



Experiential Networked Intelligence (ENI); Introduction to Artificial Intelligence Mechanisms for (Modular Systems)

[ETSI GR ENI 018 V2.1.1 \(2021-08\)](https://standards.iteh.ai/catalog/standards/sist/a1419730-d9cd-4a2f-b818-30910265a1e9/etsi-gr-eni-018-v2-1-1-2021-08)

<https://standards.iteh.ai/catalog/standards/sist/a1419730-d9cd-4a2f-b818-30910265a1e9/etsi-gr-eni-018-v2-1-1-2021-08>

Disclaimer

The present document has been produced and approved by the Experiential Networked Intelligence (ENI) ETSI Industry Specification Group (ISG) and represents the views of those members who participated in this ISG. It does not necessarily represent the views of the entire ETSI membership.

ReferenceDGR/ENI-0028_AI_Mechanisms

Keywordsartificial intelligence, cognition, design, software

ETSI650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - APE 7112B
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° w061004871

Important notice

The present document can be downloaded from:

<http://www.etsi.org/standards-search>

The present document may be made available in electronic versions and/or in print. The content of any electronic and/or print versions of the present document shall not be modified without the prior written authorization of ETSI. In case of any existing or perceived difference in contents between such versions and/or in print, the prevailing version of an ETSI deliverable is the one made publicly available in PDF format at www.etsi.org/deliver.

Users of the present document should be aware that the document may be subject to revision or change of status.

Information on the current status of this and other ETSI documents is available at

<https://portal.etsi.org/TB/ETSIDeliverableStatus.aspx>

If you find errors in the present document, please send your comment to one of the following services:

<https://portal.etsi.org/People/CommitteeSupportStaff.aspx>

Notice of disclaimer & limitation of liability

The information provided in the present deliverable is directed solely to professionals who have the appropriate degree of experience to understand and interpret its content in accordance with generally accepted engineering or other professional standard and applicable regulations.

No recommendation as to products and services or vendors is made or should be implied.

No representation or warranty is made that this deliverable is technically accurate or sufficient or conforms to any law and/or governmental rule and/or regulation and further, no representation or warranty is made of merchantability or fitness for any particular purpose or against infringement of intellectual property rights.

In no event shall ETSI be held liable for loss of profits or any other incidental or consequential damages.

Any software contained in this deliverable is provided "AS IS" with no warranties, express or implied, including but not limited to, the warranties of merchantability, fitness for a particular purpose and non-infringement of intellectual property rights and ETSI shall not be held liable in any event for any damages whatsoever (including, without limitation, damages for loss of profits, business interruption, loss of information, or any other pecuniary loss) arising out of or related to the use of or inability to use the software.

Copyright Notification

No part may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm except as authorized by written permission of ETSI.

The content of the PDF version shall not be modified without the written authorization of ETSI.

The copyright and the foregoing restriction extend to reproduction in all media.

© ETSI 2021.
All rights reserved.

