TECHNICAL REPORT

ISO/IEC TR 24028

First edition
2020-05

Information technology — Artificial intelligence — Overview of trustworthiness in artificial intelligence

Technologies de l'information — Intelligence artificielle — Examen d'ensemble de la fiabilité en matière d'intelligence artificielle

Reference number
ISO/IEC TR 24028:2020(E)

ISO copyright office
CP 401 • Ch. de Blandonnet 8
CH-1214 Vernier, Geneva
Phone: +41 22 749 01 11
Fax: +41 22 749 09 47
Email: copyright@iso.org
Website: www.iso.org

Published in Switzerland

Contents

Page

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular, the different approval criteria needed for the different types of document should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see www.iso.org/directives).

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see www.iso.org/patents) or the IEC list of patent declarations received (see http://patents.iec.ch).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation of the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT), see www.iso.org/iso/foreword.html.

This document was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information Technology*, Subcommittee SC 42, *Artificial Intelligence*.

Any feedback or questions on this document should be directed to the user’s national standards body. A complete listing of these bodies can be found at www.iso.org/members.html.

Introduction

The goal of this document is to analyse the factors that can impact the trustworthiness of systems providing or using AI, called hereafter artificial intelligence (AI) systems. The document briefly surveys the existing approaches that can support or improve trustworthiness in technical systems and discusses their potential application to AI systems. The document discusses possible approaches to mitigating AI system vulnerabilities that relate to trustworthiness. The document also discusses approaches to improving the trustworthiness of AI systems.

Information technology — Artificial intelligence — Overview of trustworthiness in artificial intelligence

1 Scope

This document surveys topics related to trustworthiness in AI systems, including the following:

— approaches to establish trust in AI systems through transparency, explainability, controllability, etc.;

— engineering pitfalls and typical associated threats and risks to AI systems, along with possible mitigation techniques and methods; and

— approaches to assess and achieve availability, resiliency, reliability, accuracy, safety, security and privacy of AI systems.

The specification of levels of trustworthiness for AI systems is out of the scope of this document.

2 Normative references

There are no normative references in this document.

3 Terms and definitions

For the purposes of this document, the following terms and definitions apply.

ISO and IEC maintain terminological databases for use in standardization at the following addresses:

— ISO Online browsing platform: available at https://www.iso.org/obp

— IEC Electropedia: available at http://www.electropedia.org/

3.1
accountability
property that ensures that the actions of an *entity* (3.16) may be traced uniquely to that entity

[SOURCE: ISO/IEC 2382:2015, 2126250, modified — The Notes to entry have been removed.]

3.2
actor
entity (3.16) that communicates and interacts

[SOURCE: ISO/IEC TR 22417:2017, 3.1]

3.3
algorithm
set of rules for transforming the logical representation of *data* (3.11)

[SOURCE: ISO/IEC 11557:1992, 4.3]

3.4
artificial intelligence
AI
capability of an engineered *system* (3.38) to acquire, process and apply knowledge and skills

Note 1 to entry: Knowledge are facts, *information* (3.20) and skills acquired through experience or education.

3.5
asset
anything that has *value* (3.46) to a *stakeholder* (3.37)

Note 1 to entry: There are many types of assets, including:

a) *information* (3.20);

b) software, such as a computer program;

c) physical, such as computer;

d) services;

e) people and their qualifications, skills and experience; and

f) intangibles, such as reputation and image.

[SOURCE: ISO/IEC 21827:2008, 3.4, modified — In the definition, “the organization” has been changed to “a stakeholder”. Note 1 to entry has been removed.]

3.6
attribute
property or characteristic of an object that can be distinguished quantitatively or qualitatively by human or automated means

[SOURCE: ISO/IEC/IEEE 15939:2017, 3.2]

3.7
autonomy
autonomous
characteristic of a *system* (3.38) governed by its own rules as the result of self-learning

Note 1 to entry: Such systems are not subject to external *control* (3.10) or oversight.

3.8
bias
favouritism towards some things, people or groups over others

3.9
consistency
degree of uniformity, standardization and freedom from contradiction among the documents or parts of a *system* (3.38) or component

[SOURCE: ISO/IEC 21827:2008, 3.14]

3.10
control
purposeful action on or in a *process* (3.29) to meet specified objectives

[SOURCE: IEC 61800-7-1:2015, 3.2.6]

3.11
data
re-interpretable representation of *information* (3.20) in a formalized manner suitable for communication, interpretation or processing

Note 1 to entry: *Data* (3.11) can be processed by human or automatic means.

[SOURCE: ISO/IEC 2382:2015, 2121272, modified — Notes 2 and 3 to entry have been removed.]

3.12
data subject
individual about whom *personal data* (3.27) are recorded

[SOURCE: ISO 5127:2017, 3.13.4.01, modified — Note 1 to entry has been removed.]

3.13
decision tree
supervised-learning model for which inference can be represented by traversing one or more tree-like structures

3.14
effectiveness
extent to which planned activities are realized and planned results achieved

[SOURCE: ISO 9000:2015, 3.7.11, modified — Note 1 to entry has been removed.]

3.15
efficiency
relationship between the results achieved and the resources used

[SOURCE: ISO 9000:2015, 3.7.10]

3.16
entity
any concrete or abstract thing of interest

[SOURCE: ISO/IEC 10746-2:2009, 6.1]

3.17
harm
injury or damage to the health of people or damage to property or the environment

[SOURCE: ISO/IEC Guide 51:2014, 3.1]

3.18
hazard
potential source of *harm* (3.17)

[SOURCE: ISO/IEC Guide 51:2014, 3.2]

3.19
human factors
environmental, organizational and job factors, in conjunction with cognitive human characteristics, which influence the behaviour of persons or organizations

3.20
information
meaningful *data* (3.11)

[SOURCE: ISO 9000:2015, 3.8.2]

3.21
integrity
property of protecting the accuracy and completeness of *assets* (3.5)

[SOURCE: ISO/IEC 27000:2018, 3.36, modified — In the definition, "protecting the" has been added before "accuracy" and "of assets" has been added after "completeness".]

3.22
intended use
use in accordance with *information* (3.20) provided with a product or *system* (3.38) or, in the absence of such information, by generally understood *patterns* (3.26) of usage.

[SOURCE: ISO/IEC Guide 51:2014, 3.6]

3.23
machine learning
ML
process (3.29) by which a functional unit improves its performance by acquiring new knowledge or skills or by reorganizing existing knowledge or skills

[SOURCE: ISO/IEC 2382:2015, 2123789]

3.24
machine learning model
mathematical construct that generates an inference or prediction, based on input *data* (3.11)

3.25
neural network
computational model utilizing distributed, parallel local processing and consisting of a network of simple processing elements called artificial neurons, which can exhibit complex global behaviour

[SOURCE: ISO 18115-1:2013, 8.1]

3.26
pattern
set of features and their relationships used to recognize an *entity* (3.16) within a given context

[SOURCE: ISO/IEC 2382:2015, 2123798]

3.27
personal data
data (3.11) relating to an identified or identifiable individual

[SOURCE: ISO 5127:2017, 3.1.10.14, modified — The admitted terms and Notes 1 and 2 to entry have been removed.]

3.28
privacy
freedom from intrusion into the private life or affairs of an individual when that intrusion results from undue or illegal gathering and use of *data* (3.11) about that individual

[SOURCE: ISO/IEC 2382:2015, 2126263, modified — Notes 1 and 2 to entry have been removed.]

3.29
process
set of interrelated or interacting activities that use inputs to deliver an intended result

[SOURCE: ISO 9000:2015, 3.4.1, modified — The notes to entry have been omitted.]

3.30
reliability
property of consistent intended behaviour and results

[SOURCE: ISO/IEC 27000:2018, 3.55]

3.31
risk
effect of uncertainty on objectives

Note 1 to entry: An effect is a deviation from the expected. It can be positive, negative or both and can address, create or result in opportunities and *threats* (3.39).

Note 2 to entry: Objectives can have different aspects and categories and can be applied at different levels.

Note 3 to entry: Risk is usually expressed in terms of risk sources, potential events, their consequences and their likelihood.

[SOURCE: ISO 31000:2018, 3.1]

3.32
robot
programmed actuated mechanism with a degree of *autonomy* (3.7), moving within its environment, to perform intended tasks

Note 1 to entry: A robot includes the *control* (3.10) system and interface of the control *system* (3.38).

Note 2 to entry: The classification of robot into industrial robot or service robot is done according to its intended application.

[SOURCE: ISO 18646-2:2019, 3.1]

3.33
robotics
science and practice of designing, manufacturing and applying *robots* (3.32)

[SOURCE: ISO 8373:2012, 2.16]

3.34
safety
freedom from *risk* (3.31) which is not tolerable

[SOURCE: ISO/IEC Guide 51:2014, 3.14]

3.35
security
degree to which a product or *system* (3.38) protects *information* (3.20) and *data* (3.11) so that persons or other products or systems have the degree of data access appropriate to their types and levels of authorization

[SOURCE: ISO/IEC 25010:2011, 4.2.6]

3.36
sensitive data
data (3.11) with potentially harmful effects in the event of disclosure or misuse

[SOURCE: ISO 5127:2017, 3.1.10.16]

3.37
stakeholder
any individual, group or organization that can affect, be affected by or perceive itself to be affected by a decision or activity

[SOURCE: ISO/IEC 38500:2015, 2.24]

3.38
system
combination of interacting elements organized to achieve one or more stated purposes

Note 1 to entry: A system is sometimes considered as a product or as the services it provides.

[SOURCE: ISO/IEC/IEEE 15288:2015, 3.38]

3.39
threat
potential cause of an unwanted incident, which may result in *harm* (3.17) to *systems* (3.38), organizations or individuals

3.40
training
process (3.29) to establish or to improve the parameters of a *machine learning model* (3.24) based on a machine learning *algorithm* (3.3) by using training *data* (3.11)

3.41
trust
degree to which a *user* (3.43) or other *stakeholder* (3.37) has confidence that a product or *system* (3.38) will behave as intended

[SOURCE: ISO/IEC 25010:2011, 4.1.3.2]

3.42
trustworthiness
ability to meet *stakeholders'* (3.37) expectations in a verifiable way

Note 1 to entry: Depending on the context or sector and also on the specific product or service, *data* (3.11) and technology used, different characteristics apply and need *verification* (3.47) to ensure stakeholders expectations are met.

Note 2 to entry: Characteristics of trustworthiness include, for instance, *reliability* (3.30), availability, resilience, *security* (3.35), *privacy* (3.28), *safety* (3.34), *accountability* (3.1), transparency, *integrity* (3.21), authenticity, quality, usability.

Note 3 to entry: Trustworthiness is an *attribute* (3.6) that can be applied to services, products, technology, data and *information* (3.20) as well as, in the context of governance, to organizations.

