.

The blue part summarizes systems engineering key concepts more precisely part of the non-functional specification: they do not define what the system ”does” or how the system works, but what the system ”is”. The attributes are also

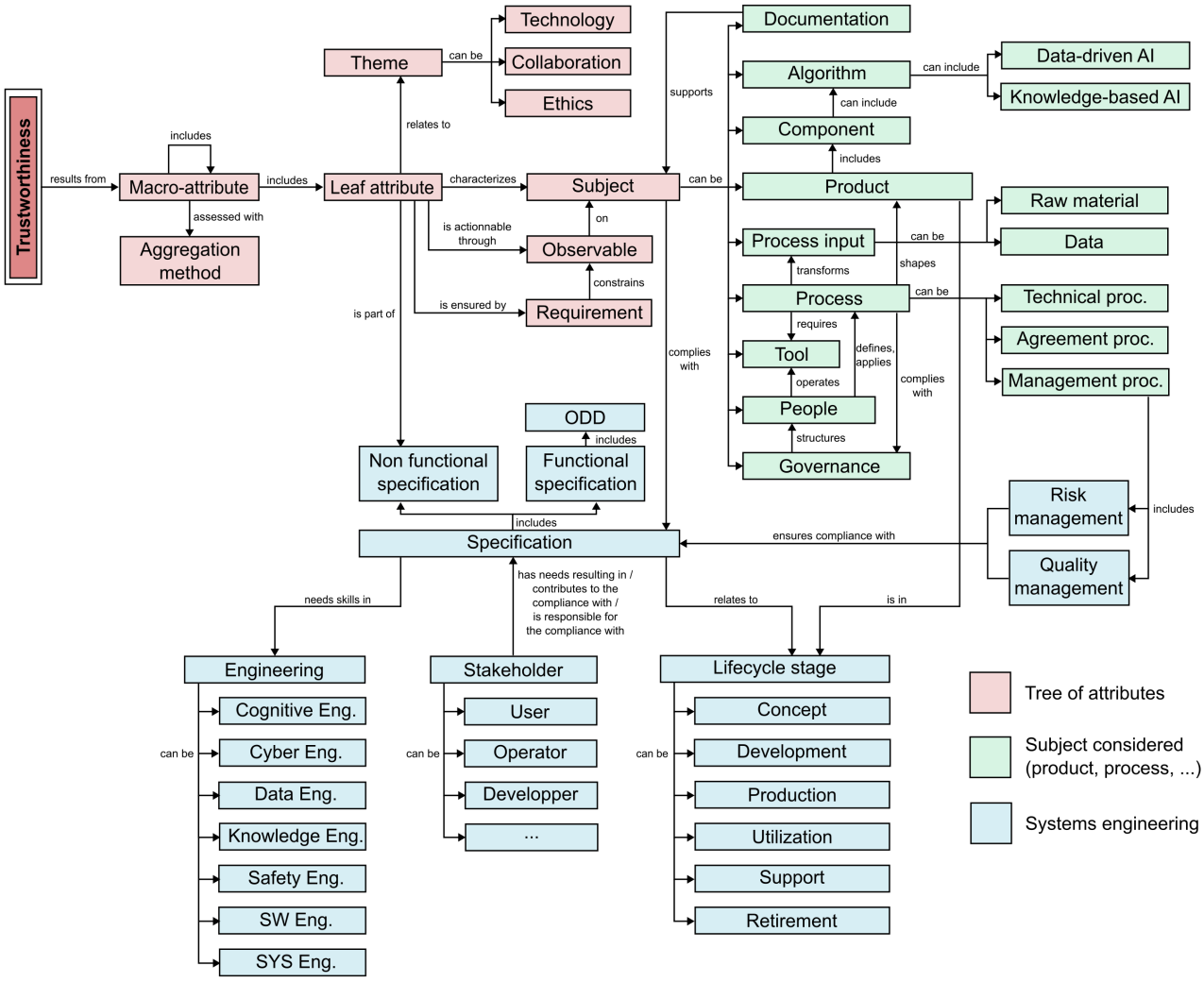

Figure 2: A new ML trustworthiness meta-model

commonly referred to as "*-ilities*" as they often have this suffix. They can also be referred to as quality requirements. Whether a specification is functional or non-functional, it is influenced by stakeholders such as the user, the operator, the developer, *etc.*

As opposed to non-functional requirements which define what the system is, functional requirements define what the system does: should it move? roll? roll fast? under what conditions? From this point of view, the Operational Design Domain (ODD), which characterizes the conditions of operation of the system, can be considered part of the functional specification relating to trustworthiness attributes in different ways: 1) Transparency on the ODD permits to understand the system's capabilities and limits (which is part of the AI Act's requirements); 2) The ODD is the domain to consider for the different operational trustworthiness attributes; 3) The ODD has its own attributes (it should be complete, free of inconsistencies, human readable, *etc.*).