Contents

| | |
|---|----|
| Intellectual Property Rights | 4 |
| Foreword..... | 4 |
| Modal verbs terminology..... | 4 |
| Executive summary | 4 |
| 1 Scope | 5 |
| 2 References | 5 |
| 2.1 Normative references | 5 |
| 2.2 Informative references..... | 5 |
| 3 Definition of terms, symbols and abbreviations..... | 5 |
| 3.1 Terms..... | 5 |
| 3.2 Symbols..... | 6 |
| 3.3 Abbreviations | 6 |
| 4 Artificial Intelligence for Modular Systems..... | 6 |
| 4.1 Introduction | 6 |
| 4.2 Types of Learning | 6 |
| 4.2.1 Introduction..... | 6 |
| 4.2.2 Supervised..... | 7 |
| 4.2.3 Semi-Supervised..... | 7 |
| 4.2.4 Unsupervised | 7 |
| 4.2.5 Reinforcement Learning | 7 |
| 4.2.6 Feature Learning | 7 |
| 4.2.7 Rule-Based Learning | 7 |
| 4.2.8 Explanation-Based Learning..... | 7 |
| 4.2.9 Federated Learning | 7 |
| 4.2.10 Active Learning | 8 |
| 4.3 Model Training..... | 8 |
| 4.3.1 Online Model Training..... | 8 |
| 4.3.2 Offline Model Training..... | 8 |
| 4.3.3 Comparison..... | 8 |
| 4.4 Bias..... | 9 |
| 4.4.1 Definition..... | 9 |
| 4.4.2 Types of Bias | 9 |
| 4.4.2.1 Algorithmic Bias | 9 |
| 4.4.2.2 Technical Bias..... | 9 |
| 4.4.2.3 Inductive Bias | 10 |
| 4.4.2.4 Emergent Bias | 10 |
| 4.5 Ethics and Ethical Decision-Making | 10 |
| 4.5.1 Introduction..... | 10 |
| 4.5.2 Definitions | 10 |
| 4.5.3 Embedding Ethical Decision-Making into AI Systems | 10 |
| 4.5.4 Existing Work in Standards and Fora | 11 |
| 4.6 Natural Language Processing using AI: an Overview | 11 |
| 4.6.1 Introduction..... | 11 |
| 4.6.2 Embeddings | 12 |
| 4.6.3 Long Short-Term Memory Models..... | 12 |
| 4.6.4 Attention | 13 |
| 4.6.5 Transformer Models | 13 |
| 4.6.5.1 Introduction..... | 13 |
| 4.6.5.2 BERT Models..... | 13 |
| 4.6.5.3 GPT Models | 13 |
| 4.6.5.4 Sparse Transformers..... | 13 |
| 4.6.5.5 Mixture of Experts Model..... | 13 |
| 5 Summary and Recommendations | 14 |
| History | 15 |

Intellectual Property Rights

Essential patents

IPRs essential or potentially essential to normative deliverables may have been declared to ETSI. The declarations pertaining to these essential IPRs, if any, are publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: "*Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards*", which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (<https://ipr.etsi.org/>).

Pursuant to the ETSI Directives including the ETSI IPR Policy, no investigation regarding the essentiality of IPRs, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Trademarks

The present document may include trademarks and/or tradenames which are asserted and/or registered by their owners. ETSI claims no ownership of these except for any which are indicated as being the property of ETSI, and conveys no right to use or reproduce any trademark and/or tradename. Mention of those trademarks in the present document does not constitute an endorsement by ETSI of products, services or organizations associated with those trademarks.

DECT™, **PLUGTESTS™**, **UMTS™** and the ETSI logo are trademarks of ETSI registered for the benefit of its Members. **3GPP™** and **LTE™** are trademarks of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners. **oneM2M™** logo is a trademark of ETSI registered for the benefit of its Members and of the oneM2M Partners. **GSM®** and the GSM logo are trademarks registered and owned by the GSM Association.

ITih STANDARD PREVIEW
(standards.iteh.ai)

Foreword

This Group Report (GR) has been produced by ETSI Industry Specification Group (ISG) Experiential Networked Intelligence (ENI).

ETSI GR ENI 018 V2.1.1 (2021-08)

<https://standards.iteh.ai/catalog/standards/sist/af119790-6969-4a2f-b07f-30910265a1e9/etsi-gr-eni-018-v2-1-1-2021-08>

Modal verbs terminology

In the present document "**should**", "**should not**", "**may**", "**need not**", "**will**", "**will not**", "**can**" and "**cannot**" are to be interpreted as described in clause 3.2 of the [ETSI Drafting Rules](#) (Verbal forms for the expression of provisions).

"**must**" and "**must not**" are **NOT** allowed in ETSI deliverables except when used in direct citation.

Executive summary

The present document specifies a high-level functional abstraction of the ENI System Architecture in terms of Functional Blocks and External Reference Points. This includes describing how different classes of systems interact with ENI. Processes, models, and detailed information are beyond the scope of the present document.

1 Scope

The purpose of the present document is to provide information on different types of AI mechanisms that can be used for cognitive networking and decision making in modern system design. Bias and ethics will also be addressed. This information can be applied to the ENI reference system architecture (and any other applicable ETSI reports or standards).