3.43
user
individual or group that interacts with a *system* (3.38) or benefits from a system during its utilization

[SOURCE: ISO/IEC/IEEE 15288:2015, 4.1.52, modified — Note 1 to entry has been removed.]

3.44
validation
confirmation, through the provision of objective evidence, that the requirements for a specific *intended use* (3.22) or application have been fulfilled

Note 1 to entry: The right *system* (3.38) was built.

[SOURCE: ISO/IEC TR 29110-1:2016, 3.73, modified — Only the last sentence of Note 1 to entry has been retained and Note 2 to entry has been removed.]

3.45
value
<data> unit of *data* (3.11)

[SOURCE: ISO/IEC/IEEE 15939:2017, 3.41]

3.46
value
<social> belief(s) an organization adheres to and the standards that it seeks to observe

[SOURCE: ISO 10303-11:2004, 3.3.22]

3.47
verification
confirmation, through the provision of objective evidence, that specified requirements have been fulfilled

Note 1 to entry: The *system* (3.38) was built right.

[SOURCE: ISO/IEC TR 29110-1:2016, 3.74, modified — Only the last sentence of Note 1 to entry has been retained.]

3.48
vulnerability
weakness of an *asset* (3.5) or *control* (3.10) that can be exploited by one or more *threats* (3.38)

[SOURCE: ISO/IEC 27000:2018, 3.77]

3.49
workload
mix of tasks typically run on a given computer *system* (3.38)

[SOURCE: ISO/IEC/IEEE 24765:2017, 3.4618, modified — Note 1 to entry has been removed.]

4 Overview

This document provides an overview of topics relevant to building trustworthiness of AI systems. One of the goals of this document is to assist the standards community with identifying specific standardization gaps in the area of AI.

In Clause 5, the document briefly surveys existing approaches being used for building trustworthiness in technical systems and discusses their potential applicability to AI systems. In Clause 6, the document identifies the stakeholders. In Clause 7, it discusses their considerations related to the responsibility, accountability, governance and safety of AI systems. In Clause 8, the document surveys the vulnerabilities of AI systems that can reduce their trustworthiness. In Clause 9, the document identifies possible measures that improve trustworthiness of an AI system by mitigating vulnerabilities across its lifecycle. Measures include those related to improving AI system transparency, controllability, data handling, robustness, testing and evaluation and use. Conclusions are presented in Clause 10.

5 Existing frameworks applicable to trustworthiness

5.1 Background

For the purposes of this document, it is important to provide working definitions of artificial intelligence (AI) systems and trustworthiness.

Here, we consider an AI system to be any system (whether a product or a service) that uses AI. There are many different kinds of AI systems. Some are implemented completely in software, while others are mostly implemented in hardware (e.g. robots).

A working definition of trustworthiness is the ability to meet stakeholders’ expectations in a verifiable way. This definition can be applied across the broad range of AI systems, technologies and application domains.

As with security, trustworthiness has been understood and treated as a non-functional requirement specifying emergent properties of a system — i.e. a set of characteristics with their attributes — within the context of quality of use. This is indicated in ISO/IEC 25010[20].

Additionally, like with security, trustworthiness can be improved through an organizational process with specific measurable outcomes and key performance indicators (KPIs).

In summary, trustworthiness has been understood and treated as both an ongoing organizational process as well as a (non-functional) requirement.

According to UNEP[26], the "precautionary principle" means that where there are threats of serious or irreversible harm, lack of full scientific certainty shall not be used as a reason for postponing effective measures to prevent harm. In safety engineering, a process for capturing and then sizing, stakeholder "value" requirements includes the understanding of the system's context of use, the risks of harm and, when applicable, an application of the "precautionary principle" as a risk mitigation technique against potential unintended consequences, such as harm to rights and freedom of natural persons, life of any kind, the environment, a species or a community.

AI systems are often existing systems enhanced with AI capabilities. In this case, all the approaches and considerations regarding trustworthiness that applied to the old version of the system, continue to apply to enhanced system. These include approaches to quality (both metrics and measurement methodologies), safety and risk of harm and risk management frameworks (such as those existing for security and privacy). Subclauses 5.2 to 5.5 present different frameworks for contextualizing the trustworthiness of AI systems.

5.2 Recognition of layers of trust

An AI system can be conceptualized as operating in an ecosystem of functional layers. Trust is established and maintained at each layer in order for the AI system to be trusted in its environment. For example, the ITU-T report on Trust Provisioning[27] introduces three layers of trust: physical trust, cyber trust and social trust, taking into account the physical infrastructure for data collection (e.g. sensors and actuators), IT infrastructure for data storage and processing (e.g. cloud) and end-applications (e.g. ML algorithms, expert systems and applications for end-users).

Regarding the layer of physical trust, the concept is often synonymous to the combination of reliability and safety because the metrics are based on a physical measurement or test. For instance, the technical control of a car makes the car and its inner mechanisms trustworthy. In this context, the level of trust can be determined through the level of fulfilment of a checklist. In addition, some processes such as sensor calibration can guarantee the correctness of measurements and, therefore, the data produced.

At the cyber trust layer, concerns often shift to IT infrastructure security requirements, such as access control and other measures to maintain AI system integrity and to keep its data safe.

Trust at the end-applications layer of an AI system requires, among other things, software that is reliable and safe. In the context of critical systems, the production of software is framed by a set of processes to verify and validate the "product"[28]. The same is true for AI systems and more. With the stochastic nature of AI systems based on machine learning, trustworthiness also implies fairness of the system's behaviour, corresponding to the absence of inappropriate bias.

Moreover, social trust is based on a person's way of life, belief, character, etc. Without a clear understanding of the internal functioning, its operating principles are not transparent to the non-technical segment of population. In this case, the establishment of trust might not be dependent on objective verification of the AI system's performance, but rather based on a subjective pedagogical explanation of the AI system's observed behaviour.

5.3 Application of software and data quality standards

Software has an important effect on the trustworthiness of a typical AI system. As a result, identifying and describing the quality attributes of its software can help to improve trustworthiness of the whole system[29]. These attributes can contribute to both cyber and social trust. For example, from a societal perspective, trustworthiness can be described in terms of ability, integrity and benevolence[30]. Below are examples of how these terms are being interpreted today in the context of AI systems.

- Ability is the capability the AI system to do a specific task (e.g. discover the tumour within a medical image or identify a person using face recognition over a video monitoring system). The attributes related to ability include robustness, safety, reliability, etc.
- Integrity is an AI system's respect of sound moral and ethical principles or the assurance that information will not be manipulated in a malicious way by the AI system. Thus, the attributes of integrity include completeness, accuracy, certainty, consistency, etc.
- Benevolence is the extent to which the AI system is believed to do good or in other terms, to what extent the "Do No Harm" principle is respected.

The ISO/IEC SQuaRE series deals with software quality through models and measurement (ISO/IEC 2501x on models and ISO/IEC 2502x on measurement) resulting in a list of characteristics for software quality and characteristics for data quality.

SQuaRE series distinguish between the following models:

- quality of a software product resulting in a list of 8 characteristics;
- quality in use of a software product, data and IT services resulting in a list of 5 characteristics, giving way to differentiate cyber trust and social trust, specifying also possible risks to mitigate;
- data quality, resulting in a list of 15 characteristics; and
- IT service quality, resulting in a list of 8 characteristics.

For example, in terms of ISO/IEC 25010[20], emerging social requirements falls within the category of "freedom from risk". According to [20], freedom from risk is the "degree to which a product or system mitigates the potential risk to economic status, human life, health or the environment".

ISO/IEC 25010 is a part of the SQuaRE series of International Standards and describes a model, consisting of characteristics and sub-characteristics, for software product quality and software quality in use. ISO/IEC 25012[19] is part of the same series and in turn defines a general data quality model for data processed in a structured format within a computer system. ISO/IEC 25012 focuses on the quality of the data as part of a computer system and defines quality characteristics for target data used by humans and systems.

The SQuaRE series have been developed for traditional software systems that store their data in a structured manner and process it using explicit logic. ISO/IEC 25012 describes its data quality model by using 15 different characteristics such as accuracy, completeness, accessibility, traceability and portability.

It can be more challenging to measure both system and data quality characteristics in AI systems. The data quality model in ISO/IEC 25012 does not sufficiently address all of the characteristics of the data-driven nature of AI systems. For example, deep learning is an approach to creating rich hierarchical representations through the training of neural networks with many hidden layers on large amounts of data[31]. In addition, a data quality model for AI systems needs to take into consideration other characteristics not currently described in ISO/IEC 25012 such as bias in the data used to develop the AI system.

To more adequately cover AI systems and the data they depend on, it is possible that there is a need for extending or modifying existing standards to go beyond the characteristics and requirements of traditional systems and software development described in ISO/IEC 25010 and the data quality model described in ISO/IEC 25012.

5.4 Application of risk management

Risk management is a preventative process that helps to ensure that "by design" a specific AI product or AI service is trustworthy throughout its lifecycle. The general process of risk management is defined in ISO 31000:2018[14] and involves identifying stakeholders and their vulnerable assets and values, assessing associated risks with their consequence or impact and making conscious risk treatment decisions based on the organization's objectives and its risk tolerance. Risk according to ISO 31000[14] is defined as "effect of uncertainty on objectives", where an effect is a deviation from the expected and it is measured or assessed in terms of the likelihood of the unexpected event and the level of impact it can have on the stakeholders. Risk management is especially suited to new technologies where the unknown is greater than the known. It is also best suited to deal with situations carrying risk inherently, such as dealing with human errors and malicious attacks. Moreover, risk management helps dealing with uncertainty in the areas where no recognized measurements of quality have been established yet. These characteristics are common to AI systems making them particularly suitable for risk management.

Key concepts from ISO 31000 are presented here to show how they can be applied to AI systems. These include identifying stakeholders, an organization's objectives, control objectives, controls and associated measures. In the case of AI systems, the range of stakeholders can be especially broad and includes not only the organization itself, its partners and customers, but also the human society at large and the environment. Whether a developer, distributor or user of AI systems, an organization's objectives at the top-level would include reputation, security and privacy, fairness and safety. Achieving trustworthiness relies on maintaining all these organizational objectives, which are translated into more tangible control objectives (or risk sources). Control objectives typically correspond to vulnerabilities, pitfalls or anticipated threats.[1)] For AI systems, these would include (but not be limited to) challenges to accountability, new security threats, new privacy threats, improper specification, deficient implementation, incorrect use and different sources of bias. For each of the identified control objectives, a set of possible controls (or mitigations) can be identified. For AI systems, these would include (but not be limited to):

- approaches to transparency;
- new security controls;
- new privacy controls;
- considerations related to robustness and resilience;
- considerations related to the choice and the configuration of ML algorithms;
- considerations related to data; and
- considerations related to system controllability.