According to the ML capabilities and the 7 pillars of trustworthiness [1], we characterize ML trustworthiness through 6 macro-attributes: data quality, dependability, operability, robustness, human centered quality including explainability / interpretability, and human oversight (see. fig. 5).

Data quality

ML-based system quality strongly depends on the quality of (training/test/validation) data sets where they are defined as an identifiable collection of data. Without a systematic assessment of their quality, ML approaches risk losing control of the various steps of data engineering such as data collection, annotation, feature engineering, and corpus balancing.

As proposed by [34], a first Data Quality (DQ) structure is based on 3 main key attributes (cf. Table 1): *Inherent DQ*: the degree to which quality attributes of data have the intrinsic potential to satisfy stated and implied needs when data is used under specified conditions; *System-Dependent DQ*: the degree to which data quality is reached and preserved within a computer system when data is used under specified conditions; and *Inherent and System-Dependent DQ*.

Data quality (DQ) requirements should be characterized, for each type of data representing an operating parameter of the ODD. In the ML context, well-founded metrics are needed to assess the DQ level. While both research and practice have realized the high relevance of well-founded DQ

Macro attributes	Leaf attributes
System Dependent DQ	Availability [34] Portability [34] Recoverability [34] Timeliness [11], [48]
Inherent & System Dependent DQ	Accessibility [48] Confidentiality [34] Compliance [34] Efficiency [34] Integrity [15] Precision [34] Traceability [34], [48]
Inherent DQ	Accuracy [21], [11],[48], [60] Completeness [71], [15], [23], [34] Consistency [20], [34], [48], [28] Correctness [21], [15] Currency [61], [5], [47] Diversification [44], [24] Usability [48] Representativeness [14], [48] Reliability [18]

Table 1: Data Quality macro and leaf attributes

metrics such as accuracy, many of them still lack an appropriate methodical foundation [29] to cover the overall data life cycle: *data collection*; *data labeling* needed for supervised ML; *data augmentation* to avoid overfitting on training data by introducing some data enrichment (diversity); data preparation including pre-processing, data transformation, and *feature engineering*; identification of the various datasets used in the learning phase (typically training, validation, and test datasets); datasets validation and verification (including accuracy, completeness, and representativeness, with respect to the ML requirements and the ODD); independence requirements between datasets; identification and elimination of unwanted biases inherent to the datasets... The quality dimensions, e.g., accuracy, can be easily detected in some cases (e.g., mispellings for natural language processing application) but are more difficult to detect in other cases (e.g., where admissible but not correct values are provided) [21].

Many researchers have used metrics for *data accuracy* based on the rate of correct data items over an entire dataset, using a 1 for an accurate data item, and a 0 otherwise: $data\_accuracy = \sum_{i=1}^{N} \alpha(d_i)/N$ where N is the number of data elements in the data-set, and $\alpha(d_i)$ is 1 if data element d_i is correct, and 0 otherwise. Data diversity is the ratio between the number of available data sources, their size, and the dataset are finally used [24]. Other DQ KPIs could be find in [5, 18, 48, 50].

Operability

Operability is the ability to keep a piece of equipment, a system or a whole industrial installation in a safe and reliable functioning condition, according to pre-defined operational requirements. This is thus considered one of the ilities and is closely related to reliability, supportability and maintainability. It is the degree to which a product or system is easy

Macro attributes	Leaf attributes
Effectiveness	Accuracy, Precision [15], [41] Functional suitability [33] Functional completeness [33] Functional correctness [74] Functional appropriateness [33]
Efficiency	Performance efficiency [33] Sustainability [70]
Adaptability/ Durability	Extensibility [33] Flexibility [31] Controllability [38]

Table 2: Operability macro and leaf attributes

to operate, control and appropriate to use [33].

The words accuracy and precision are important differentiated terms when referring to measurements in the scientific and technical context. Generally speaking, *accuracy* refers to how close a measured value is in relation to a known value or standard. On the other hand, *precision* is related to how close several measurements of the same quantity are to each other. In ML context, classification is a prediction type used to give the output variable in the form of categories with similar attributes. Some of the popular metrics for its assessment are accuracy, precision, recall, F1 Score, ROC Curve, Fowlkes–Mallows index [22] or the ABC metric (attribution-based confidence metric) [9, 51] (see fig. 3).

		Target		
		Positive	Negative	
Model	Positive	TP	FP	$Precision = \frac{TP}{TP+FP}$
	Negative	FN	TN	
		$Recall = \frac{TP}{TP+FN}$	$Specificity = \frac{TN}{FP+TN}$	$Accuracy = \frac{TP+TN}{TP+FP+FN+TN}$
		$F1 = \frac{2}{\frac{1}{Recall} + \frac{1}{Precision}}$		

Figure 3: Performance metrics for classification problems

The functional completeness measures what proportion of the specified functions has been implemented. A missing function is detected when the system or software product does not have the ability to perform a specified function. It is the fraction of 1) Number of functions missing and 2) Number of functions specified. For example, being evaluated for ML-based systems, we interpret that "missing" functions are the functions that were not successfully trained, even though developers specified to train them.