2 References

2.1 Normative references

Normative references are not applicable in the present document.

2.2 Informative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

NOTE: While any hyperlinks included in this clause were valid at the time of publication, ETSI cannot guarantee their long term validity.

The following referenced documents are not necessary for the application of the present document but they assist the user with regard to a particular subject area.

- [i.1] ETSI GS ENI 005 (V2.1.1): "Experiential Networked Intelligence (ENI); System Architecture".
- [i.2] Cointe, N., Bonnel, G. and Boissier, O.: "Ethical judgment of agents' behaviors in multi-agent systems", AAMAS, pgs 1106-1114, 2016.
<https://standards.iteh.ai/catalog/standards/sist/a1419730-d9cd-4a2f-b818-7d74187c9c38/etsi-gr-eni-018-v2-1-1-2021-08>
- [i.3] Anderson, M. and Anderson, S.L.: "GenEth: A general ethical dilemma analyzer", AAAI, pgs 253-261, 2014.
- [i.4] Koene, A., Smith, A.L., Egawa, T., Mandalh, S. and Hatada, Y.: "IEEE™ P70xx, Establishing Standards for Ethical Technology", KDD, 2018.

NOTE: Available at http://www.kdd.org/kdd2018/files/project-showcase/KDD18_paper_1743.pdf.

3 Definition of terms, symbols and abbreviations

3.1 Terms

For the purposes of the present document, the terms given in ETSI GS ENI 005 [i.1] and the following apply:

active learning: learning algorithm that can query a user interactively to label data with the desired outputs

NOTE: The algorithm proactively selects the subset of examples to be labeled next from the pool of unlabeled data. The idea is that an ML algorithm could potentially reach a higher level of accuracy while using a smaller number of training labels if it were allowed to choose the data it wants to learn from.

batch learning: type of offline learning algorithm that is updated (i.e. retrained) periodically

catastrophic forgetting: tendency of an artificial neural network to forget previously learned information when learning new information

concept drift: not taking changing data and its meanings into account when training an ML model

one-cold vector: $1 \times N$ matrix (vector) used to distinguish each word in a vocabulary from every other word in the vocabulary, where the vector consists of 1s in all cells with the exception of a single 0 in a cell used uniquely to identify the word

one-hot vector: $1 \times N$ matrix (vector) used to distinguish each word in a vocabulary from every other word in the vocabulary, where the vector consists of 0s in all cells with the exception of a single 1 in a cell used uniquely to identify the word

3.2 Symbols

Void.

3.3 Abbreviations

For the purposes of the present document, the following abbreviations apply:

| | |
|------|---|
| AI | Artificial Intelligence |
| ANN | Artificial Neural Network |
| BERT | Bidirectional Encoder Representations from Transformers |
| GPT | Generative Pre-trained Transformer |
| LSTM | Long Short-Term Memory |
| MIT | Massachusetts Institute of Technology |
| ML | Machine Learning |
| NLP | Natural Language Processing |

IT&T STANDARD PREVIEW
(standards.iteh.ai)

4 Artificial Intelligence for Modular Systems

[ETSI GR ENI 018 V2.1.1 \(2021-08\)](https://standards.iteh.ai/catalog/standards/sist/a1419730-d9cd-4a2f-b818-30910265a1e9/etsi-gr-eni-018-v2-1-1-2021-08)

4.1 Introduction

<https://standards.iteh.ai/catalog/standards/sist/a1419730-d9cd-4a2f-b818-30910265a1e9/etsi-gr-eni-018-v2-1-1-2021-08>

Machine learning algorithms learn a solution to a problem from sample data.

Historically, machine learning has focused on non-incremental, offline learning tasks (i.e. where the training set can be constructed a priori and training stops once this set has been duly processed). There are, however, a number of areas, such as agent-based learning and processing sequential data, where learning tasks are inherently incremental.