The risk management process further takes each control and points it to a set of corresponding guidelines (or measures) to choose from the organization's policy and the circumstances. Once this risk management process framework is established, its proper implementation and correct deployment is subject to continuous testing, review and improvement using different assessment and measurement approaches including (but not be limited to) algorithmic performance metrics and field trials.

5.5 Hardware-assisted approaches

Typical machine learning systems (both training and use) are deployed on common and untrusted off-the-shelf platforms, which can influence the correct execution of the system. For example, machine learning applications are frequently deployed in a multi-tenant cloud. Hardware mechanisms reduce an attack surface by providing trusted execution environments (TEEs), which protect confidentiality and integrity of both the data and computation and for both training and use.

1) Note that the mapping between the organization's objectives and the control objectives may not be one-to-one. For complex systems (such as AI systems), achieving each of the organization's objectives typically requires achieving many of the identified control objectives.

TEEs are used to protect selected code and data from disclosure or modification by providing hardware enforced isolation of programs or protected areas of execution that increase security even on compromised platforms. Using trusted execution environments, developers can protect machine learning model throughout its lifecycle (e.g. its training and use) by effectively treating the model as protected data or intellectual property as needed. TEEs enforce confidentiality and integrity of memory used by ML workload (typically using memory encryption engines) even in the presence of privileged malware at the system software layers.

6 Stakeholders

6.1 General concepts

This document adopts a broad definition of stakeholders from ISO/IEC 38500[17] which, in addition to recognizing individuals and organizations:

- acknowledges a group of people as a type of stakeholder which is important when understanding the collective viewpoints shared by a population of individuals that does not constitute an organization, i.e. in that the group does not benefit from a shared administration to represent that population; and
- encompasses stakeholder types that can be affected by the system, which is relevant but can extend beyond those who have needs or expectations for the systems, which implies some foreknowledge of the system.

This broader definition is important when considering the identification of stakeholders, some of whom can be unaware of the existence, goals or capabilities of the system.

The definition of the term “asset” in ISO/IEC 27000:2018[1] is insufficient when considered in relation to stakeholders as discussed above. Instead, by using the term asset, this document refers to “anything that has value to a stakeholder". This extends the assumption of Reference [1] that only organizations would possess assets of value, whereas this can also be the case for individuals and groups of people.

Values, in the context of this document, are not limited to organizations (as per Reference [23]), but include the beliefs any stakeholder “adheres to and the standards that it seeks to observe”.

Given the possibility of operating AI systems in pliable and dynamic fashion, approaches to the trustworthiness of AI need to focus on both gaining and maintaining trust. This can be achieved by using definitions that provide a clear context for specific characteristics of trustworthy AI, such that a change to the context can trigger a critical re-evaluation of the stated characteristic[32]. In this sense, it is insufficient to simply refer to the "trustworthy AI", but to specify who trusts whom in what aspects of AI development and use. Such stakeholder contextualization can therefore apply to consideration of trustworthy AI characteristics such as transparency (see 9.2), explainability (see 9.3) and controllability (see 9.4). Contextualization requires the careful identification of stakeholders and a clear understanding of their involvement at different points in the AI system lifecycle and value chains.

Different stakeholders can hold differing views of the relative importance of different proposed characteristics for a trustworthy AI. The standardization of terms and a conceptual framework for trustworthy AI would therefore enable clear, unambiguous communication between different stakeholders, so that these differences in view can be understood and resolutions to these differences sought. Such stakeholder communication would address:

- how different stakeholder can be affected by AI technology deployed in a product or service;
- how any assets that are valued by different stakeholder are used or affected by the use of AI in a product or service;
- how the use of AI in a product or service relates to values held by different stakeholders.

6.2 Types

There is not yet a clear consensus on a typology of stakeholders related to the use of AI in products or services. In business, stakeholder theory[33] highlights the benefit of an approach to decision-making that looks beyond the fiduciary obligation of management to generate profits for shareholders and considers benefits to other types of stakeholders in an organization, including: employees, customers, management, suppliers, creditors, government and regulators, society in general and the natural environment (as a proxy representing future generations).

In the context of AI, we can consider such stakeholder types in relation to the following distinct roles in the AI value chain (noting that a single stakeholder can undertake several such roles):

- data source: an organization or an individual providing data that is used to train an AI system;
- AI system developer: an organization or an individual that designs, develops and trains an AI system;
- AI producer: an organization or an individual that designs, develops, tests and deploys a product or a service that uses at least one AI system;
- AI user: an organization or an individual that consumes a product or a service that uses at least one AI system;
- AI tools and middleware developer: an organization or an individual that design and develop AI tools and pretrained AI building blocks;
- test and evaluation agency: an organization or an individual that offer independent testing and possibly a certification;
- the broader society in which the AI system is deployed (as even an accurate AI system can lead to a confirmation of existing inequalities);
- associations representing the viewpoints of individuals;
- governance organizations that monitor and study the usage of AI including national governments and international organizations, such as the International Monetary Fund (IMF).

6.3 Assets

It is possible to characterize stakeholders by the assets they value that play a role in or are affected by the use of AI in a product or service.

Tangible assets specific to AI can include:

- data used to train an AI system;
- a trained AI system;
- a product or service that uses one or more AI system;
- data used to test the AI-related behaviour of a product or service;
- data fed to a product or service operation, based on which AI-based decisions are made;
- computing resources and software used to train, test and operate AI systems;
- human resources with the skills to:
  - train, test and operate AI systems;
  - develop software used in or for those tasks; and/or
  - generate, annotate or select data needed for AI training.

Less tangible assets include:

- the reputation of and trust placed in, a stakeholder involved developing, testing or operating an AI system or the service or product that uses it;
- time, e.g. the time saved by the user of a product or service employing AI or time wasted in reacting to an inappropriate recommendation from an AI system;
- skills, which can become less valued due to automation enabled by an AI system;
- autonomy, which can be enhanced by more efficient provision of task related information by an AI system or which can be eroded, e.g. by persuasive content, such as advertising or political messages, targeted to an individual or a group of individuals, using AI profiling.

6.4 Values

Stakeholders can hold different views on the appropriate characteristics for a trustworthy AI system based on the different values they adhere to or seek to observe. Some proposals for trustworthy AI are grounded in a specific set of values established in a particular policy, such as the European Commission's High Level Expert Group working paper on Trustworthy AI[34], which proposes principles grounded in the European Charter of Fundamental Rights. Different views of trustworthy AI can also result from different moral worldviews or systems of values. The relevance and impact of different worldviews, such as Western Ethics, Buddhism, Ubuntu, Shinto, on AI is still relatively unexplored[35]. More generally, with respect to value-sensitive design[36], a clear understanding of these differences in values is essential in communicating trustworthy AI characteristics at a global level.

7 Recognition of high-level concerns

7.1 Responsibility, accountability and governance

The development and application of AI systems represent an application of IT in a multi-stakeholder environment. To build and maintain trust in such an environment, it is important to define responsibilities and the accountability between stakeholders. As AI systems can exist in both complex international commercial value chains and across trans-national societal frameworks, it is important that all stakeholders share an understanding of the responsibilities they undertake towards other stakeholders and how they will be held accountable for those responsibilities. One of the main reasons for agreeing on such a framework is to be able to define decision-making points across the lifecycle of AI system.

Within an organization, the responsibilities for decisions and accountability for the outcomes of such decisions are typically captured in a governance framework. ISO/IEC 38500[17] guides high-level decision-makers in an organization in understanding and fulfilling their legal, regulatory and ethical obligations in the use of IT. It defines the tasks of evaluating, directing and monitoring aspects of IT in implementing principles of responsibility, strategy, acquisition, performance, conformance and human behaviour. ISO/IEC 38505[37] applies this model of IT governance to the governance of data. It does not specifically address the vulnerabilities of AI but does address some related issues relevant to data-driven AI, such as machine learning. There can be opportunities to further align the IT governance model from ISO/IEC 38500 with the need for trustworthy AI systems, especially in relation to their interaction with other societal decision-making frameworks and to their use in autonomous systems. The term “autonomous system” is used in this document to refer to any kind of systems that operates relatively independently and is not limited to cars or “machinery”.

For example, a doctor can use AI to improve his/her diagnosis. The doctor can be accountable for the diagnosis because he/she is a qualified subject matter expert and, therefore, takes responsibility for factoring the output to an AI system in diagnostic decisions. On the other hand, an end-user applying for a job is not a subject matter expert and can have more difficulty in understanding why his/her application has been rejected. In this latter case, the chain of responsibility needs to be well identified.

To achieve trustworthiness of AI-driven autonomous systems, it is necessary to address responsibilities and accountability in event that an autonomous system fails[38][40]. This allows holding relevant stakeholders legally accountable if an autonomous system causes harm. The European Group on Ethics in Science and New Technologies[41] highlighted in their 2018 statement that an AI-driven system cannot be autonomous in the legal sense and a clear framework of responsibility and legal liability needs to be established to enable recourse for any harm caused by the operation of any autonomous system.

7.2 Safety

Safety is a critical aspect of trustworthiness. Therefore, consideration of safety aspects has high priority. Usually the higher the perceived risks of a system to cause harm is, the higher the demands on trustworthiness are.

AI systems, as any other systems, are expected not to cause any unintentional harm. This includes not only tangible harm (for example, to the health of living beings, to property and to the physical environment), but also intangible harm (for example, to social and cultural environments).

ISO/IEC Guide 51[10] states that "All products and systems include hazards and, therefore, some level of residual risk. However, the risk associated with those hazards should be reduced to a tolerable level. Safety is achieved by reducing risk to a tolerable level which is defined in this Guide as tolerable risk." AI systems, which provided a certain level of autonomy, are often seen as more safety critical. However, the hazards and risks are application-dependent and not necessarily directly related to the level of autonomy (e.g. autonomous road vehicle vs. autonomous household vacuum cleaner).

The complete lifecycle of an AI system, from its design to its disposal, becomes subject to consideration for safety aspects. The AI system application, its intended use and reasonably foreseeable misuse, as well as the environment in which it is used, and the technologies that are used, become subject to careful consideration. ISO/IEC Guide 51[10] defines "reasonably foreseeable misuse" as the use of a product or system in a way not intended by the supplier, but which can result from readily predictable human behaviour. It is possible for AI systems to introduce new risk due to AI specific vulnerabilities. This would lead to new measures to reduce the risk to a tolerable level due to AI specific behaviour such as non-transparent or non-deterministic decision processes. For a specific system, not only the AI part, but all used technologies and their interaction are subject to careful consideration. ISO/IEC Guide 51[10] provides general guidance on achieving tolerable risks.

