



International
Standard

ISO/IEC 15938-17

**Information technology —
Multimedia content description
interface —**

**Part 17:
Compression of neural networks for
multimedia content description and
analysis**

*Technologies de l'information — Interface de description du
contenu multimédia —*

*Partie 17: Compression des réseaux neuronaux pour la
description et l'analyse du contenu multimédia*

**Second edition
2024-01**



COPYRIGHT PROTECTED DOCUMENT

© ISO/IEC 2024

All rights reserved. Unless otherwise specified, or required in the context of its implementation, no part of this publication may be reproduced or utilized otherwise in any form or by any means, electronic or mechanical, including photocopying, or posting on the internet or an intranet, without prior written permission. Permission can be requested from either ISO at the address below or ISO's member body in the country of the requester.

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

Published in Switzerland

Contents

Page

Foreword	v
Introduction	vi
1 Scope	1
2 Normative references	1
3 Terms and definitions	1
4 Abbreviated terms, conventions and symbols	3
4.1 General	3
4.2 Abbreviated terms	3
4.3 List of symbols	3
4.4 Number formats and computation conventions	6
4.5 Arithmetic operators	6
4.6 Logical operators	7
4.7 Relational operators	7
4.8 Bit-wise operators	7
4.9 Assignment operators	8
4.10 Range notation	8
4.11 Mathematical functions	8
4.12 Array functions	9
4.13 Order of operation precedence	11
4.14 Variables, syntax elements and tables	11
5 Overview	13
5.1 General	13
5.2 Compression tools	13
5.3 Creating encoding pipelines	14
6 Syntax and semantics	15
6.1 Specification of syntax and semantics	15
6.1.1 Method of specifying syntax in tabular form	15
6.1.2 Bit ordering	16
6.1.3 Specification of syntax functions and data types	16
6.1.4 Semantics	17
6.2 General bitstream syntax elements	18
6.2.1 NNR unit	18
6.2.2 Aggregate NNR unit	18
6.2.3 Composition of NNR bitstream	19
6.3 NNR bitstream syntax	20
6.3.1 NNR unit syntax	20
6.3.2 NNR unit size syntax	20
6.3.3 NNR unit header syntax	20
6.3.4 NNR unit payload syntax	25
6.3.5 Byte alignment syntax	31
6.4 Semantics	31
6.4.1 General	31
6.4.2 NNR unit size semantics	31
6.4.3 NNR unit header semantics	31
6.4.4 NNR unit payload semantics	39
7 Decoding process	45
7.1 General	45
7.2 NNR decompressed data formats	46
7.3 Decoding methods	47
7.3.1 General	47
7.3.2 Decoding method for NNR compressed payloads of type NNR_PT_INT	47
7.3.3 Decoding method for NNR compressed payloads of type NNR_PT_FLOAT	48

7.3.4	Decoding method for NNR compressed payloads of type NNR_PT_RAW_FLOAT	48
7.3.5	Decoding method for NNR compressed payloads of type NNR_PT_BLOCK	49
7.3.6	Decoding process for an integer weight tensor	50
8	Parameter reduction	51
8.1	General	51
8.2	Methods	51
8.2.1	Batchnorm folding	51
8.3	Syntax and semantics	52
8.3.1	Sparsification using compressibility loss	52
8.3.2	Sparsification using micro-structured pruning	52
8.3.3	Combined pruning and sparsification	52
8.3.4	Unstructured statistics-adaptive sparsification	53
8.3.5	Structured sparsification (global and local approach)	53
8.3.6	Weight unification	53
8.3.7	Low rank/low displacement rank for convolutional and fully connected layers	54
8.3.8	Batchnorm folding	54
8.3.9	Local scaling adaptation (LSA)	54
9	Parameter quantization	55
9.1	General	55
9.2	Methods	55
9.2.1	Uniform quantization method	55
9.2.2	Codebook-based method	55
9.2.3	Dependent scalar quantization method	55
9.2.4	Predictive residual encoding (PRE)	55
9.3	Syntax and semantics	55
9.3.1	Uniform quantization method	55
9.3.2	Codebook-based method	56
9.3.3	Dependent scalar quantization method	56
10	Entropy coding	56
10.1	Methods	56
10.1.1	DeepCABAC	56
10.2	Syntax and semantics	58
10.2.1	DeepCABAC syntax	58
10.3	Entropy decoding process	64
10.3.1	General	64
10.3.2	Initialization process	64
10.3.3	Binarization process	65
10.3.4	Decoding process flow	66
	Annex A (normative) Implementation for NNEF	73
	Annex B (informative) Implementation for ONNX®	75
	Annex C (informative) Implementation for PyTorch®	77
	Annex D (informative) Implementation for TensorFlow®	79
	Annex E (informative) Recommendation for carriage of NNR bitstreams in other containers	81
	Annex F (informative) Recommendation for naming method regarding performance metric type	83
	Annex G (informative) Encoding side information for selected compression tools	84
	Bibliography	95

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 or www.iec.ch/members_experts/refdocs).