Machine learning systems that use offline learning do not change their approximation of the target function when the initial training phase has been completed unless the system can detect that the input data has changed significantly. In contrast, machine learning systems that use online learning continuously re-evaluate their target function. The advantage of offline learning is that training is computationally intensive. The advantage of online learning is that if small changes, such as trends, are expected, online training will perform better than offline variants.

The machine learning process adjusts parameters to minimize observed errors; this does not mean that the error rate reaches 0. However, it does mean that if the error rate becomes too high, then the Artificial Neural Network (ANN) needs to be redesigned. This in turn is done by defining a function that minimizes the difference between the observed and desired values.

4.2 Types of Learning

4.2.1 Introduction

In each of the following clauses, each type of algorithm has a large number of variants available. No single learning algorithm works in an optimal manner for all problems.

4.2.2 Supervised

Supervised learning defines a function that maps an input to an output based on example pairs of labelled inputs and outputs. Each input is a tuple that consists of an input object and a desired output value. The learning function analyses the training data and produces a function that can determine the class labels for new data.

There are a large number of algorithms available. No single supervised learning algorithm works in an optimal manner for all problems.

4.2.3 Semi-Supervised

Semi-supervised learning is a hybridisation of supervised and unsupervised learning, where the training data consists of both labelled and unlabelled data.

4.2.4 Unsupervised

Unsupervised learning defines a function that maps an input to an output without the benefit of the data being classified or labelled. The input data can be modelled as probability densities.

4.2.5 Reinforcement Learning

Reinforcement learning uses software agents to take actions in an environment in order to maximize a cumulative reward. In this approach, the learning agent is not told which actions to take, but instead is responsible for discovering which actions yield the highest reward.

4.2.6 Feature Learning

Feature learning analyses raw input data to learn the most important characteristics and behaviour representations of those data that make it easier to discover information from raw data when building different types of predictors (e.g. classifiers).

ITeH STANDARD PREVIEW
(standards.iteh.ai)
ETSI GR ENI 018 V2.1.1 (2021-08)
<https://standards.iteh.ai/catalog/standards/sist/a1419730-d9cd-4a2f-b818-30910265a1e9/etsi-gr-eni-018-v2-1-1-2021-08>

4.2.7 Rule-Based Learning

Rule-based learning uses learned rules to represent the knowledge of a system. This type of system learns rules to make decisions, instead of using a model. These rules are different than other types of rule-based systems because this set of rules are learned, while rules in other types of systems are defined. The rules are typically imperative rules.

4.2.8 Explanation-Based Learning

Explanation-based learning uses an explanation-driven approach that enables a search procedure, constrained by general domain knowledge related to the context of the actual problem, to be used to provide more accurate and efficient learning in knowledge-intensive systems.

4.2.9 Federated Learning

Federated learning is an approach that trains a centralized model using decentralized data that is distributed across multiple entities holding local data samples, without exchanging their data samples. Each device trains the model on their own local data set, and then each client sends a model or model update to a centralized service, which aggregates each client's contribution into one global model. The centralized service then distributes the global model back to the clients.

4.2.10 Active Learning

Active learning is an iterative supervised learning algorithm where the algorithm can actively query an oracle (e.g. a human annotator) to obtain the correct label. This approach enables the learning algorithm to interactively choose the data it will learn from. Active learning iteratively selects the most informative examples to acquire their labels and trains a classifier from the updated training set, which is augmented with the newly selected examples. Unlike conventional supervised learning, it permits a learning model to evolve and adapt to new data. Active learning is concerned with learning accurate classifiers by choosing which examples will be labelled, reducing the labelling effort and the cost of training an accurate model. Active learning is appropriate for machine learning applications where labeled data is costly to obtain but unlabeled data is abundant. Active learning is especially important where objects can have multiple labels that belong to various categories (e.g. a network device has multiple roles, or an image can be labelled as containing both mountains, beach, and ocean). The main challenge in performing

4.3 Model Training

4.3.1 Online Model Training

A dynamic model is a model that is trained online. That is, data is continually ingested and is incorporated into the model through continuous updates. The motivation is that most information sources change, and if the model is not trained on the data changes from its information sources, it becomes stale and out-of-date. Worse, it will likely produce inaccurate, or even incorrect, predictions.