8 Vulnerabilities, threats and challenges

8.1 General

This clause describes potential vulnerabilities of AI systems and the threats associated with them. Vulnerabilities is defined by ISO/IEC 27000[1] as weakness of an asset or control that can be exploited by one or more threats. Threats is defined by ISO/IEC 27000[1] as potential cause of an unwanted incident, which may result in harm to a system or organization.

Different stakeholders use different terms to describe the concept of "vulnerability". These include risk sources, pitfalls, sources of failures, root causes and challenges.

Many vulnerabilities are related to the use of machine learning in AI systems. These include dependency on data, opaqueness of ML models and unpredictability. Specifically, the use of data can lead to new security threats and biased results.

Challenges related to the lack of "best practices" for design, development and deployment of AI systems can introduce additional or exacerbate existing vulnerabilities and threats.

Certain threats arise from the insufficient understanding of the technological capabilities of AI systems and their unfitting use by different stakeholders.

Subclauses 8.2 to 8.10 describe the potential AI systems' vulnerabilities, threats and other challenges in more detail.

8.2 AI specific security threats

8.2.1 General

The development of AI has held both advantages and disadvantages for digital security. On one hand, AI technologies can be used to profile attackers and their malicious activities and then to devise security solutions for fighting them. On the other hand, advanced technology developed using AI and machine learning can be misused for malicious purposes. For instance, AI can be used to guess passwords and compromise digital accounts.

In addition to common IT security threats applicable to most systems (e.g. software bugs, hardware backdoors, data security breaches), certain AI systems, such as machine learning systems, can be vulnerable to specialized or targeted security threats. Such threats include the following[43]:

- data poisoning that results in a malfunctioning AI system;
- adversarial attacks that abuse a benign AI system; and
- model stealing.

8.2.2 Data poisoning

In data poisoning attacks, attackers deliberately influence the training data to manipulate the results of a predictive model. Data poisoning attacks aim at letting the server train a bad model, which cannot detect the malicious attacks. This type of attack is especially valid in case the model is exposed to the Internet in an online mode, i.e. the model is continuously updated by learning from new data. Proper filtering of the training data can help to detect and filter out anomalous data and thus minimize the possible damage.

8.2.3 Adversarial attacks

One particular security threat associated with AI systems is the adversarial attack on the machine learning systems. An adversarial attack consists of providing slightly perturbed input data to a valid model, e.g. a slightly modified traffic sign to an autonomous vehicle system, in an attempt to trick it into misclassifying this input data.

The recent impressive advances in the field of ML, especially in the area of deep learning, have led to an increased interest in companies to apply ML algorithms in safety-critical and security-critical application contexts. Examples include the integration of semantic segmentation based on convolutional neural networks (CNN) into autonomous cars or into neural networks for medical diagnosis. Obviously, these new application contexts have extremely high demands on quality and quality assurance. Most of the time, it is not enough to prove high accuracy on well-crafted training, test and validation data sets. Besides handling the common cases of the given data distribution, it is necessary that the trained ML module is able to deal with rare corner cases or even with maliciously crafted input points[44].

The publications of Szegedy et al.[45] and Goodfellow et al.[46] have shown that computer vision ML algorithms, in particular deep neural networks, can be susceptible to attacks based on adversarial examples (or adversarial perturbations).

These adversarial perturbations are created by an adversary with the help of an adversarial attack and often imply erroneous behaviour of the considered ML module when added to an ordinary input point. Furthermore, these adversarial perturbations are often hard to detect or even imperceptible to the human eye. The imperceptibility is not only challenging for the desired deployment in safety- and security-critical industries, but also hints at a crucial difference between the sensory information processing in humans and in artificial neural networks[47]. Since the discovery of this vulnerability, a lot of different adversarial attacks and defences (defence strategies) have been published[48][49]. It has become an arms race between attackers and defenders. Newly published defences against a set of existing attacks are often rendered useless within a few weeks due to the creation of stronger attacks[50]. While such attacks are difficult to generalize, and they rely on close knowledge of the typically obscured internal network topology of production systems, they nonetheless warrant serious consideration from trustworthiness perspective.

8.2.4 Model stealing

Affecting both security and privacy, model stealing attacks are used to “steal” models by replicating their internal functioning. This is done by sending to the targeted model a high number of prediction queries and using the response received (the prediction) to train another model.

8.2.5 Hardware-focused threats to confidentiality and integrity

Machine learning applications are frequently subject to similar attacks as other sensitive applications. Typical software and hardware attacks on machine learning applications are digital attacks affecting confidentiality of the data and integrity of data and computation. There are other forms of attacks leading to denial of service (loss of availability) or causing leakage of information or leading to invalid computation.

Ensuring confidentiality and integrity of data and code, via traditional mechanism such as memory integrity and trusted platform modules (included in TEEs) is necessary but not sufficient to ensure confidentiality and integrity of the code and data of the machine learning engines – enforcing that the execution of the ML programs follows the programmed-intended logic is equally critical. For example, a control-flow attack on a ML application can defeat/circumvent ML model inference or can cause invalid training. A second category that is critical to enforce runtime integrity are mechanisms to prevent memory safety bugs. Logic flaws in programs can be leveraged for buffer overflows, use-after-free, out-of-bounds exploits which can lead to faulty operation of ML applications.

Threats to complex device models, including hardware accelerators, that can be used by machine learning applications need to be considered. In many cases, these accelerators or devices can be para-virtualized or emulated and in some cases cloud-based applications can benefit from using devices directly assigned to them. However, verification (via attestation) of such devices would help to ensure that the device is capable of upholding the privacy and security requirements of the ML applications. Hardware input/output (IO) memory management capabilities can be used to securely bind devices to workloads including DMA into protected memory. Future attack vectors that need attention include device spoofing, runtime memory remapping attacks and man-in-the-middle attacks.

8.3 AI specific privacy threats

8.3.1 General

The evolution of AI algorithms and the usage of big data have provided sophisticated solutions in many areas. Many AI techniques (e.g. deep learning) highly depend on big data since their accuracy relies, in part, on the amount of data they use. The misuse or disclosure of some data, particularly personal and sensitive data (e.g. health records) can have harmful effects on data subjects. Thus, privacy protection has become a major concern in big data analysis and AI. The challenge starts from the early stage of the data lifecycle (i.e. collecting and sharing data among different entities) to the last stage of data analysis and applying AI algorithms (e.g. risk of re-identification after analysing data from multiple data sources).

These privacy threats can result in negative affects to self-determination, dignity, freedom and fundamental rights of individuals.

8.3.2 Data acquisition

During data acquisition, the privacy threat is mainly about the amount of data to collect for the given purpose. One of the privacy principles, introduced in ISO/IEC 29100[51] is the data minimization principle. Given the dependency of machine learning models on the availability of large amounts of rich quality data, preferably from a variety of data sources, it is challenging to limit the data acquisition.

In addition, the other privacy threat comes from the risk of the storage corruption. If an attacker compromises the storage, the privacy of the data subjects can be breached.

8.3.3 Data pre-processing and modelling

There exist potential privacy threats while processing the data:

- the inference of sensitive data from non-sensitive data by using machine learning and AI techniques;
- personal data is available from multiple sources. Although the data is de-identified, it is possible that AI re-identifies data using the inferences based on the data from the other sources.

8.3.4 Model query

Model stealing by querying the model for inappropriate reasons is described in 8.2.4. Such security attacks are designed to expose confidential information represented by a ML model. This type of attack can be used to expose sensitive information about individuals turning it into a privacy attack.

Such attacks can happen throughout the whole model's lifecycle, including its development, deployment and operation. The attacks can be performed by both actors that are authorized to query the model and others that need to breach security to access to the model first.

Another threat is related to the inappropriate use of the model that was not accepted by individuals, such as profiling, sorting or classifying them, that can affect their social life (e.g. social services, credit cards).

8.4 Bias

Bias is defined as favouritism towards some things, people or groups over others. Bias typically arises from sources including human cognitive bias, societal bias and statistical bias (e.g. selection bias, sampling bias, coverage bias) or simply technical errors. Bias manifests in different stages of the development of an AI system and can take the form of data bias affecting the labels, the training data sets, missing features/labels, data processing issues or as architectural issues affecting models or combinations of models. Bias can also manifest as automation bias, that is, the over-reliance on the recommendations of AI systems. The effects of bias in the data can affect the model and lead to undesirable outcome that can reach from a decreased accuracy up to a complete misclassification for classification tasks. Removing these biases is not always possible and can produce wrong results. Bias caused by the training data set is often based on an incorrect application or disregard of statistical methods and rules.

Evaluation of bias requires that metrics are defined and measured for system performance in the context of specific groups of objects. Numerous metrics are available that are specific to this goal. However, it is essential to use a great care in selecting the metric to be used, as complex trade-offs can result in unintended outcomes. Examples are available where efforts to compensate for bias for a specific group (of objects) have led to an increase in bias in the context of another group.

Many examples of causes of bias are available, with related manifestations in AI systems and associated mitigation measures. The detailed description of this complex topic is subject to a detailed technical report that is under development.

8.5 Unpredictability

Predictability plays an important role in the acceptability of AI systems. The notion of predictability corresponds to the human capacity to infer the next actions of an AI system in a given environment. Trust in technology is often based on predictability: a system is trusted if it is possible to infer what the system does in a particular situation, even if one cannot explain why it is doing it. Conversely, trust would be reduced if a system operates unpredictably in familiar scenarios.

In an operating environment where the AI system interacts with humans and where human safety depends on that interaction, system predictability is necessary, not just desirable. The use of AI in autonomous vehicles is an obvious use case, as the widespread adoption of AI-based autonomous vehicles is likely predicated on the ability of such vehicles to behave in a predictable fashion. Similarly, for the acceptability of collaborative robots in direct interaction with humans, human operators need to be able to predict robot behaviour to ensure operators' safety[55].

In the case of human-driving cars, our cognitive mechanisms make fast and almost unconscious judgments about the likely actions of people and objects around us on the basis of experience, repetition and exposure to similar scenarios. Even slight changes in external behaviours can result in a level of unpredictability at odds with our experience. For instance, a self-driving car can collide with a non-autonomous car because of the inability of a human driver to discern the future actions of the autonomous car and vice versa.

Machine learning algorithms present specific challenges with regards to predictability compared to more traditional programming techniques. ML algorithms learn by analysing large amounts of data and discovering new patterns and solutions. The composition of training data and variations in underlying models in which patterns are realized, establishes parameters for the range of inputs that AI systems are capable of processing correctly. Scenarios that confuse the ML model can result in unpredictable behaviour, absent fail-safes or other override mechanisms. Further, in the case of continual learning or lifelong learning, the "logic" on which ML systems make decisions can evolve over time, introducing additional algorithmic unpredictability.

