ETSI TR 104 031 V1.1.1 (2024-01)

TECHNICAL REPORT

Securing Artificial Intelligence (SAI); Collaborative Artificial Intelligence

Reference

DTR/SAI-003

Keywords

artificial intelligence, model, trust

ETSI

650 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:
https://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/CommiteeSupportStaff.aspx

If you find a security vulnerability in the present document, please report it through our Coordinated Vulnerability Disclosure Program:
https://www.etsi.org/standards/coordinated-vulnerability-disclosure

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

Contents

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.

Foreword

This Technical Report (TR) has been produced by ETSI Technical Committee Securing Artificial Intelligence (SAI).

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.

1 Scope

The present document describes collaborative Artificial Intelligence (AI) from securing AI perspectives. Collaborative AI could take place among AI agents, between AI agents and human, and even among people who provide and use AI. As such, the security and performance of collaborative AI may range from AI/ML-specific issues to other system-specific issues (e.g. AI-to-AI communications, joint computing and communicating optimization, etc.). The present document investigates collaborative AI use cases and involved technical aspects, and analyses potential security and performance issues (e.g. AI-to-AI communications, trustworthy collaboration, etc.) among those AI-related entities. The present document also overviews existing approaches to tackle and/or mitigate these issues.

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 GR SAI 009 (V1.1.1): "Securing Artificial Intelligence (SAI); Artificial Intelligence Computing Platform Security Framework".

[i.2] J. Urbanek: "Introducing Mephisto: A new platform for more open, collaborative data collection", March 2022.

[i.3] Q. Yang, Y. Liu, T. Chen and Y. Tong: "Federated Machine Learning: Concept and Applications", ACM Transactions on Intelligent System and Technology, vol. 10, no. 2, March 2019.

[i.4] K. Zhang, Z. Yang, T. Başar: "Multi-Agent Reinforcement Learning: A Selective Overview of Theories and Algorithms", April 2021.

[i.5] R. Sim, Y. Zhang, M. C. Chan and B. Low: "Collaborative Machine Learning with Incentive-Aware Model Rewards", International Conference on Machine Learning (ICML), 2020.

[i.6] N. Wang, Y. Duan and J. Wu: "Accelerate Cooperative Deep Inference via Layer-wise Processing Schedule Optimization", 2021 International Conference on Computer Communications and Networks (ICCCN), 2021.

[i.7] P. Barham, A. Chowdhery, J. Dean, et al.: "Pathways: Asynchronous Distributed Dataflow for ML", Proceedings of the 5th MLSys Conference, Santa Clara, CA, USA, 2022.

[i.8] F. Zhuang, Z. Qi, et al.: "A Comprehensive Survey on Transfer Learning", June 2020.

[i.9] ETSI GR SAI 004 (V1.1.1): "Securing Artificial Intelligence (SAI); Problem Statement".

[i.10] ETSI GR SAI 002 (V1.1.1): "Securing Artificial Intelligence (SAI); Data Supply Chain Security".

[i.11] C. Xie, M. Chen, P. Chen and B. Li: "CRFL: Certifiably Robust Federated Learning against Backdoor Attacks", in Proceedings of the 38th International Conference on Machine Learning, PMLR 139:11372-11382, 2021.

[i.12] Z. Yang, Y. Shi, Y. Zhou, Z. Wang and K. Yang: "Trustworthy Federated Learning via Blockchain," in IEEE Internet of Things Journal, 2022, doi: 10.1109/JIOT.2022.3201117.

[i.13] H. Kim, J. Park, M. Bennis and S. -L. Kim: "Blockchained On-Device Federated Learning", in IEEE Communications Letters, vol. 24, no. 6, pp. 1279-1283, June 2020, doi: 10.1109/LCOMM.2019.2921755.

[i.14] V. Mothukuri, R. M. Parizi, S. Pouriyeh, Y. Huang, A. Dehghantanha, G. Sivastava: "A Survey on Security and Privacy of Federated Learning", Future Generation Computer Systems, Volume 115, 2021, pp. 619-640.

[i.15] B. Wang, Y. Yao, B. Viswanath, H. Zheng, and B. Y. Zhao: "With Great Training Comes Great Vulnerability: Practical Attacks against Transfer Learning", in 27th USENIX Security Symposium (USENIX Security 18), pp. 1281-1297, 2018.