Hence, online machine learning ingests data sequentially; the ingested data is then used to update the best predictor for future data at each step. This is fundamentally different from offline or batch learning algorithms that generate the best predictor by learning on the entire training data set at once. Online learning is a common technique used in areas of machine learning where it is computationally infeasible to train over the entire dataset, requiring the need of out-of-core algorithms. It is also used in situations where it is necessary for the algorithm to dynamically adapt to new patterns in the data, or when the data itself is generated as a function of time, e.g. stock price prediction. Online learning algorithms may be prone to catastrophic forgetting, which can be addressed by using incremental learning in addition to online learning.

[ETSI GR ENI 018 V2.1.1 \(2021-08\)](https://standards.iteh.ai/catalog/standards/sist/a1419730-d9cd-4a2f-b818-728101651e9/etsi-gr-eni-018-v2-1-1-2021-08)

[https://standards.iteh.ai/catalog/standards/sist/a1419730-d9cd-4a2f-b818-](https://standards.iteh.ai/catalog/standards/sist/a1419730-d9cd-4a2f-b818-728101651e9/etsi-gr-eni-018-v2-1-1-2021-08)

[728101651e9/etsi-gr-eni-018-v2-1-1-2021-08](https://standards.iteh.ai/catalog/standards/sist/a1419730-d9cd-4a2f-b818-728101651e9/etsi-gr-eni-018-v2-1-1-2021-08)

4.3.2 Offline Model Training

Offline model training is the use of a static model that is trained once; that trained model is then used for a period of time. The term batch learning is used when an offline model is updated (i.e. retrained) periodically. Batch learning is a compromise between online and offline model training, and seeks to keep the algorithm updated by scheduling training updates. In either case, the learned models are trained once, and then updated as needed.

4.3.3 Comparison

Static models are easier to build and test, but cannot adapt to changing information sources. Static models still need updating on a regular basis; otherwise, the data that the model is trained on because stale and out-of-date.

In contrast, online learning is better suited for those problems where samples are presented over time and where the probability distribution of samples is expected to also change over time.

Online learning models are harder to be managed in production level stage after deployment. Since the model has to churn large amounts of the dataset in real-time, any changes in the pattern of data - or more commonly known as concept drift - will affect the overall performance and prediction of the data. In contrast, Offline learning, however, with the model being constant after the deployment stage (with same types and pattern of data), it is easier to maintain the whole network or cluster with minimal supervision and control.

Table 1: Comparison of Online vs. Offline Learning

| Features | Online Learning | Offline Learning |
|-------------------------------------|--|--|
| Model Complexity | Complexity varies dynamically as a function of changing data | Complexity is much less since the model training is constant |
| Update Period | Model is sequentially trained on one instance at a time | Model is trained after consuming the entire batch of new data |
| Computational Power Required | More computations since model is continuously refined | Fewer and more efficient computations since they are scheduled periodically |
| Implementation | Harder to implement and manage | Easier to implement and manage |
| Usage | Applications where new data and patterns are frequently emerging (e.g. networking, health, finance) | Applications where data changes very slowly (if at all), such as image classification |
| Potential Problems | Prone to <i>catastrophic forgetting</i> (i.e. forgetting previously learned information when new information is learned) | The predictor is prone to <i>concept drift</i> (i.e. not taking changing data and its meanings into account) |

4.4 Bias

4.4.1 Definition

Bias is defined as "the systematic difference in treatment of certain objects, ideas, or people in comparison to others". In this definition, *systematic* means that the difference in treatment is predictable and typically constant.

4.4.2 Types of Bias

4.4.2.1 Algorithmic Bias (standards.iteh.ai)

Algorithmic bias is defined as "an algorithm that possesses systematic and repeatable errors that create unfair outcomes". In this definition, "algorithmic" means the combination and interaction of the algorithm with data that it uses and outputs that it provides. Algorithmic bias may exist even when there is no intention by the algorithm developer to discriminate. For example, the data set may contain historical discrimination. A more subtle issue is when the data set contains correlations between sensitive privacy attributes.