8.6 Opaqueness

Artificial Intelligence systems can exhibit many forms of opaqueness. Firstly, the AI model itself can be technically opaque, in that the decision procedures it uses are not easily interpretable by humans due to their nature. Secondly, if the data and data sources are not transparent, the behaviour of the whole system becomes opaque to an outside observer. Thirdly, an AI system is always implemented in a context of organizational practices, such as data collection, management, operationalization of AI results and system development. If these practices are undisclosed, even an interpretable AI model becomes an opaque system to users and other external stakeholders.

For a discussion of mitigation measures, see 9.2.

8.7 Challenges related to the specification of AI systems

Most failures of a product originate the specification phase. Because this phase defines the complete product, including its capabilities and environment and its output serves as input to the implementation phase, failures made in this phase have a large impact and are difficult or impossible to correct in later phases of the product lifecycle.

Errors in this phase occur especially when the environment of the product has not been analysed completely or in enough detail. A complete analysis includes all environmental influences that can have an effect on the intended functionality of the product, but also the observance of safety and security threats as well as an investigation of the legal, regulatory and ethical framework of the product. In addition, it is important to consider performance and usability aspects for the intended use, taking into account any changes to the deployment environment as well as different user groups.

For the analysis of potential hazards and risks from AI systems based on methods of machine learning, it is essential to consider a failure of the system caused by bias in the training data or by erroneous training of the algorithm. In addition, risks can be avoided by observing the special features of the methods used later and formulating the task accordingly.

Attribution of risk and legal responsibility within legal systems is a complex task. There is a possibility that by using AI, the attribution process becomes more difficult or it creates changes in how we appreciate risk/responsibility.

Since AI systems are used in to solve complex tasks in heterogeneous environments, the creation of a complete and correct specification is essential. Specifying the aims and the level of explainability (for more detail, see 9.3) would be an important component of any AI system specification.

Once the specification has been defined, it needs to be communicated to the individual actors involved in the project. Inconclusive definitions of the specification are therefore one more source of failures as this increases the risk of misinterpretations and the falsification of the specification through multiple informal transmissions. In general, the mental concepts that are written down in the specification are interpreted by the person who has to create a system on the basis of this specification. This creates a first mismatch between the ideal specification and the final design. Further mismatches are created when using machine learning as the final algorithm is created by concepts that are derived by a training process out of data.

The sum of these uncertainties makes it necessary that the specification contains verifiable and validatable requirements against which the resulting product would be tested.

8.8 Challenges related to the implementation of AI systems

8.8.1 Data acquisition and preparation

In the data acquisition phase, the data sources necessary to solve the problem are identified. Depending on the complexity of the problem to solve, it can be a challenge to find a representative set of data. In case data is coming from multiple sources, normalization or weighting issues can arise. Importantly, defects in handling of the data sources (which are an input to the model) cannot be detected during model evaluation, as it is often conducted in isolation rather than in an integrated environment.

At this point, the data set or sets used to train the models are checked and documented for how well they represent the data that the model will be run on.

After finding the necessary data, a set of tasks, often called data preparation tasks, are performed in order to clean the data and put it into the right format to be exploited by the model. An important aspect in this phase is to verify the quality of the data (e.g. missing data, duplications, inconsistent data or data in the wrong format).

8.8.2 Modelling

8.8.2.1 General

In the modelling phase, selected algorithms are trained in order to generate the candidate models. A model is the representation of a machine learning algorithm when trained with data. Indeed, it is a best practice to build several models and then select the one that performs best for the particular business problem under analysis. In order to generate an accurate model, a common technique is to split the available data into three groups: training data, validation data and test data. The training dataset, as the term suggests, is the portion of data that is used to train the model and ensure it is fit. The validation dataset is used in a following step to validate the predicting capacity of a trained model and to tune the model. Finally, the test dataset is used during the testing phase in order to provide a final evaluation of the trained, fit and tuned model.

With today's technology, building a machine-learning AI system involves the phases described in 8.8.2.2 to 8.8.2.5 often iteratively.

8.8.2.2 Feature engineering

In machine learning, a feature is an input variable that is used by the model to make predictions. Feature engineering is the process of transforming raw data into features that best represent the underlying problem to the predictive models, resulting in improved model accuracy on unseen data[56]. Features are created via a sequence of data transformation steps (e.g. rescaling, discretization, normalization, data mapping, aggregations and ratios) usually involving some programming. Given that the features can be the result of several data transformation steps, the link with the original raw data can be difficult to reconstruct unless the process has been carefully documented.

Feature engineering heavily impacts the performance of the model, in a positive or in a negative way. For example, a single feature that contributes in a predominant way to the prediction of the model can affect the robustness of the model, since the final prediction strongly depends on the value of only that variable instead of being linked proportionally to all features. This can lead to inaccurate results.

8.8.2.3 Model training

Target leakage (also called data leakage) occurs when the training dataset contains some information related to the variable being predicted (target variable), that would not be the case in production. This can happen, for example, when the training data includes information that is not available at the time of the prediction (since the corresponding variable/feature is updated only after the target value is predicted). This can also occur when the target variable is inferred from the input data, through a proxy variable which cannot be included as a feature. Models with target leakage tend to be very accurate during evaluation but perform poorly in production.

The algorithm selected needs to be trained using the training data in order to build the model. The challenge related to this phase is building a model that provides a good representation of the training data with respect to the problem being solved or a model that is fit. In the training process, there is a risk of creating an overfit or underfit model. An overfit model is a model that has learned too many details and is so tightly fit to the underlying data set (including its noise or inherent error in the dataset), that it performs poorly at making predictions when new data comes in. This problem often happens when too many features have been selected as input for the model. On the other hand, underfitting occurs when the model has not captured the underlying patterns in the data and is therefore too generic for good predictions. This occurs more frequently when the model does not have enough relevant features. Therefore, in order to avoid becoming too specific (with too many features) or too vague (with not enough features), it is important to select the right features with the right amount of predictive information. Fitting the model is a challenge that arises during the training phase. However, the quality of predictions is measured during the validation and back-testing phases.

Other approaches to model selection and development include transfer learning, which aims to leverage knowledge of one task to learn a new task and federated learning, which aims to learn new models in a distributed and collaborative manner.

8.8.2.4 Model tuning and hyper-parameter optimization

During the training phase, the models are calibrated/tuned by adjusting their hyperparameters. Examples of hyperparameters are the depth of the tree in the decision tree algorithm, the number of trees in a random forest algorithm, the number of clusters *k* in the *k*-means algorithm, the number of layers in the neural network, etc. Selection of incorrect hyperparameters can be a source of failures of prediction models.

The quality of a model not only depends on its structure, the training algorithm and the data, a crucial factor is also the choice of the model's hyper-parameters. In some applications, the optimization of hyper-parameters has advanced the state of the art more than the learning algorithms. Hyper-parameter optimization usually forms an outer loop of the learning process.

8.8.2.5 Model validation & evaluation

After having tuned the models, these are evaluated against the validation datasets, in order to check their performance on data that is different from the dataset used for training. The simple model validation technique uses only one validation dataset. However, in order to build more robust models, a K-fold cross validation technique can be used. This technique consists of dividing the data into *k* subsets, each one of which is used as the validation set while the other *k*-1 subsets are combined to form the training set. The results of the *k* validation tests are compared to identify the highest performance and robust model (in terms of sensitivity to the noise in the training data). The selection of the model is based on its performance compared to other models. Examples of statistical metrics that are used to evaluate the model performance are the ROC AUC (area under curve), the confusion matrix (which compares the predicted values with the actual values from the test dataset) or the F-1 score (which is calculated based on the confusion matrix and represents the ideal cut-off between precision and recall).

In a separate back-testing phase, the model that has been selected after the modelling phase is once again tested with new data (the testing dataset) for final consistency. Some final settings to tune the model (like the cut-off threshold in classification problems, which defines the probability to fall in one class or the other and therefore the trade-off between false positives and false negatives) are defined together with the business users since they depend on the specific business application.

Production deployment typically takes place after the back-testing phase.

8.8.3 Model updates

After a model has been deployed into production, it can require an update based on the newly acquired data. It is important to continuously monitor the performance/accuracy of the model in order to promptly identify when the model needs re-training/updating. The model updates generally aim to make the model more robust and/or generalize the model to different tasks or to improve its accuracy against new data sets.

One straightforward method to update the model is to simply use both the initial and new data to retrain the model. There can be challenges with this approach, such as collecting all data in a central location, as well as high computation to retrain a new model based on a larger and growing volume of data. One approach to these challenges is incremental learning, in which existing models are extended based on new data. In general, data-efficient and computationally efficient algorithms to update and extend models based on new data is an active area of research.

The main risk to be aware of when updating a model is the impact on performance. The updated model would be validated and back-tested to ensure that there is no degradation relative to earlier performance on the initial task and that the performance on any new tasks is suitable for the specific business application. Moreover, it is important to ensure the full traceability and auditability of the different versions of the models deployed in production.

8.8.4 Software defects

Methods of artificial intelligence are based on the implementation of algorithms in software. Because of this, the development process shares the same pitfalls with every other software development process. Software defects can occur such as erroneous memory accesses and memory handling, erroneous inputs and outputs and erroneous data and control flows. As AI algorithms often require substantial computational resources, they are often implemented on multi- core systems. In these cases, concurrency bugs, such as race conditions, deadlocks and measuring effects ("Heisenbugs"), would also be considered.

8.9 Challenges related to the use of AI systems

8.9.1 Human-computer interaction (HCI) factors

There are many pitfalls that are based on human relating factors. According to Reference [57], it is possible to group these factors into four main categories:

1) use, when automation enables humans to achieve their goals;

2) misuse, when over-reliance on automation perpetrates an unforeseen negative outcome. For example, misuse would be the individual being too reliant on automation and not paying attention to the road;

3) disuse, when under-reliance on automation perpetrates a negative outcome. For example, disuse would be the individual overriding the correctly behaving automation system and causing an accident;

4) abuse, when an automated system is set up without adequately respecting the end user's interests. For example, abuse would be design that does not allow the individual to easily override the automated system.

8.9.2 Misapplication of AI systems that demonstrate realistic human behaviour

AI systems can be designed to impersonate or emulate human characteristics and behaviours, such as handwriting[58], voice[59] and spoken or textual conversations[60][61]. If misapplied by bad actors, these technologies can be used to deceive individuals. There are cases where chatbots or email-bots[62] emulated humans to create the illusion of real human membership in a dating service.

8.10 System hardware faults

Hardware for AI systems needs to have robust fault tolerance. Faults in the hardware can be the source of violation of the correct execution of any algorithm implementation by corrupting both its control and data flow. In the case of AI, faults in hardware interfere with the correctness of algorithmic execution of both inference and training.

Evidence of hardware faults varies and can include errors such as data corruption, data loss or temporal data flow issues. Such errors can be attributed to a single issue or a combination of different types of failures and need to be further investigated in the context of AI.