Dependability

Dependability can be defined as the ability of a system to deliver a service that can be justifiably trusted [4]. Over the years, the dependability concept has evolved to integrate other qualitative attributes such as: *availability* (readiness for correct service); *reliability* (continuity of correct service); *safety*; *security*; *confidentiality* (absence of unauthorized disclosure of information); *integrity* (absence of improper system alterations); *maintainability* (ability to un-

dergo modifications, and repairs)... In real-time computing, dependability is the ability to provide services that can be trusted within a time period. The service guarantees must hold even when the system is subject to attacks or natural failures. Moreover, [1] defines dependability as the ability to deliver services that can justifiably be trusted.

Concerning security, a cyber-attack can be generic, resulting in denial or degradation of service; or targeted, aiming to cause a model to behave in a specific way. For example, though poisoning attacks, ambiguity attacks *etc.,* typically affect the integrity of data, [17] notes that they can also be considered attacks on availability, as the aim of an attacker can be to increase misclassification to the point of making a system unusable. [43] proposes some mitigation methods based on three countermeasures that could be applied in order to prevent ambiguity attacks.

Regarding safety, normative works such as the ISO/IEC CD TR 5469 [37] (under development) offer principles for functional safety of AI models used in E/E/PE (Electrical/Electronic/Programmable Electronic) safety-related systems, including identification of risk factors and verification and validation techniques. However, safety verification needs are not limited to the sole proper functioning of the safety-related components, but also relates to the adapted and safe behavior of the complete system.

Macro attributes	**Leaf attributes**
Availability [10], [16]	
Safety [12], [18], [45]	
Security	Confidentiality [58] Integrity [58], [17] Non-repudiation [58] Authenticity [58],[43]
Portability	Adaptability [10] Installability [10] Replaceability [33]
Reliability	Maturity [33], [48] Fault Tolerance [33], [48], [18] Recoverability [33] Consistency [10] Reproductibility/Repeatability [10]
Maintainability [55]	Modularity [36], [48] Reusability [25] Modifiability [33], [25] Analyzability [33] Testability [33], [48]

Table 3: Dependability macro and leaf attributes

There is no single metric that can accurately capture the notion of maintainability of an application. [55] introduced a composite metric for quantifying software maintainability which could be used for ML based software to help predict the maintainability of the application using the Halstead Volume [25] (effort or volume), Cyclomatic Complexity, Total SLOC (source lines of code) and Comments Ratio.

Robustness

Taking into account the recent progress in AI, the negative consequences of its use have led to multiple initiatives from the European Commission to set up the principles of a trustworthy and secure AI. Among the identified requirements, the concept of robustness [26] has emerged as a key element for a future regulation, recognized as a desirable property in systems where the consequences were deemed unacceptable relative to the initiating damage.

The IEEE software engineering glossary [3] defines *robustness* as *the degree to which a system or component can function correctly in the presence of invalid inputs or stressful environmental conditions*. In [40], it is also *the ability of an AI system to maintain its level of performance under any circumstances*.

Macro attributes	**Leaf attributes**
System-Dependent Robustness	Adaptability [33] Flexibility [31] Agility [10] Scalability [10] Evolvability [10]
Inherent Robustness	Global Robustness [12], [26], [75] Local Robustness [26], [48], [75] Stability [10]
Resilience	Recoverability [48] Survavibility [10] Durability [10]

Table 4: Robustness macro and leaf attributes

Robustness matters for a number of reasons. First, trust depends on reliable performance. Trust can erode when an ML system performs in an unpredictable way that is difficult to understand. Second, deviation from anticipated performance may indicate important issues that require attention. These issues can include malicious attacks, unmodeled phenomena, undetected biases, or significant changes in data. Thus, robustness ensures nothing about "correctness" of a model: robust predictions can still be wrong; a very robust model can be completely useless. Then, Model robustness refers to the degree that a model's performance changes when using new data versus training data. Ideally, performance should not deviate significantly. To ensure that a model is performing according to its intended purpose, it's critical to understand, monitor, and manage robustness as part of model risk governance. In addition, robustness can (should) be tested at two levels of possible perturbations as follows [75]: *Local robustness* and *Global robustness*. Local robustness is satisfied by a single data input $x \in D$ of a model M and a given perturbation x' within a neighborhood δ iff $M(x)$ is identical to $M(x')$, in other words: $\forall x',\ d(x,x') \leq \delta \Rightarrow M(x) = M(x')$. Global robustness is satisfied by the set of data D of a model M, considering possible δ perturbations x' for all inputs $x \in D$, and exhibiting smooth convergence of $M(x')$ towards $M(x)$ during classification, in other words: $\forall x, x' \in D,\ d(x,x') \leq \delta \Rightarrow M(x) \rightarrow M(x')$. If the model outputs $M(D)$ conform a dense set allowing a distance metrics $s(.)$, the con-

vergence can be validated for a given $\varepsilon > 0$ satisfying $s(M(x), M(x')) < \varepsilon$. In practice, such post-condition could be difficult or unfeasible to verify depending upon the nature of $M(D)$. Further means are thus needed to understand how perturbations impact misclassification.