In the fields of statistics and artificial intelligence, bias typically has five causes:

- Variables might not be independently identically distributed.
- Errors might be concentrated in a particular class in the data set.
- Unfair sampling (e.g. incorrect collection or selection of objects from a statistical population).
- Unfair training (e.g. training set does not contain a diverse set of data).
- Context-specific implementation problems.

While hard to detect, the first four types of bias can be fixed using a set of processes developed specifically for each problem. The fifth source of bias is context-specific, and consists of technical and emergent bias.

4.4.2.2 Technical Bias

Technical bias is an epistemological problem. Put another way, when developers translate constructs into a form that computers can process, the translation process itself may introduce bias. Hence, understanding how features are engineered and performed and processed, and where errors can be introduced, is critical for avoiding technical bias.

There are four types of technical bias. Labeling bias occurs when the label assigned to the training data does not adequately represent the underlying semantics of that outcome. Measurement bias is introduced when an inappropriate measurement scale is used (e.g. changing from an ordinal to a nominal scale), since this affects both the sample and how the consumer using the sample is represented. Modeling bias occurs when features are introduced that are not truly representative of the population under study and, more importantly, are predictive of the outcome. This can be viewed as a compression problem in which high-dimensional data that interact in multiple ways are represented by lower-dimensional data that interact in more limited ways. Optimization bias is introduced when the model parameters are tuned to achieve certain objectives.

4.4.2.3 Inductive Bias

The inductive bias of a learning algorithm is defined by how the algorithm chooses among all models that fit the data being examined equally well. The underlying problem is how to choose a learner's hypothesis space so that it is large enough to contain a solution to the problem being learnt, yet small enough to ensure reliable generalization from reasonably-sized training sets. Learning from examples thus has an inherent bias that determines which model is best, even though the instances have not yet been observed by the learner. Hence, the goal is to ensure that the learning program has a desirable and productive inductive bias.

4.4.2.4 Emergent Bias

Emergent bias occurs once the ML model is in use. It is caused by such factors as:

- the introduction of new forms and types of data that were not considered and trained by the existing algorithm;
- a mismatch between the original purpose of the ML system and the (new) application that is using it;
- a change in the underlying environment that the ML system is being used on (e.g. the introduction of COVID-19 and its effects on retail prediction systems).

IT'S STANDARD REVIEW
(standards.iteh.ai)

4.5 Ethics and Ethical Decision-Making

<https://standards.iteh.ai/catalog/standards/sist/a1419730-d9cd-4a2f-b818-30910265a1e9/etsi-gr-eni-018-v2-1-1-2021-08>

4.5.1 Introduction

<https://standards.iteh.ai/catalog/standards/sist/a1419730-d9cd-4a2f-b818-30910265a1e9/etsi-gr-eni-018-v2-1-1-2021-08>

There are many definitions of ethics. One definition [i.2] defines ethics as a normative philosophical discipline of how a person or object should act towards others. It comprises three dimensions:

- 1) Consequentialist Ethics: an agent is ethical if and only if it considers the consequences of each decision and chooses the decision that has the most moral outcome.
- 2) Deontological Ethics: an agent is ethical if and only if it respects obligations, duties, and rights appropriate for a given situation.
- 3) Virtue Ethics: an agent is ethical if and only if it acts according to a set of moral values.

4.5.2 Definitions

An *ethical dilemma* is a situation in which any available decision leads to infringing one or more ethical principles.

4.5.3 Embedding Ethical Decision-Making into AI Systems

An approach to embedding ethics into AI systems is based on the above four notions. The hardest is to translate ethical dilemmas to a form that a computer is able to understand. Ethical issues have important, social, political, and legal connotations that typically are not part of the skill set of an AI developer. Consequently, software tools are used to detecting and removing ethics in AI systems. Two types of software tools have emerged:

- 1) those based on expert review of proposed decisions; and
- 2) those based on crowdsourcing mechanisms.