Specifically, hardware faults can be permanent (permanent issue of a component or module in a system), transient (temporary malfunctions which disappear) or intermittent (ongoing intermittent issues) in nature. Hardware faults can also be benign or malicious resulting from random or systematic causes.

Faults that causes a unit to stop functioning can be benign classic hardware faults due to defective components. Far more insidious are faults that cause a unit to produce reasonable looking but incorrect, outputs or that cause a component to "act maliciously." These faults are soft errors – unwanted temporary state changes of memory cells or logic components that are usually caused by high-energy radiation from sources such as alpha particles from package decay, neutrons and external EMI effects like electromagnetic noise and electromagnetic beams, but can also be caused by internal cross talk between conductor paths or component parts or malicious injection of perturbations such as clock glitches.

Faulty drivers can introduce another dimension to hardware faults in computing based on erroneous software.

The sources and impact of such errors depends on both the AI applications and its deployment on the specific class of systems. AI applications are deployed on systems that range from end devices, mainly used for inference, to cloud-class compute and storage resources, used for both inference and training. In the lifecycle of an AI application, a trained model undergoes a number of transformations that specializes the AI applications to the specific system platform, e.g. a resource constrained end device.

Different fault models would account for the possible sources of error and subsequently for effective fault tolerant strategies. For example, real-time distributed systems tend to incorporate off-the-shelf hardware parts (e.g. general-purpose CPU, GPU and FPGAs), software (e.g. general purpose operating systems) and protocols (e.g. TPC/IP stack based protocols) to run AI applications. Sources of errors include data corruption, unintended repetition of messages, an incorrect sequence of messages, data loss, unacceptable delays, insertion of messages. In systems supporting asynchronous thread scheduling in hardware, such as GPUs, errors affecting the thread scheduler can have large effects, e.g. related to failing to meet system workload deadlines. Furthermore, the diagnosis of the memory can be difficult due to the specific system architecture aspects.

Other sources of errors that can influence the system operation are related to the AI applications lifecycle. For example, additional sources of errors occur during model transformation – when a trained model is mapped to an end device, e.g. a resource constrained embedded system, for example due to the need of specializing data or pruning the model.

9 Mitigation measures

9.1 General

Mitigation measures are the possible controls and guidelines that can mitigate known AI vulnerabilities described in Clause 8. Note that a certain control or a guideline can contribute to mitigation of several vulnerabilities.

A system behaving in a non-reliable way would not be considered as trustworthy. In some cases, the system can be functioning correctly, but producing corrupted output because of newly introduced incorrect input data. In this context, there would be control points to determine whether the trust is maintained or not. Such control points can be done regularly throughout AI system lifecycle or at the moments when the AI system is used for decision-making.

9.2 Transparency

Transparency provides visibility to the features, components and procedures of an AI system. Ideally, a transparent AI system would exhibit repeatable behaviour. Transparency involves making data, features, models, algorithms, training methods and quality assurance processes available for external inspection. Transparency enables stakeholders to assess the development and operation of an AI system against the values they wish to see upheld by AI processing. These values can be based on goals of fairness or privacy or can be derived from a particular stakeholder's ethical worldviews, such as virtue ethics or other global value system.

Integrating transparency into all levels of the AI processes helps ameliorate the problems caused by issues of opaqueness described in 8.6. A transparent AI system informs the stakeholders where, why and what data are collected, especially personal data, provided such metadata was captured at the time of data collection. It can also inform stakeholders when decision-making is automated and explain the processing through which decisions are made. When processing personal data that result in decisions with a legal effect on the stakeholder, privacy regulations can require a transparent AI system to accept requests for human intervention in decision-making and thereby account for stakeholder views on that process. There are several levels and features of transparency which are defined for the development of different AI systems, e.g. in the field of open data.

The use of rating symbols, icons or marks for AI systems can help improve transparency for specific stakeholder groups. For example, the World Economic Forum and UNICEF joint initiative Generation AI[63] suggests rating symbols for AI system used in children toys that is accessible to parents/carers. Explainability of the system is important in achieving such transparency.

Transparency of AI systems relates to making the data, features, algorithms, training methods and quality assurance processes available to external inspection by a stakeholder. In addition, the level of background knowledge of the stakeholder needs to be factored into the planning of how inspections are facilitated. It can, but not necessarily would, include an explanation of:

- how the AI mechanism under inspection works in general, e.g. how decision tree induction works;
- what model class and parametrizations are used;
- what particular variables or features are used by the model; or
- how a set of candidate variables or features were selected.

For a trained expert in machine learning, a short summarising explanation would suffice, specifying the model choice and variable selection procedure. For a layperson, an introductory course in AI and data-driven model inference would be essential for example, as well as an explanation how decision tree models work, in addition to the impact of parametrization and the selection of features and variables.

The key question of transparency is how it works and whether the algorithms that have been used are suitable for purpose. Transparency makes the data, features, algorithms and training methods available to external inspection. Transparency measures would aim to complement privacy and business interests to enhance the overall trustworthiness of AI[64]. In case of opaque models, technical methods can exist to produce some degree of transparency or explainability for such a model[65].

There might be no strict correspondence between the transparency and the explainability of an AI system and the degree of trust a stakeholder would place in that system. However, transparency and explainability provide important evidence and information, which helps stakeholders to form a judgement on their trust in an AI system.

9.3 Explainability

9.3.1 General

Although the explainability alone is not sufficient to guarantee the transparency of an AI system, it is an important component of a transparent AI system. Explanations of processes relevant to the development, implementation and use of an AI system, such as data gathering practices, self-auditing processes, value commitments and stakeholder engagement also play a role. Explainability of AI systems can be regarded as a sub-form of corporate transparency within corporate social responsibility.

It is possible to categorize the explanations of AI systems according to the aims of the explanation, including the context, the needs of stakeholders and types of understanding sought and by the mode of explanation.

9.3.2 Aims of explanation

An explanation is always an attempt to communicate understanding. The effectiveness of an explanation can be improved by tailoring its form to the context in which it is given, including the intended audience and the level of understanding it aims to convey[66].

An attempt to explain can offer multiple different, but equally valid modes of explanation, depending on whether stakeholders seek:

- a causal understanding of how a result is arrived at;
- an epistemic understanding of the knowledge on which the result is based; or
- a justificatory understanding of the grounds in which the result is offered as being valid.

The subject of explanation can include the AI system itself and the result produced by the system.

Explainable AI systems would aim to provide an understanding of the processes contributing to the truth, accuracy and reasonableness of its results beyond the inductive observation that the systems seem to work. The understanding of explanations by stakeholder can be assisted by adherence to appropriate guidelines and standards.

Explanations related to AI systems can also provide justification for the validity, appropriateness and legitimacy of its results and the decisions and actions taken on those grounds. Such explanations would aim to make an AI system more scrutable and contestable, especially for the stakeholders impacted by resulting decision and actions.

Explanations are not absolute but would be defined relative to a target model and the recipient of the explanation. Explanations are understood to be contrastive. Information would be presented to a human in such a way as to improve the fidelity of that human's mental model of a system to the system itself[66][71].

9.3.3 Ex-ante vs ex-post explanation

Ex-ante explains the general properties and features of a system, before use of said system. The AI system is explained ex-ante in a way that provides relevant information to stakeholders other than the developers, on the properties and features of a system, before use of said system.

Ex-post explanation of the properties and features that play a role in the making of a decision. Symbols enhance explainability and thus trustworthiness of AI.

Ex-ante and ex-post explanations serve different functions. Ex-ante explanation strives to establish trust that the system is well designed and serves its purpose. It aims to establish trust with the users and motivate the use of the AI system in the first place. Ex-post explanation, on the other hand, allows for the explanation of specific algorithmic results and the circumstances they were made in. That is, while ex-ante explanation is important for establishing trust in the AI system, it is impossible to achieve system's transparency without access to ex-post explanation as well.

Ideally, an AI system will provide consistency between its ex-ante and ex-post explanations. The properties and features claimed by the ex-ante explanation are evidenced by execution of specific system algorithms exposed through the ex-post explanation.

Subclause 9.3.4 concentrates on ex-post explainability.

9.3.4 Approaches to explainability

It is possible to categorize approaches to explainability based on the stage, scope and granularity of the generated explanations. Explanations can be generated during different stages of developing an AI model:

1) pre-modelling;

2) modelling; and

3) post-modelling.

Pre-modelling serves in the understanding of the data before building the model. A group of methods aims at understanding a given dataset to inform subsequent development of AI models (e.g. facets, embedding projector, dataset standardization, mathematical understanding of a dataset)[72]-[76].

Modelling stage serves in developing AI models that can explain their decisions or that are inherently interpretable[77]-[79].

Post-modelling stage serves to generate explanations about the decisions of a non-interpretable AI model[80]-[88].

The process of explaining an AI model can be described:

- locally, by explaining the model decision-making process for a given input/output pair; or
- globally, by explaining the inner logic of the model about a general concept or class of samples.

In addition, explanations can be generated at different levels of detail. Given a deep neural network, one level of explanation can discuss the role of each layer in the predicted output. A more detailed understanding can be obtained by inspecting the role of each neuron in a given layer[89]-[95].

9.3.5 Modes of ex-post explanation

9.3.5.1 General

Modes of explanation cab be classified as causal, epistemic and justificatory.

These three modes of explanation can be distinct, as an organization can produce a causal explanation, without having produced an epistemic or justificatory explanation. For example, the high saliency of a gender feature in the result of a credit decision algorithm gives a partial causal explanation of how the algorithm produced the decision, but it does not answer the questions of what functional role gender plays in one's credit-viability or by what standards it is valid to justify a credit-decision on that basis.

A full explanation of an AI system can therefore consist of all the following features:

- the chain of causal attributions which track how the algorithm produces a decision;
- the functional roles of the measured features in the modelled phenomenon;
- the ethical and other principles and standards by which an algorithmic output is justified.

The selection of such explanatory features depends on who the explanation is for, what the aims of the explanation are and what level of trust is pursued for the application.

9.3.5.2 Causal explanation: How something functions

For the goal of understanding how an AI system arrives at its results, an explanation consists of a chain of causal attributions explaining the mechanisms by which the input features are processed to produce the given result.

The result of tracking the causal chains in a machine learning process are dependent on the level of abstraction chosen. That is, qualitative properties (e.g. shape of object), computational metaphors (e.g. vector value) and physical matters-of-fact (e.g. charge state in a processor register) can all play a role in the causal history of an AI process and a causal explanation of how a result is produced can proceed at any of these levels[96].

Which level of abstraction is useful, depends on one's explanatory goal. With an interpretable AI system, even the highest level of abstraction of the specific decision factors the system uses and the weight they bear on the final result can be reconciled with humanly meaningful qualitative properties. Furthermore, a causal explanation can support counterfactual interventions[97]. That is, it can yield the understanding of how the produced result would change, were the input features modified.