Human centred quality and human oversight

Human-centered quality [32] concerns *which requirements for usability, accessibility, user experience and avoidance of harm from use are met. From such perspective, trustworthy AI should possess the properties of usability, and explainability*. People often confuse usability with user experience and ease of use. The term "user-friendly" is often employed as a synonym for usable, though it may also refer to accessibility. Usability describes the quality of user experience across websites, software, products, and environments. Usability is a component of user experience design. Specifically, ML-based systems should not cease operation at inappropriate times (e.g. at times when the lack of output could lead to safety risks), and these programs or systems should be easy to use for people with different backgrounds. Last, but not least, trustworthy AI must allow for explanation and analysis by humans, so that potential risks and harm can be minimized, and human users can remain empowered. In addition, trustworthy ML should be transparent so people can better understand its mechanism.

Usability issues are critical in many AI-based critical systems, where a human works with the system and apply results, and when the AI system serves as the user interface for the user (as with chatbot systems). AI is also applied in some systems to build a computer model of the user (e.g. digital twin), which is then used to help anticipate the user's needs and optimize the interface (as in computer-aided instruction systems and adaptive systems).

Explainability

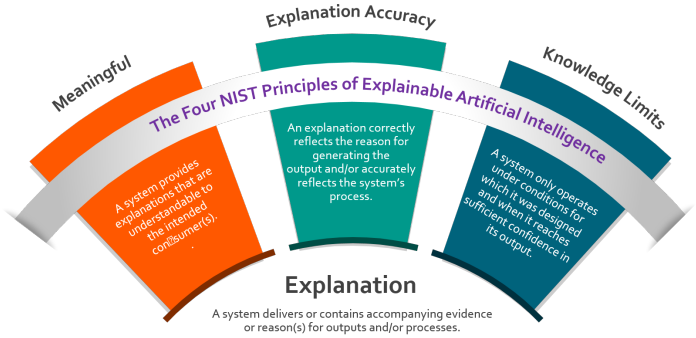

Figure 4: the NIST four principles of XAI

Explainable artificial intelligence (XAI) is a set of processes and methods that allows human users to comprehend and trust the outputs created by machine learning algorithms. XAI is used to describe an AI model, its expected impact and potential biases. It helps characterize model accuracy, fairness, transparency and outcomes in AI-powered decision-making. Some of today's AI tools can be highly complex, if not outright opaque. Workings of complex statistical pattern recognition algorithms, for example, can become too difficult to interpret and the so-called "black box" models can be too complicated for even expert users to fully understand. This is problematic for two reasons: Usability and Regulatory compliance. Therefore, Explainability [48] is a key attribute of Trustworthy AI, which is *the extent to which the behavior of a model can be made understandable to humans*. In basic terms, it is the understanding to the question "why is this happening?". Thus explainability [38] is a *property to express important factors influencing the AI system results in a way that humans can understand.* In line with such assumption, the NIST [59] proposes four principles of explainable AI based on Explanation, Meaningful, Explanation Accuracy, and Knowledge Limits (see fig. 4).

Macro attributes	Leaf attributes
Usability	Understandability [10] Stakeholder satisfaction [35]
Explainability	Explainability [26], [12], [48] Completeness of explainability [10] Precision of Explainability [10] Interpretability [12]
Human oversight	Fairness [10], [44] Inclusiveness [10] Transparency [10] Trust [10]

Table 5: Human centered quality and human oversight macro and leaf attributes

There are no unified methods or scales to evaluate explainability. Recent surveys, as the one offered by [73], suggest that explainability can be decomposed by the methods used to evaluate it.

Visualization methods pursue the characterization of a ML model by visual observation of the levels of activation/deactivation according to the input data and their influence in the classification performance, sensitivity, and other functional / structural properties.

Distillation methods aim to represent (distill) the knowledge encoded in the ML/DL network after training via a more human-readable format suitable for both user interpretation and logic/machine reasoning. Some representative instances in this family are:

- Local Approximation methods mimic the input/output behavior of the target ML model on smaller datasets, and using approximation functions, *e.g.* linear functions. Local Approximation methods are based upon the hypothesis that the ML behavior can be better and more easily characterized on local areas rather than over the entire dataset, *e.g.* LIME [62], Anchors [63].
- Model Translation methods aim to mimic input/output behavior of the target ML model however considering the whole dataset over a symbolic model, *e.g.* Graph-based [76], Rule-based [27].