[i.16] M. Xu, D. T. Hoang, J. Kang, D. Niyato, Q. Yan and D. I. Kim: "Secure and Reliable Transfer Learning Framework for 6G-Enabled Internet of Vehicles", in IEEE Wireless Communications, vol. 29, no. 4, pp. 132-139, August 2022.

[i.17] Y. Gao, and Y. Cui: "Deep Transfer Learning for Reducing Health Care Disparities Arising from Biomedical Data Inequality," Nature Communications 11, no. 1, 2020.

[i.18] S. A. Seshia, D. Sadigh and S. S. Sastry: "Toward Verified Artificial Intelligence", Communication of ACM, 65(7), pp. 46-55, July 2022.

[i.19] K. Xu, H. Ding, L. Guo and Y. Fang: "A Secure Collaborative Machine Learning Framework Based on Data Locality", IEEE Global Communications Conference (GLOBECOM), pp. 1-5, December 2015.

3 Definition of terms, symbols and abbreviations

3.1 Terms

Void.

3.2 Symbols

Void.

3.3 Abbreviations

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

6G	Six Generation
AI	Artificial Intelligence
AIA	AI Agent
AIA4IK	AI Agent for Inferring Knowledge
AIA4LM	AI Agent for Learning a Model
AIA4LI	AI Agent for Learning and Inference
AIA4PD	AI Agent for Provisioning Data
AIH	AI Host
B-FL	Blockchain-based FL
CRFL	Certifiably Robust Federated Learning
FL	Federated Learning
FLC	FL Client
FLS	FL Server
GR	Group Report
GPS	Global Positioning System
IID	Independent and Identical Distribution

IoV	Internet of Vehicles
MARL	Multi-Agent Reinforcement Learning
ML	Machine Learning
RL	Reinforcement Learning
SAI	Securing Artificial Intelligence
TL	Transfer Learning

4 Overview

4.1 Introduction

An AI system usually contains one or multiple AI Agents (AIAs), which learn and/or exploit an AI model based on different AI schemes such as deep learning, federated learning, reinforcement learning, and/or a combination of them. AI agents usually reside in different physical or logical nodes (e.g. devices, servers, a virtual machine in the cloud), referred to as AI Hosts (AIHs) (see Figure 4.1-1). The concept of "AIHs" is consistent with the concept of "AI computing platform" described in ETSI GR SAI 009 [i.1]. Each AI agent usually hosts and runs an AI task, which could be a task for learning an AI model according to an AI algorithm (e.g. a deep learning algorithm, a federated learning algorithm, a reinforcement learning algorithm) or a task for using an AI model to infer knowledge. Deep learning and reinforcement learning usually uses one AI agent, while federated learning utilizes multiple AI agents working collaboratively to learn an AI model. An AI algorithm could be supervised by relying on tagged training data or unsupervised without the use of any tagged data.

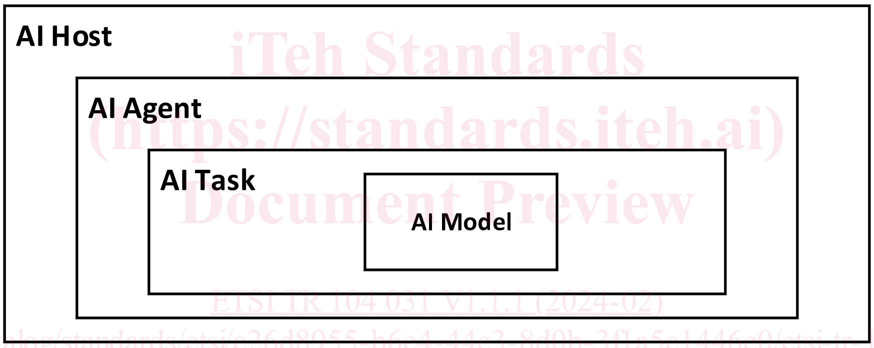

Figure 4.1-1: AI Host, AI Agent, AI Task, and AI Model

4.2 AI pipeline

A general AI pipeline for supervised learning is illustrated in Figure 4.2-1, which usually consists of multiple stages:

1) task configuration stage that includes the deployment of AI agents/tasks by an AI application or user;

2) data preparation stage that includes data collection and optional feature engineering/extraction;

3) training stage for learning an AI model;

4) validation stage for testing and validating the learned AI model;

5) model deployment stage for deploying and transferring the validated AI model; and

6) inference stage for inferring and predicting future knowledge using new data as inputs (referred to as input data for inference).