ISO and IEC draw attention to the possibility that the implementation of this document may involve the use of (a) patent(s). ISO and IEC take no position concerning the evidence, validity or applicability of any claimed patent rights in respect thereof. As of the date of publication of this document, ISO and IEC had received notice of (a) patent(s) which may be required to implement this document. However, implementers are cautioned that this may not represent the latest information, which may be obtained from the patent database available at www.iso.org/patents and <https://patents.iec.ch>. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

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. In the IEC, see www.iec.ch/understanding-standards.

This document was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 29, *Coding of audio, picture, multimedia and hypermedia information*.

This second edition cancels and replaces the first edition (ISO/IEC 15938-17:2022), which has been technically revised.

The main changes are as follows:

- Support for incremental compression of updates of neural networks respective to a base model,
- Additional sparsification tools,
- Additional entropy coding tools, leveraging dependencies in incremental updates,
- Additional quantization tools, including representation as residuals of updates, and
- Additional high-level syntax, covering the new coding tools as well as more metadata (e.g. performance metrics).

A list of all parts in the ISO/IEC 15938 series can be found on the ISO and IEC websites.

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 and www.iec.ch/national-committees.

Introduction

Artificial neural networks have been adopted for a broad range of tasks in multimedia analysis and processing, media coding, data analytics and many other fields. Their recent success is based on the feasibility of processing much larger and complex neural networks (deep neural networks, DNNs) than in the past, and the availability of large-scale training data sets. As a consequence, trained neural networks contain a large number of parameters and weights, resulting in a quite large size (e.g. several hundred MBs). Many applications require the deployment of a particular trained network instance, potentially to a larger number of devices, which may have limitations in terms of processing power and memory (e.g. mobile devices or smart cameras), and also in terms of communication bandwidth. Any use case, in which a trained neural network (or its updates) needs to be deployed to a number of devices thus benefits from a standard for the compressed representation of neural networks.

Considering the fact that compression of neural networks is likely to have a hardware dependent and hardware independent component, this document is designed as a toolbox of compression technologies. Some of these technologies require specific representations in an exchange format (i.e. sparse representations, adaptive quantization), and thus a normative specification for representing outputs of these technologies is defined. Others do not at all materialize in a serialized representation (e.g. pruning), however, also for the latter ones required metadata is specified. This document is independent of a particular neural network exchange format, and interoperability with common formats is described in the annexes.

This document thus defines a high-level syntax that specifies required metadata elements and related semantics. In cases where the structure of binary data is to be specified (e.g. decomposed matrices) this document also specifies the actual bitstream syntax of the respective block. Annexes to the document specify the requirements and constraints of compressed neural network representations; as defined in this document; and how they are applied.

- [Annex A](#) specifies the implementation of this document with the Neural Network Exchange Format (NNEF¹⁾), defining the use of NNEF to represent network topologies in a compressed neural network bitstream.
- [Annex B](#) provides recommendations for the implementation of this document with the Open Neural Network Exchange Format (ONNX²⁾), defining the use of ONNX to represent network topologies in a compressed neural network bitstream.
- [Annex C](#) provides recommendations for the implementation of this document with the PyTorch³⁾ format, defining the reference to PyTorch elements in the network topology description of a compressed neural network bitstream.
- [Annex D](#) provides recommendations for the implementation of this document with the Tensorflow⁴⁾ format, defining the reference to Tensorflow elements in the network topology description of a compressed neural network bitstream.
- [Annex E](#) provides recommendations for the carriage of tensors compressed according to this document in third party container formats.
- [Annex F](#) provides recommendations for the naming of common performance metrics to specify the metric that was used for validation.

1) NNEF is the trademark of a product owned by The Khronos® Group. This information is given for the convenience of users of this document and does not constitute an endorsement by ISO/IEC of the product named.

2) ONNX is the trademark of a product owned by LF PROJECTS, LLC. This information is given for the convenience of users of this document and does not constitute an endorsement by ISO/IEC of the product named.

3) PyTorch is the trademark of a product supplied by Facebook, Inc. This information is given for the convenience of users of this document and does not constitute an endorsement by ISO/IEC of the product named.

4) TensorFlow is the trademark of a product supplied by Google LLC. This information is given for the convenience of users of this document and does not constitute an endorsement by ISO/IEC of the product named.

- [Annex G](#) provides recommendations for implementing the encoding side of some of the compression tools.

The compression tools described in this document have been selected and evaluated for neural networks used in applications for multimedia description, analysis and processing. However, they may be useful for the compression of neural networks used in other applications and applied to other types of data.