Intrinsic methods search to integrate the means for ex-

plainability as part of the design of the ML model. The explainability of ML networks should be intrinsic and thus input/output behavior should be explicitly justified by the ML model itself. Representative instances in this family are:

- Attention Mechanisms rely upon contextual vector and attention mechanisms used to learn a conditional distribution over data inputs which provide an interpretation on the behavior of the weights of the operations of activation and deactivation, *e.g.* Single Modal Weighting [30], Multimodal Interaction [2].
- Joint Training consists in introducing an additional task in the ML/DL model, asides the original one, in charge of providing direct or indirect explanations for the main task behavior, *e.g.* Text Explanation [46], Explanation Association [42].

Conclusion

Software quality is defined as the capability of a software product to satisfy stated and implied needs when used under specified conditions. Software quality assurance is then the systematic examination of the extent to which a safety critical software product is capable of satisfying stated and implied needs. AI components, especially based on supervised ML or DL, differ fundamentally from traditional components because they are data-driven in nature, i.e., their behavior is non-deterministic, statistics-orientated and evolves over time in response to the frequent provision of new data. An AI component embedded in a system comprises the data, the ML model, and the framework. Data are collected and pre-processed for use. The learning program is the code for running to train the model. Frameworks (e.g., scikit-learn, TensorFlow) offer algorithms and other libraries for developers to write the learning program.

To characterize AI-based safety critical systems for the purpose of quality assurance, it is meaningful to consider the trustworthiness attributes defined in the report. But, trustworthiness is a complex notion, combining subjective concepts, heterogeneity of granularity in the attributes composing it, and non-commensurability of the different attributes.

Our approach consists in defining the different attributes constituting the notion of trustworthiness, exploring each attribute to determine related KPIs, assessment methods or control points, and defining an aggregation methodology [51]. Some of such KPI examples were illustrated in data-driven AI context. The work envisions the creation of a methodological framework for the assessment of trustworthiness leveraging expert knowledge (for example in the definition of thresholds), a modeling of the environment of the application (e.g. influence of the ODD on the selection of attributes), and usability in an engineering process (each atomic attribute is linked to a method or metric), covering other AI paradigm in order to go beyond ML.

Acknowledgments

This work has been supported by the French government under the "France 2030" program, as part of the SystemX Technological Research Institute within the Confiance.ai Program (www.confiance.ai).

References

[1] ALTAI. 2019. Assessment List for Trustworthy Artificial Intelligence (ALTAI). Technical report, High-Level Expert Group on Artificial Intelligence, European Commission.

[2] Anderson, P.; He, X.; Buehler, C.; Teney, D.; Johnson, M.; Gould, S.; and Zhang, L. 2018. Bottom-Up and Top-Down Attention for Image Captioning and Visual Question Answering. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 6077–6086.

[3] ANSI/ IEEE Std 729-1983. 1983. IEEE Standard Glossary of Software Engineering Terminology.

[4] Avizienis, A.; Laprie, J.-C.; Randell, B.; and Landwehr, C. 2004. Basic concepts and taxonomy of dependable and secure computing. *IEEE transactions on dependable and secure computing*, 1(1): 11–33.

[5] Batini, C.; and Scannapieco, M. 2016. *Data and Information Quality*. Springer International Publishing.

[6] Bosch, J.; Olsson, H. H.; and Crnkovic, I. 2021. Engineering AI systems: A research agenda. In *Artificial Intelligence Paradigms for Smart Cyber-Physical Systems*, 1–19. IGI Global.

[7] Braunschweig, B.; Gelin, R.; and Terrier, F. 2022. The wall of safety for AI: approaches in the Confiance.ai program. In *Proceedings of the Workshop on Artificial Intelligence Safety 2022 (SafeAI 2022)*, volume 3087 of *CEUR Workshop Proceedings*. CEUR-WS.org.

[8] Cho, J.-H.; Xu, S.; Hurley, P. M.; Mackay, M.; Benjamin, T.; and Beaumont, M. 2019. Stram: Measuring the trustworthiness of computer-based systems. *ACM Computing Surveys (CSUR)*, 51(6): 1–47.

[9] Confiance.ai; and all. 2021. Algorithm Engineering including Algorithm evaluation and KPIs - State of the Art. Technical report, Confiance.ai Program.

[10] Confiance.ai; et al. 2022. Towards the engineering of trustworthy AI applications for critical systems - The Confiance.ai program.

[11] Costabel, P.; and del Carmen, V. 2006. Data freshness and data accuracy: A state of the art. *Reportes Técnicos 06-13*.

