DRAFT INTERNATIONAL STANDARD
ISO/IEC DIS 24029-2

ISO/IEC JTC **1**/SC **42** Secretariat: **ANSI**

Voting begins on:
2022-07-12

Voting terminates on:
2022-10-04

Artificial intelligence (AI) — Assessment of the robustness of neural networks —

Part 2:
Methodology for the use of formal methods

ICS: 35.020

This document is circulated as received from the committee secretariat.

Reference number
ISO/IEC DIS 24029-2:2022(E)

Contents

Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

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 ISO documents 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 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).

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 Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 42, *Artificial intelligence*.

A list of all parts in the ISO/IEC 24029 series can be found on the ISO website.

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

Neural networks are widely used to perform complex tasks in various contexts, such as image or natural language processing, and predictive maintenance. AI system quality models comprise certain characteristics, including robustness. For example, ISO/IEC 25059:—[1] [1], which extends the SQuaRE series [2] to AI systems, considers in its quality model that robustness is a sub-characteristic of reliability. Demonstrating the ability of a system to maintain its level of performance under varying conditions can be done using statistical analysis, but proving it requires some form of formal analysis. In that regard formal methods can be complementary to other methods in order to increase the trust in the robustness of the neural network.

Formal methods are mathematical techniques for rigorous specification and verification of software and hardware systems with the goal to prove their correctness. Formal methods can be used to formally reason about neural networks and prove whether they satisfy relevant robustness properties. For example, consider a neural network classifier that takes as input an image and outputs a label from a fixed set of classes (such as car or airplane). Such a classifier can be formalized as a mathematical function that takes the pixel intensities of an image as input, computes the probabilities for each possible class from the fixed set, and returns a label corresponding to the highest probability. This formal model can then be used to mathematically reason about the neural network when the input image is modified. For example, suppose we are given a concrete image for which the neural network outputs the label “car”. We can ask the question: “can the network output a different label if we arbitrarily modify the value of an arbitrary pixel in the image?” This question can be formulated as a formal mathematical statement that is either true or false for a given neural network and image.

A classical approach to using formal methods consists of three main steps that are described in this document. First, the system to be analyzed is formally defined in a model that precisely captures all possible behaviours of the system. Then, a requirement is mathematically defined. Finally, a formal method, such as solver, abstract interpretation or model checking, is used to assess whether the system meets the given requirement, yielding either a proof, a counterexample or an inconclusive result.

This document provides the methodology including recommendations and requirements on the use of formal methods to assess the robustness of neural networks during their life cycle. The document covers several available formal method techniques. At each step of the life cycle, the document presents criteria that are applicable to assess the robustness of neural network and to establish how neural networks are verified by formal methods. Formal methods can have issues in terms of scalability, however they are still applicable to all types of neural networks performing various tasks on several data types. While formal methods have been used on traditional software systems for a while, the use of formal methods on neural networks is fairly recent and is still an active field of investigation.

This document is aimed at helping artificial intelligence engineers and quality engineers who use neural networks and who have to assess their robustness throughout their life cycle. The reader can also refer to ISO/IEC TR 24029-1:2021 [3] to have a more detailed overview of the techniques available to assess the robustness of neural networks, beyond the formal methods used by this document.

[1] Under preparation. Stage at the time of publication: ISO/IEC CD 25059:2021.

Information technology — Artificial Intelligence (AI) — Assessment of the robustness of neural networks — Part 2: Methodology for the use of formal methods

1 Scope

This document provides methodology for the use of formal methods to assess robustness properties of neural networks. The document focuses on how to select, apply and manage formal methods to prove robustness properties.

2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO/IEC 22989:—[2], *Information Technology — Artificial intelligence — Artificial intelligence concepts and terminology*

ISO/IEC 23053:—[3], *Framework for Artificial Intelligence (AI) Systems Using Machine Learning (ML)*

3 Terms and definitions

For the purposes of this document, the terms and definitions given in ISO/IEC 22989:—, ISO/IEC 23053:— and the following 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
architecture
fundamental concepts or properties of a system in its environment embodied in its elements, relationships, and in the principles of its design and evolution

[SOURCE: ISO/IEC/IEEE 42010:2011, 3.2]

3.2
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]

3.3
bounded domain
set containing a finite number of objects

[2] Under preparation. Stage at the time of publication: ISO/IEC FDIS 22989:2022.
[3] Under preparation. Stage at the time of publication: ISO/IEC FDIS 23053:2022.

EXAMPLE 1: The domain of all valid 8-bit RGB images with n-pixels is bounded by its size which is at most $256^{3.n}$.