9.3.5.3 Epistemic explanation: How we know it functions

For the goal of epistemic justification, that is, explanation of why an algorithmically produced result is true, a successful explanation tracks the functional or logical relationships in the modelled phenomenon. That is, it is not a description pertaining to the system itself, but a description pertaining to the features of the world which the system is about.

9.3.5.4 Justificatory explanation: On what grounds it functions

For an automated decision from an AI system, an explanation can provide a justification for the validity of the result. This involves going beyond causal and functional explanation to the societal context to communicate the principles, facts and standards on which the produced decision is grounded. Such an explanation communicates why the resulting decision is fair, valid and justified in light of the current state of affairs.

While this kind of justificatory explanation can refer to the AI systems' properties, like the algorithms, data used and decision features, it is incomplete without reference to institutional and social facts about the implementation of the system. This includes regulations, standards and organizational processes pertinent to the use case.

A successful justificatory explanation functions as an argument in support of the systematically produced result. Thus, a successful explanation is open to scrutiny and contestation and likewise, the result of the system is re-assessable in light of possible counter-arguments seeking reversal or redress.

9.3.6 Levels of explainability

The appropriate level of explainability of an AI system can be selected for the context of the use-case in which the system is applied. As such, clearly defined requirements for different levels of explanation can assist in the selection of the type of AI for an application based on its level explanatory power. As an example, non-interpretable systems that do not produce a meaningful causal explanation of their functioning, are not suitable to be used in products or services which expect a high level of explainability. The level of applicable explainability is something that is evaluated on a case-by-case basis.

In selecting the necessary level of explainability for an AI system, application considerations can include the following:

- the AI system uses sensitive personal data of individuals as its input;
- AI system results are used in a way that has significant impact on the welfare of individuals;
- the consequences of failures in AI-driven decision-making are significant;
- the application can result in the autonomy of the user or of third parties being restricted;
- the system has a non-trivial effect on bystanders and the wider society in which it is deployed, e.g. only showing certain job advertisements to men leads to increased gender inequality.

Levels of explanation can vary based on the specific needs of different stakeholder groups, the data aspects on which the AI system results are based, the need to obtain human intervention over decisions based on AI results, and the need for stakeholders to express their own views on, and challenge, such decisions.

9.3.7 Evaluation of the explanations

It is important also to consider the measurement of the quality of explanations. This includes considering the following aspects:

- continuity, for which the associated explanation for the predictions of nearby points would be nearly equivalent;
- consistency, where if we change the model such that the contribution of a certain feature on the predicted output is increased, the importance-score of that feature, estimated by the explainability method, would not be decreased;
- selectivity, where for importance-based explanations, it is desirable that the contribution score be distributed among the features that have the strongest impact on the generated prediction. That is, the removal of a feature (or a set of features) with highest relevance score, would result in a sharp change in the model output. This ensures that the correct features are distinguished as relevant in the generated explanation.

It is also important to consider the trade-off between the accuracy and understandability of an explanation[98]-[105].

9.4 Controllability

9.4.1 General

It is possible to achieve controllability by providing reliable mechanisms for an operator to take over control from the AI system. To achieve controllability, the questions to address first include who is offered what control over whose AI systems where multiple stakeholders are involved, e.g. the service provider or product vendors, the provider of the constituent AI, the user or an actor with regulatory authority.

We describe below the need to integrate the control points in the AI system lifecycle, as a step towards reliable decision-making.

9.4.2 Human-in-the-loop control points

Regarding the role of humans within the lifecycle of AI systems, two roles are particularly relevant as human-in-the-loop control points:

- decision makers having agency and autonomy in the final decision-making process to factor into account the outcomes of AI systems where they are used to augment human decision-making;
- domain experts given the opportunity to provide feedback to not only re-assess the level of trust of the system but also to improve the operation of the system. In this context, results checked/ contextualized by domain experts is important for AI systems because domain experts can be able to spot spurious correlations or rationalise why a system works in a certain way with data that is not available to the AI system.

9.5 Strategies for reducing bias

Many strategies exist to address bias:

- consideration of legal and other requirements relating to bias can be explicitly identified when defining system requirements, including setting appropriate thresholds;
- analysis of the provenance and completeness data sources can reveal risks and the processes used to collect or annotate data can be reviewed;
- technical techniques can be used as part of model training processes, to detected and mitigate bias;
- specific testing and evaluation techniques can be used to detect bias;
- trials or regular operational reviews can be used to detect bias related issues in the actual context of use.

Each of these approaches has advantages and disadvantages. Investigating the risk related to the biases and documenting the mitigation techniques helps to build trustworthiness on AI.

9.6 Privacy

Syntactic methods (such as k-anonymity) or semantic methods (such as differential privacy) are used to de-identify personal data[52]. Even when the data is de-identified, when personal data is available from multiple sources, it is possible for AI to re-identify the data using the inferences based on the data from the other sources. For example, research[53],[54] shows that *k*-anonymity can be insufficient.

Regardless of the initial de-identification approach, it is possible to manage the residual risk of re-identification with data-usage agreements between the parties receiving the data.

9.7 Reliability, resilience and robustness

Publication[30] points to a system "ability" as one of the crucial components for achieving trustworthiness. Ability can be described as a system characteristic to perform a specific task and can be assessed in terms of several attributes including reliability, resilience and robustness.

Reliability is the ability of a system or an entity within that system to perform its required functions under stated conditions for a specific period of time[106]. In other words, a reliable AI system produces the same outputs for the same inputs consistently.

With AI systems, as with other software systems types, hardware faults can affect the correct execution of the algorithm. Fault tolerance is the system's ability to continue to operate when disruption, faults and failures occur within the system, potentially with degraded capabilities. The aspect of the overall system that depends on a system or equipment operating correctly in response to its inputs is generally known as functional safety[107].

Resilience is the ability of the system to recover operational condition quickly following an incident. Resilience relates to reliability, but the expected service levels and expectations are different. With resilience expectations possibly lower as defined by stakeholders, as well as offering recovery (see Reference [42], subclause 11.5).

For AI systems, robustness is often used to describe the ultimate ability of a system to maintain its level of performance under any circumstances including external interference or harsh environmental conditions. Robustness encompasses resilience, reliability and potentially more attributes, as related to proper operation of a system as intended by its developers. Obviously, the proper operation of a system is directly related or leads to the safety of its stakeholders in a given environment/context. For example, a robust ML-based AI system would have the ability to generalize on unknown inputs, e.g. an absence of overfitting. To achieve that, it is essential to train the model or models using large training datasets including noisy training data.

9.8 Mitigating system hardware faults

Robust and fault tolerant systems are achieved by different methods that are related to the architecture and detailed design of the hardware but also the whole development process. Therefore, every phase of a product's life cycle, especially the design and specification phase, is in scope.

One of these methods is to exploit redundancy to mask or otherwise work around failures, thus maintaining the desired level of functionality. Hardware faults can be dealt with by using hardware (e.g. *n*-plication at a course or fine-grain), information (e.g. check bits) or time (e.g. re-computation at different, usually random times) redundancy, whereas software faults are protected against by software redundancy (e.g. software diversity or other forms of moving target mitigation).

The former type of faults is mitigated by incorporating extra hardware into the design to either detect or override the effect of a failed component. Hardware redundancy can be static or dynamic. It can thus range from a simple duplication to complicated structures that switch in spare units when active ones become faulty.

To avoid the malicious effects of common cause failures, such as influences from environmental conditions or weaknesses of particular sensor technologies, more measures (e.g. the use of diversity) is necessary. In addition, various diagnostic measures can help to detect errors at runtime and to take countermeasures or to transfer the system to a safe state.

A comprehensive description of methods and processes for implementing fail-safe hardware and a description of certifiable functional safety levels can be found in IEC 61508[107].

9.9 Functional safety

To ensure functional safety of a system, specific functionality can be introduced that performs safety related aspects. Such functionality can be an integral part of the control functionality of a system or a dedicated system that interfaces with the systems under consideration. For AI systems, safety-related functionality can, for example, monitor the decisions taken by the AI in order to ensure that they are in a tolerable range or bring the system into a defined state in case they detect problematic behaviour.

IEC 61508[107] sets out a generic approach for all safety lifecycle activities for systems comprised of electrical and/or electronic and/or programmable electronic elements that are used to perform safety functions. It is the base for product and application sector international standards, dealing with such safety-related systems. ISO 26262[108], IEC 62279[109] and IEC 61511[110] are examples for sector-specific adaptation for the automotive, railway and process industry. IEC 61508[107] is applicable to all electrical and/or electronic and/or programmable electronic (E/E/PE) safety-related systems irrespective of the application. It is mainly concerned with the such systems whose failure can have an impact on the safety of persons and/or the environment. However, it is recognized that the consequences of failure can also have serious economic implications and, in such cases, this document can be used to specify any E/E/PE system used for the protection of equipment or product.

9.10 Testing and evaluation

9.10.1 General

Different approaches to testing and evaluation of AI systems exist. While their applicability and effectiveness can differ case by case, a combination of multiple approaches would typically be needed to achieve acceptable levels of trustworthiness.

9.10.2 Software validation and verification methods

9.10.2.1 General

To achieve trustworthiness, traditional (non-AI) software systems rely on the following two axes:

- an architecture that allows redundancy or monitoring of critical functions; and
- the validation and verification of the code, which consists of demonstrating through an engineering approach that the executable code meets a need and is fully tested. The demonstration currently relies on the fact that the behaviour of the system is known and deterministic.

According to Reference [111], validation is "confirmation, through the provision of objective evidence, that the requirements for a specific intended use or application have been fulfilled. Note 1: the right system was built." Verification is the "confirmation, through the provision of objective evidence, that specified requirements have been fulfilled. Note 1: the system was built right"[24].

Software systems are also subjected to formal software validation, verification and testing methods, such as defined in Reference [111] The primary goals of software tests are stated in Reference [111]:

> "Provide information about the quality of the test item and any residual risk in relation to how much the test item has been tested; to find defects in the test item prior to its release for use; and to mitigate the risks to the stakeholders of poor product quality."

By design, AI systems are often less deterministic than traditional software systems and rarely exhaustively explainable. The software of an AI system comprises both AI and non-AI components.

While all components of an AI system need to follow accepted software and hardware practices (including unit and functional tests) to operate correctly, its AI components would use a modified version of these practices as discussed below.

In the case of AI systems, it is necessary for functional tests to be able to handle uncertainty when applicable. It is a challenge to specify and test the requirements of non-deterministic software components using existing standards and practices. This is known as an "oracle problem" and can be described as difficulties in establishing whether an individual test has met its success criteria. The prevalence of this issue in AI systems suggests that new standardization efforts can be initiated to encourage new verification and validation techniques.