[12] Delseny, H.; Gabreau, C.; Gauffriau, A.; Beaudouin, B.; Ponsolle, L.; Alecu, L.; Bonnin, H.; Beltran, B.; Duchel, D.; Ginestet, J.-B.; et al. 2021. White Paper Machine Learning in Certified Systems. *arXiv preprint arXiv:2103.10529*.

[13] Demetrescu, C.; Finocchi, I.; and Italiano, G. F. 2004. Algorithm engineering. In *Current Trends in Theoretical Computer Science: The Challenge of the New Century Vol 1: Algorithms and Complexity Vol 2: Formal Models and Semantics*, 83–104. World Scientific.

[14] EASA. 2021. Concept Paper First Usable Guidance for Level 1 Machine Learning Applications.

[15] ED-76A. 2015. Standards for processing aeronautical Data.

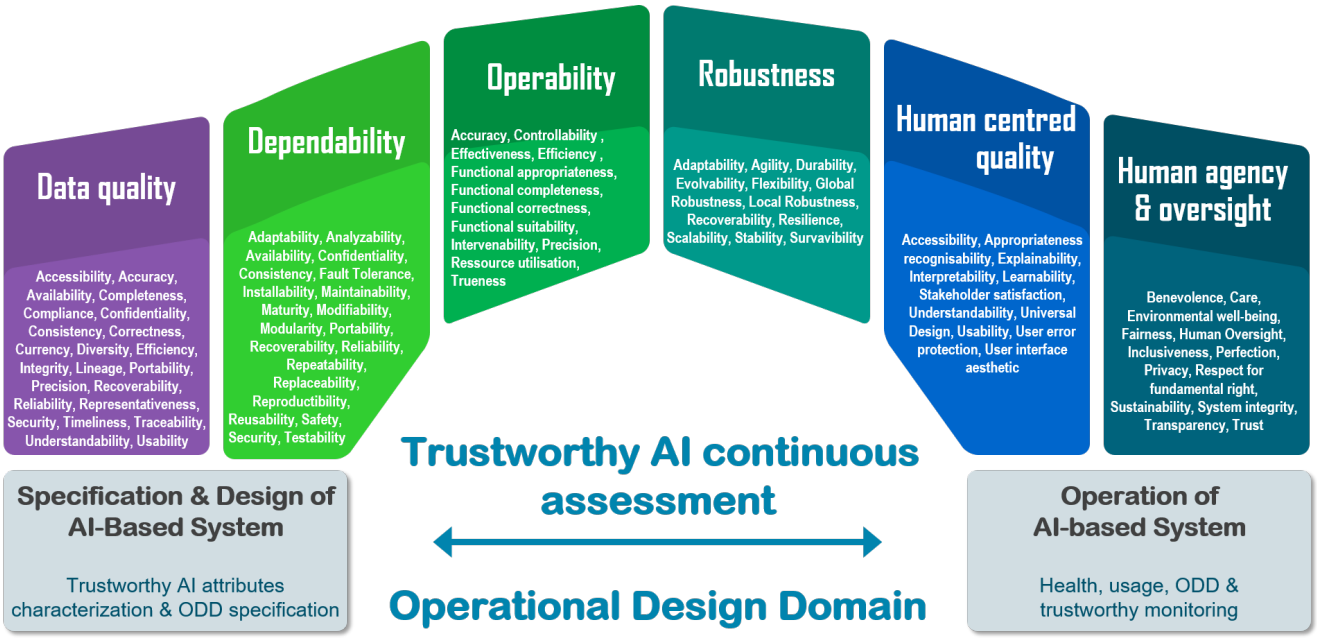

Figure 5: Trustworthiness AI assessment is the cornerstone between AI-based system specification and operation

[16] EN 50129. 2018. Railway applications - Communication, signalling and processing systems - Safety-related electronic systems for signalling.

[17] ETSI. 2021. Securing Artificial Intelligence (SAI); Mitigation Strategy Report.

[18] EUROCAE WG114 – SAE G34. 2021. A joint standardization initiative to support Artificial Intelligence revolution in aeronautics.

[19] European Commission. 2021. Proposal for a Regulation of the European Parliament and of the Council laying down Harmonised Rules on Artificial Intelligence (Artificial Intelligence Act) and Amending Certain Union Legislative Acts.

[20] Fan, W.; and Geerts, F. 2012. *Foundations of Data Quality Management*. Morgan & Claypool Publishers. ISBN 160845777X.

[21] Firmani, D.; Mecella, M.; Scannapieco, M.; and Batini, C. 2016. On the Meaningfulness of "Big Data Quality" (Invited Paper). *Data Science and Engineering*, 1.

[22] Fowlkes, E. B.; and Mallows, C. L. 1983. A method for comparing two hierarchical clusterings. *Journal of the American statistical association*, 78(383): 553–569.

[23] Ge, M.; and Helfert, M. 2006. A Framework to Assess Decision Quality Using Information Quality Dimensions. In *ICIQ*, 455–466.