EXAMPLE 2: The number of all valid English sentences is infinite; therefore this domain is unbounded.

Note 1 to entry: The number of objects in an unbounded domain can be infinite.

3.4
bounded object
object represented by a finite number of attributes

Note 1 to entry: Contrary to a bounded object, an unbounded object is represented with an infinite number of attributes.

3.5
criteria
criterion
rules on which a judgment or decision can be based, or by which a product, service, result, or process can be evaluated

[SOURCE: ISO/IEC/IEEE 15289:2019(en), 3.1.6, added criterion as admitted term]

3.6
domain
set of possible inputs to a neural network characterized by attributes of the environment

EXAMPLE 1: A neural network performing a natural language processing task is manipulating texts composed of words. Even though the number of possible different texts is unbounded, the maximum length of each sentence is always bounded. An attribute describing this domain can therefore be the maximum length allowed for each sentence.

EXAMPLE 2: A face capture requirements can include, inter alia, that the size of faces is at least 40x40 pixels. That half-profile faces are detectable at a lower level of accuracy, provided most of the facial features are still visible. Similarly, partial occlusions are handled to some extent. Detection typically requires that more than 70% of the face is visible. Views where the camera is the same height as the face perform best and performance degrades as the view moves above 30 degrees or below 20 degrees from straight on.

Note 1 to entry: An attribute is used to describe a bounded object even though the domain can be unbounded.

3.7
model
<model checking> formal expression of a theory

3.8
stability
extent to which the output of a neural network remains the same when its inputs are changed

Note 1 to entry: Stability is not responding to change when input change is noise.

3.9
sensitivity
extent to which the output of a neural network varies when its inputs are changed

Note 1 to entry: Sensitivity is responding to change when input change is informative.

3.10
time series
sequence of values sampled at successive points in time

[SOURCE: ISO/IEC 19794-7:2007(en), 4.2]

4 Abbreviated terms

AI	artificial intelligence
BNN	binarized neural networks
MILP	mixed-integer linear programming
MRI	magnetic resonance imaging
PLNN	piecewise linear neural networks
ReLU	rectified linear unit
RNN	recurrent neural networks
SAR	synthetic aperture radar
SMC	satisfiability modulo convex
SMT	satisfiability modulo theories

5 Robustness assessment

5.1 General

In the context of neural networks, robustness specifications typically represent different conditions that can naturally or adversarially change in the domain (see Clause 5.2) in which the neural network is deployed.

EXAMPLE 1: Consider a neural network that processes medical images, where inputs fed to the neural network are collected with a medical device that scans patients. Taking multiple images of the same patient naturally does not produce identical images. This is because the orientation of the patient can slightly change, the lighting in the room can change, an object can be reflected or random noise can be added by image post-processing steps.

EXAMPLE 2: Consider a neural network that processes the outputs of sensors and onboard cameras of an self-driving vehicle. Due to the dynamic nature of the outside world, such as weather conditions, pollution and lighting conditions, the input to the neural network is expected to have wide variations of various attributes.

Importantly, these variations introduced by the environment are typically not expected to change the neural network's output. The robustness of the neural network can then be verified against changes to such environmental conditions by verifying its robustness against relevant proxy specifications within the domain of use of the neural network.

Robustness properties can be local or global [10]. It is more common to verify local robustness properties than global robustness properties, as the former are easier to specify. Local robustness properties are specified with respect to a sample input from the test dataset. For example, given an image correctly classified as a car, the local robustness property can specify that all images generated by rotating the original image within 5 degrees are also classified as a car. A drawback of verifying local robustness properties is that the guarantees are local to the provided test sample and do not extend to other samples in the dataset. In contrast, global robustness properties define guarantees that hold deterministically over all possible inputs [11]. For domains where input features have semantic meaning, for example, air traffic collision avoidance systems, the global properties can be specified by defining valid input values for the input features expected in a real-world deployment. Defining meaningful input values is more challenging in settings where the individual features have no semantic meaning.

5.2 Notion of domain

Most AI systems, including artificial neural networks, are intended to operate in a particular environment where their performance characteristics can be defined and evaluated (typical metrics of evaluation can be found in Table 1 of ISO/IEC TR 24029-1:2021 [3]). Robustness, being one of the key performance characteristics, is inseparable from the domain where a neural network is operating. The existence of a bounded domain is implicit in many neural network applications (e.g. image classification expects images of certain quality and in a certain format).