9.10.2.2 Formal methods

It is possible to use formal methods to test and evaluate artificial neural networks for the purpose of software validation and verification. To do so, several metrics can be used, such as:

- uncertainty, which correlates to variation of response of the network in order to check if its generalization does not introduce unstable behaviour;
- maximum stable space, which correlates to the ability of the AI system to prove that the classification done will be stable around the training set.

9.10.2.3 Empirical testing

Various techniques exist for empirical testing of non-deterministic solutions for the purpose of software validation and verification, including:

- metamorphic testing – a technique that establishes relationships between inputs and outputs of the system and relies on running multiple iterations of testing and comparing the results. This is typically used on systems which do have an oracle problem[112];
- expert panels – where AI systems are built to replace the judgement of experts, panels are established to review the test results. This approach introduces additional challenges, e.g. when experts disagree[113];
- benchmarking – a technique that measures the performance of a system on carefully designed data sets that are publicly available and/or used for testing different systems competitively[114]. In pattern recognition and similar applications of AI methods, benchmarking has been an established practice for establishing trust in a certain method[115].

9.10.2.4 Intelligence comparison

When no automated evaluation method is available, comparison of the applied intellectual abilities of an AI system and human can provide confidence in AI system quality by confirming the functionality of the AI system. This approach relies on a comparison of certain indicators with a given criteria threshold. Various methodologies can then be applied (for example the concordance coefficient, Pearson's test) over various environments (for example "sandbox" or physical containment).

9.10.2.5 Testing in simulated environment

In some cases, when the task to be performed by an AI system is characterized by physical action on the environment (e.g. for AI embedded in a robot), performance evaluation and compliance analysis with risk-related requirements needs to be performed in a real or representative environment. To determine the operating perimeter of the embedded AI, which is needed to promote the acceptability of such intelligent mechatronic systems, it is possible to carry out tests in controlled environments. Physical tests in climatic chambers, vibration, shock and constant acceleration tests can also be performed to evaluate the performance of systems under extreme conditions and to accurately determine operating boundary conditions. For the evaluation of AI systems in open and changing environments, having an almost infinite number of configurations that are possible to encounter, the development of virtual test environments allowing validation by simulation can also be useful.

9.10.2.6 Field trials

Due to the difference between testing environments and actual operating conditions, field trials are often a very effective way to improve the quality of the deployed system by testing its performance, efficiency or durability.

Some prominent examples and areas are:

- facial recognition trials[116];
- tests of decision support systems for agricultural applications[117];
- practice for testing driverless cars[118],[119];
- tests of speech and voice recognition systems[120],[121];
- health robotics[122];
- measuring the cognitive workload of chatbots (vocal assistants, etc.)[123]; and
- testing intelligent tutoring systems[124].

Field trials for AI systems vary greatly with respect to methodology, number of users or use cases involved, status of the responsible organization/persons and documentation of the results. Whether field trials can be applied as a measure to improve the quality of AI systems depends on the risks associated with the application of such systems. In many applications, A/B testing is used as a technique to deliver different versions of systems to different users, in order to compare the performance of the system.

Beyond strict rationality of the AI software built, acceptability to humans needs to be taken into account and field trials can help to achieve that. In addition, the failure of an AI system on a functional test can be unnecessary or can be impossible to resolve. AI systems showing variable results can be regarded as useful for their intended purpose and the ability of the system to achieve the planned and desired result of an AI system cannot always be measurable by conventional approaches to software testing.

Another fundamental difference between many AI systems and conventional systems is that the latter are designed to be developed, produced and quality controlled to strictly meet certain specifications. Traditional software is designed to be reproducible in its behaviour, whereas AI systems instead seek generalizability. This leads to challenges in empirical testing and field trials can be more effective at assessing quality.

How to deal with the uncertainty of a product's results and the risks of its deployment are subjects of many regulations in the medical domain. Medical AI systems can be required to comply with ISO 14155[125]. They can have to undergo "clinical investigations", a procedure that resembles "clinical trials"[126],[127]. This is true for other domains as well, such as nuclear systems and flight control systems.

9.10.2.7 Comparison to human intelligence

In cases when an AI system is designed to automate human activity associated with data processing and decision-making, one of the ways to validate an AI system is by comparison to human intelligence capabilities. Such an approach can allow different stakeholders, primary AI systems' users, outside parties and regulators in the AI implementation area (including regulatory authorities) to trust AI systems with carrying out some application tasks associated with data processing and decision-making that previously were carried out primarily by humans.

Examples of when comparison to human capabilities would be helpful are activities that are traditionally licensed, such as operating a motor vehicle or healthcare. Allowing autonomous vehicles to drive on city streets or an autonomous system to make any treatment, would happen only if there is an evidence that the AI system conducting these activities performs not worse than a human. Such an approach allows for the following:

— AI systems' users and stakeholders can expect that the AI system's quality, while performing an information processing task, are not worse than the quality of the solution of the same problem performed by a human-operator;

— third parties can expect that the operation of an AI system will not cause damage to people and material goods.

It would be able to conclude that an AI system is not worse than human capabilities, while performing some application tasks associated with data processing and process safety, if AI statistical metrics are not worse than a defined threshold value.

To obtain such threshold values and confidence in process safety of AI systems, it is important to use representative data samples that reflect the nature of the applied information processing task, to which the AI system or natural human intellect are applied.

9.10.3 Robustness considerations

A definition of robustness is "the ability of a system to maintain its level of performance under any condition". In order to understand what robustness is in a more general sense, it is important to note that what AI systems are commonly used for is, for example, to infer knowledge (symbolic approach) or to generalize from data (sub-symbolic approach).

The main principle is that an AI system is expected to be able to work on data that is not known in advance and under contexts that can vary considerably. An AI system is expected to deal with working conditions that can vary a lot and its robustness corresponds to its ability to continue to operate according to its design. Depending on the type of AI system, different metrics are needed to assess the robustness of the system.

When an AI system is used to perform interpolation, its robustness is viewed as its "ability to have acceptable metrics of amplitude of response on any valid input". This means that it is expected that the AI system does not exhibit erratic behaviour in its interpolation.

When an AI system is used to perform classification, its robustness is viewed as its "ability to assign consistent classification on both known inputs and inputs within a certain range". This means that it is expected that the AI system is able to properly conduct classification on both known and unknown inputs as long as they (the unknowns) are not too different from the known inputs.

When an AI system is used to perform a solving task, its robustness is viewed as its "ability to have a still effective solution after an acceptable change of the initial problem". This means that it is expected that the AI system is able to produce acceptable solutions to different problems as long as they are not too different from the original problem.

When an AI system is used to perform scoring, its robustness is viewed as the "ability to assign consistent confidence measures of ranking on both known inputs and inputs within an acceptable range". This means that, in the case of unknown inputs and outputs, it is expected that the AI system assigns a score that is not radically different from a score being assigned to known inputs and unknown inputs as long as they are not too different from the known inputs.

9.10.4 Privacy-related considerations

In addressing privacy threats in AI, privacy metrics help to evaluate levels of privacy and amount of protection provided by the system. Defining and applying privacy metrics aims at addressing this challenge. There are various privacy-preserving machine learning or privacy-enhancing techniques in AI to protect sensitive data in different domains. The purpose of defining privacy metrics is to quantify the data privacy level that results in improving the privacy model within a specific AI model[128],[129]. Technically, a privacy metric considers different properties of data and yields a value that represents the privacy level in the system. The advantage of privacy metrics is the ability to compare different privacy-preserving techniques, evaluate different methods within a specific domain and to minimize the privacy exposure. Privacy metrics are useful when sensitive data is threatened by an adversary. Privacy metrics differ considering the data source, aspects of privacy they evaluate and threats by an adversary.

9.10.5 System predictability considerations

Some of the test and validation approaches described above are essential for assessing the predictability of an AI system. Predictability can be measured through subjective explicit feedback from questionnaire-based experiments where participants are asked to infer the goals and predict the future actions of a robot, for example[130]. Other metrics can be used, such as the reaction time for a user to determine the intention of an AI system and react accordingly, assuming that shorter reaction times indicate higher predictability. The gaze behaviour also gives an indirect indication of the robot predictability, assuming that the more often and the longer the robot is watched by a participant, the less it is predictable[131]. Testing the AI system on a large number of combinations of environmental conditions would allow for a more complete characterization of its behaviour. Based on this characterization, users know what to expect from the AI system, which facilitates predictability.

9.11 Use and applicability

9.11.1 Compliance

The need for compliance is, in part, attributed to the various standards and regulations that are already in operation in various industries. AI systems need to take into account existing regulations and compliance standards and not be judged in isolation of the use case.

9.11.2 Managing expectations

Expectation management is necessary to avoid breakdown of trust because the system was not able to perform to unrealistic expectations. This requires clarity about the realistic capability of the AI system, including the range of inputs for which a reliable output is expected and the certainty of outputs (especially for systems that rely on statistical inference).

9.11.3 Product labelling

Product and system labelling (including the links to up-to-date information) can be essential for the safety of both the users and the AI system providers:

- that the end user is interacting with an AI agent and declaring the intent/purpose of AI system;
- risks and limitations of the model;
- re-training frequency as necessary;
- date when last performance assessment was done;
- source and date of training data[132].

9.11.4 Cognitive science research

Based on the discussion in this subclause, application specific guidance and considerations are essential to achieve the right level of trustworthiness and the correct use of the systems. Nevertheless, there are several recognized patterns of interaction between the quality of a system and its potential for misuse or disuse. For instance, heavily reliable systems are known to inspire over-reliance leading to misuse (such as in aviation), whereas unreliable systems (such as EHR) are known to inspire mistrust and hence disuse[133]. The same pattern holds for other metrics such as robustness, resilience and accuracy - better systems inspire trust, which can cause systemic failure in situations the automated system is not built to handle.

As a result, keeping human factors in view, it is best to take a nuanced view of optimizing these metrics in human-facing AI systems. This view would draw on human-computer interaction (HCI) and cognitive science research to yield useful results and potentially scientifically justified standards.

10 Conclusions

The realization of the potential benefits of AI systems can be impeded by a lack of trust from customers, users and society in general in the reliability, effectiveness, fairness and even the intent of AI applications. Business, governmental, societal and ethical concerns can, if not addressed systematically, erode trust in AI. Such concerns can be driven by vulnerabilities exhibited by ML-based AI systems, e.g. bias, unpredictability and opaqueness. Since many ML applications are driven by big data, data privacy and other data management issues, e.g. data provenance and quality, can become major concerns in building and using AI systems. To improve the trustworthiness of AI, the ML and data vulnerabilities need to be explicitly considered and addressed in policies, processes and on a per-use-case basis.

The potential effect of AI vulnerabilities on stakeholders needs to be examined to decide whether the use of AI would be appropriate in a specific case. An organization developing or using AI can apply a risk-based approach to identify possible impacts to the organization, to its partners, to the intended users