[24] Gong, Z.; Zhong, P.; and Hu, W. 2018. Diversity in Machine Learning. *CoRR*, abs/1807.01477.

[25] Halstead, M. H. 1977. Halstead. Elements of Software Science.

[26] Hamon, R.; Junklewitz, H.; and Sanchez, I. 2020. Robustness and explainability of artificial intelligence. *Publications Office of the European Union*.

[27] Harradon, M.; Druce, J.; and Ruttenberg, B. E. 2018. Causal Learning and Explanation of Deep Neural Networks via Autoencoded Activations. *CoRR*, abs/1802.00541.

[28] Heinrich, B.; and Helfert, M. 2003. Analyzing Data Quality Investments in CRM-A model-based approach. In *International Conference on Information Quality (ICIQ)*.

[29] Heinrich, B.; Hristova, D.; Klier, M.; Schiller, A.; and Szubartowicz, M. 2018. Requirements for data quality metrics. *Journal of Data and Information Quality (JDIQ)*, 9(2): 1–32.

[30] Hendricks, L. A.; Akata, Z.; Rohrbach, M.; Donahue, J.; Schiele, B.; and Darrell, T. 2016. Generating Visual Explanations. In Leibe, B.; Matas, J.; Sebe, N.; and Welling, M., eds., *Computer Vision – ECCV 2016*, 3–19. Springer International Publishing.

[31] IEEE. 1990. IEEE standard glossary of software engineering terminology. *New York Inst. Electr. Electron. Ehgineers*.

[32] ISO 9241-210. 2019. Ergonomics of human-system interaction — Part 210: Human-centred design for interactive systems.

[33] ISO/IEC 25010. 2011. Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — System and software quality models.

[34] ISO/IEC 25012. 2008. Software engineering — Software product Quality Requirements and Evaluation (SQuaRE) — Data quality model.

[35] ISO/IEC 25022. 2016. Systems and software engineering — Systems and software quality requirements

and evaluation (SQuaRE) — Measurement of quality in use.

[36] ISO/IEC 25023. 2015. Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — Measurement of system and software product quality.

[37] ISO/IEC CD TR 5469. 202X. Artificial intelligence — Functional safety and AI systems.

[38] ISO/IEC DIS 22989. 2021. Information technology — Artificial intelligence — Artificial intelligence concepts and terminology.

[39] ISO/IEC DIS 42001. 2022. Information technology — Artificial intelligence — Management system.

[40] ISO/IEC TR 24029-1. 2021. Artificial Intelligence (AI) — Assessment of the robustness of neural networks — Part 1: Overview.

[41] ISO/IEC TR 29119-11. 2020. Software and systems engineering — Software testing — Part 11: Guidelines on the testing of AI-based systems.

[42] Iyer, R.; Li, Y.; Li, H.; Lewis, M.; Sundar, R.; and Sycara, K. 2018. Transparency and Explanation in Deep Reinforcement Learning Neural Networks. In *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, 144–150. Association for Computing Machinery.

[43] Kapusta, K.; Mattioli, L.; Addad, B.; and Lansari, M. 2023. Protecting ownership rights of ML models using watermarking in the light of adversarial attacks. In *AAAI Spring Symposium - AITA: AI Trustworthiness Assessment*.

[44] Kenya Matsushita CMA, C. 2020. DATA BIAS AND DIVERSITY AND INCLUSION. *Strategic Finance*, 102(6): 16–17.

[45] Koopman, P.; and Wagner, M. 2016. Challenges in autonomous vehicle testing and validation. *SAE International Journal of Transportation Safety*, 4(1): 15–24.

[46] Liu, H.; Yin, Q.; and Wang, W. Y. 2018. Towards Explainable NLP: A Generative Explanation Framework for Text Classification. *CoRR*, abs/1811.00196.

[47] Loshin, D. 2009. Chapter 5 - Data Quality and MDM. In Loshin, D., ed., *Master Data Management*, The MK/OMG Press, 87–103. Boston: Morgan Kaufmann. ISBN 978-0-12-374225-4.

[48] Mamalet, F.; Jenn, E.; Flandrin, G.; et al. 2021. White Paper Machine Learning in Certified Systems. Research report, IRT Saint Exupéry ; ANITI.

[49] Mattioli, J.; Delaborde, A.; and other. 2022. Empowering the trustworthiness of ML-based critical systems through engineering activities. *arXiv preprint arXiv:2209.15438*.

[50] Mattioli, J.; Robic, P.-O.; and Jesson, E. 2022. Information Quality: the cornerstone for AI-based Industry 4.0. *Procedia Computer Science*, 201: 453–460.

[51] Mattioli, J.; Sohier, H.; Delaborde, A.; et al. 2023. Towards a holistic approach for AI trustworthiness assessment based upon aids for multi-criteria aggregation. In *Safe AI*.