The agent paradigm shown in Figure 1 drawn from ISO/IEC 22989:— postulates that an agent senses its environment and acts on this environment towards achieving certain goals. The distinct concepts AI Agent and environment are emphasized in this paradigm. The notion of domain captures the limitations of current technology where a neural network, being a particular type of AI agent, is technically capable of achieving its goal only if it is operating on appropriate inputs. An example is a neural network operating in an environment where all relevant features and qualities for its goal have been taken into consideration in the design, training and deployment.

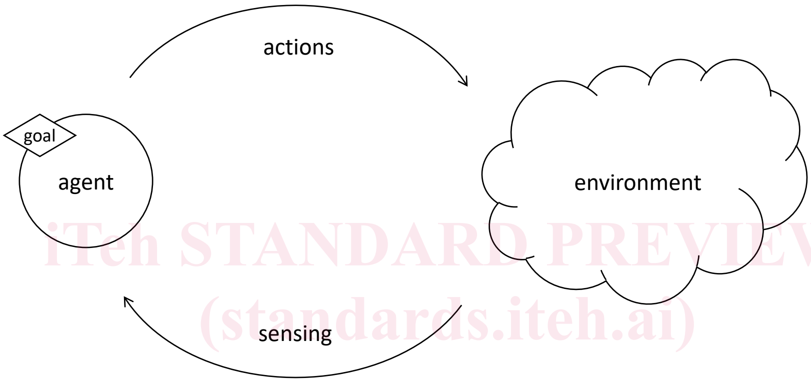

Figure 1 — The agent paradigm

The definition rests on the following pillars:

— to be of practical use, a domain needs to be determined by a set of attributes which are clearly defined;

— the specification of domain should be sufficient for the AI system to conduct one or more given tasks as intended;

— data used for training should be representative of data expected to be used for inference.

Establishing a domain involves specifying all data attributes essential for the neural network to be capable of achieving its goal.

Several popular domains of application of neural networks cover applications in vision, speech processing and robotics. To describe these domains, and more importantly their variability, the attributes used are generally numerical in essence. For example, the shape of an object in an image, the intensity of some pixels or the amplitude of an audio signal.

However, there are other domains that can be expressed through non-numerical attributes. natural language processing (NLP), BigCode (the use of automatically learning from existing code) and graphs are examples of such domains. In these cases, the attributes can be non-numerical, for example, the words in a sentence or the edges in a graph.

The attributes allow the user to morph one instance in the domain to another instance and should be bounded in the robustness specification.

5.3 Stability

5.3.1 Stability property

A stability property expresses the extent to which a neural network output remains the same when its inputs vary over a specific domain. Checking the stability over a domain where the behaviour is supposed to hold allows for checking whether or not the performance will hold too. A stability property can be expressed either in a closed-end form (e.g. “is the variation under this threshold?”) or an open-ended form (e.g. “what is the largest stable domain?”).

In order to prove that a neural network remains performant in the presence of noisy inputs, a stability property shall be expressed. A stability property should only be used on domains of uses which, in terms of expected behaviour, present some regularity properties. It should not be used on a chaotic system, for example, as it will not be relevant. When the regularity of the domain is not easy to affirm (e.g. chaotic system), it can still be useful to use the stability property to compare neural networks.

5.3.2 Stability criterion

A stability criterion establishes whether a stability property holds within a specific domain, not a specific set of examples nor a subset of the domain such as training or validation datasets. A stability criterion can be checked using formal methods described in 6.2.

A stability criterion shall define at least the domain value space and output value space on which it has been measured and the stability property expected.

A stability criterion may be used as one of the criteria to compare models.

In order to be a fair comparison the neural networks need to have performed the same tasks, the criterion needs to have been used on the same domain and the criterion needs to have the same objective to be proven.

For example, for a neural network doing classification, a stability criterion assesses whether or not a particular decision holds for every input in the domain. For a neural network doing regression, a stability criterion assesses whether or not the regression remains stable on the domain.

To be applicable, a stability criterion relies on pre-existing information of the expected output of the neural network. This information can be known by the user or can be determined by another means (using simulation or solvers systems). It is well-suited to assess the robustness over a domain where the expected answer is known to be similar. For this reason a stability criterion is especially recommended for any decision-making process handled by a neural network (e.g. classification, identification).

5.4 Sensitivity

5.4.1 Sensitivity property

A sensitivity property on a neural network expresses the extent to which the output of a neural network varies when its inputs are changed. In order to assess the robustness on a domain it is sometimes necessary to check the variability of a system. A sensitivity analysis can be carried out to determine how much the system varies and the inputs which can influence that variance. This analysis is then compared to a pre-existing understanding of the expected performance of the system.

Sensitivity analysis shall be used over a domain to prove that a neural network stays bounded. As is the case for the stability property, sensitivity analysis can be more suited for domains of use which present some regularity properties.