[52] Mock, M.; Schmitz, A.; Adilova, L.; et al. 2021. *Management System Support for Trustworthy Artificial Intelligence*. Fraunhofer IAIS.

[53] Nakajima, S. 2018. Quality assurance of machine learning software. In *2018 IEEE 7th Global Conference on Consumer Electronics (GCCE)*, 601–604. IEEE.

[54] NSTC. 2019. *The national artificial intelligence research and development strategic plan: 2019 update*. National Science and Technology Council (US).

[55] Oman, P.; and Hagemeister, J. 1992. Metrics for assessing a software system's maintainability. In *Proceedings Conference on Software Maintenance 1992*, 337–338. IEEE Computer Society.

[56] Ozkaya, I. 2020. What is really different in engineering ai-enabled systems? *IEEE Software*, 37(4): 3–6.

[57] O'Neill, O. 2014. Trust, Trustworthiness, and Accountability. In Morris, N.; and Vines, D., eds., *Capital Failure: Rebuilding Trust in Financial Services*, 0. Oxford University Press. ISBN 978-0-19-871222-0.

[58] Pendleton, M.; Garcia-Lebron, R.; Cho, J.-H.; and Xu, S. 2016. A survey on systems security metrics. *ACM Computing Surveys (CSUR)*, 49(4): 1–35.

[59] Phillips, P. J.; Hahn, C. A.; Fontana, P. C.; Broniatowski, D. A.; and Przybocki, M. A. 2020. Four principles of explainable artificial intelligence. *Gaithersburg, Maryland*.

[60] Pushpalatha, R.; and MeenakshiSundaram, K. 2017. Dice Similarity Based Ensemble Clustering for Sparsely Distributed High Dimensional Data. *International Journal of Applied Engineering Research*, 12(23).

[61] Redman, T. 1996. *Data Quality for the Information Age*. Artech House Telecommunications Library. Artech House. ISBN 9780890068830.

[62] Ribeiro, M. T.; Singh, S.; and Guestrin, C. 2016. "Why Should I Trust You?": Explaining the Predictions of Any Classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 1135–1144. New York, NY, USA: Association for Computing Machinery.

[63] Ribeiro, M. T.; Singh, S.; and Guestrin, C. 2018. Anchors: High-precision model-agnostic explanations. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 1527–1535.

[64] Sanders, P. 2009. Algorithm engineering–an attempt at a definition. In *Efficient algorithms*, 321–340. Springer.

[65] Schmidt, E.; Work, B.; Catz, S.; et al. 2021. National security commission on artificial intelligence (AI). Technical report, National Security Commission on Artificial Intelligence.

[66] Schwalbe, G.; and Schels, M. 2020. A survey on methods for the safety assurance of machine learning based systems. In *10th European Congress on Embedded Real Time Software and Systems (ERTS 2020)*.

[67] Serban, A.; van der Blom, K.; Hoos, H.; and Visser, J. 2021. Practices for engineering trustworthy machine learning applications. In *2021 IEEE/ACM 1st Workshop on AI Engineering-Software Engineering for AI (WAIN)*, 97–100.

[68] Stanton, B.; Jensen, T.; et al. 2021. Trust and artificial intelligence. *preprint*.

[69] Treveil, M.; Omont, N.; Stenac, C.; Lefevre, K.; et al. 2020. *Introducing MLOps*. O'Reilly Media.

[70] van Wynsberghe, A. 2021. Sustainable AI: AI for sustainability and the sustainability of AI. *AI and Ethics*, 1(3): 213–218.

[71] Wang, R. Y.; and Strong, D. M. 1996. Beyond accuracy: What data quality means to data consumers. *Journal of management information systems*, 12(4): 5–33.

[72] Wing, J. M. 2021. Trustworthy ai. *Communications of the ACM*, 64(10): 64–71.

[73] Xie, N.; Ras, G.; van Gerven, M.; and Doran, D. 2020. Explainable Deep Learning: A Field Guide for the Uninitiated. *CoRR*, abs/2004.14545.

[74] Zhang, J. M.; Harman, M.; Ma, L.; and Liu, Y. 2020. Machine learning testing: Survey, landscapes and horizons. *IEEE Transactions on Software Engineering*.

[75] Zhang, J. M.; Harman, M.; Ma, L.; and Liu, Y. 2022. Machine Learning Testing: Survey, Landscapes and Horizons. *IEEE Transactions on Software Engineering*, 48(1): 1–36.

[76] Zhang, Q.; Cao, R.; Shi, F.; Wu, Y. N.; and Zhu, S.-C. 2018. Interpreting cnn knowledge via an explanatory graph. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 4454–4463.

[77] Zinkevich, M. 2017. Rules of machine learning: Best practices for ML engineering. *URL: https://developers. google. com/machine-learning/guides/rules-of-ml*.