The outcome/results from the inference stage could be leveraged to action or trigger going back to training stage to re-train the AI model.

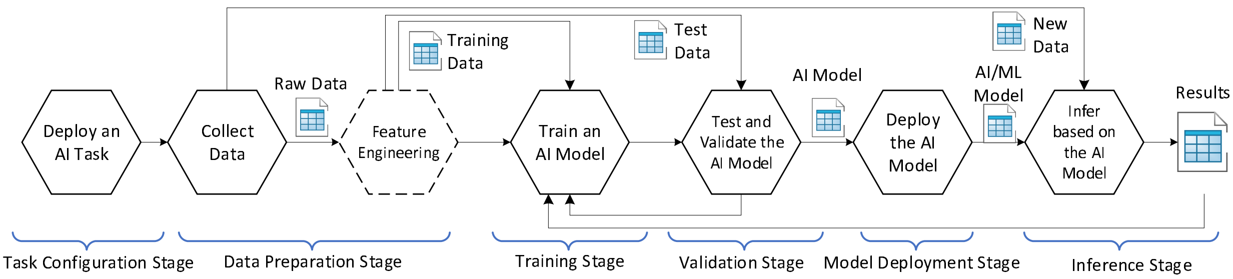

Figure 4.2-1: General AI Pipeline for Supervised Learning

Dependent on AI deployment choices, an AI agent could be:

1) An AI Agent for Learning a Model (AIA4LM) that is only responsible for learning an AI model;

2) An AI Agent for Inferring Knowledge (AIA4IK) that uses a learned AI model for inference and predication; and

3) An AI Agent for Learning and Inference (AIA4LI). AI model transfer generally occurs between an AIA4LM and an AIA4IK or between multiple AIA4LIs.

4.3 Collaborative AI

Collaborative AI can be classified to the following categories:

- Category 1: Agent-to-Agent Collaboration. In this category, more than one AI agent collaborate with each other during the partial or the whole AI pipeline to perform an AI task. Examples of this collaborative AI category include collaborative data collection [i.2], federated learning [i.3], multi-agent reinforcement learning [i.4], collaborative machine learning [i.5], collaborative inference [i.6], Pathway as next-generation AI [i.7].

- Category 2: Agent-to-Human Collaboration. In this category, AI agents and human collaboratively work together to solve a shared task.

- Category 3: Collaborative AI Marketplace. In this category, human collaboratively exchanges and shares data, training capability, AI models, and/or inferred knowledge via an open AI marketplace.

5 Use cases

5.1 Introduction

Clause 5 describes three categories of collaborative AI use cases, which are collaborative distributed AI/ML, Human-AI collaboration, and collaborative AI/ML marketplace.

5.2 Collaborative distributed AI/ML

5.2.1 Federated Learning (FL)

Federated Learning (FL) [i.3] is a framework for distributed machine learning, where a FL Server (FLS) and FL Clients (FLCs) collaboratively learn an AI model. In FL, training data is maintained locally at multiple distributed FLCs (e.g. mobile devices). Each FLC performs local training (e.g. deep learning), generates local model updates, and sends local model updates to the FLS. The FLS as a central entity aggregates local model updates received from FLCs and generates global model updates, which will be sent to all participating FLCs for the next training round. In fact, the FLS hosts an AI agent for learning, while each FLC has an AI agent that could be for both learning and inferring.

Figure 5.2.1-1 illustrates the general federated learning process, where the FLS and FLCs jointly take the following steps to perform an FL task:

- step 1: The FLS selects a set of FLCs to participate in a FL task;
- step 2: The FLS configures the FL task to each selected FLC;
- step 3: The FLS sends an initial global model to each selected FLC;
- step 4: Each FLC independently trains the global model based on the received initial global model and its local data;
- step 5: After each training round, each FLC generates a local model update and sends it to the FLS;
- step 6: The FLS receives local model updates from all FLCs, aggregates them, and generates new global model update. The FLS may need to wait for receiving local model updates from all FLCs before performing the aggregation (i.e. synchronous FL) or it can start the aggregation after receiving the local model updates from some of FLCs (i.e. asynchronous FL). Note that the FLS may (re)select some new FLCs for next training round;
- step 7: Similar to Step 3, the FLS sends the global model updates to all FLCs; and
- step 8: Similar to Step 4, each FLC starts next local training.

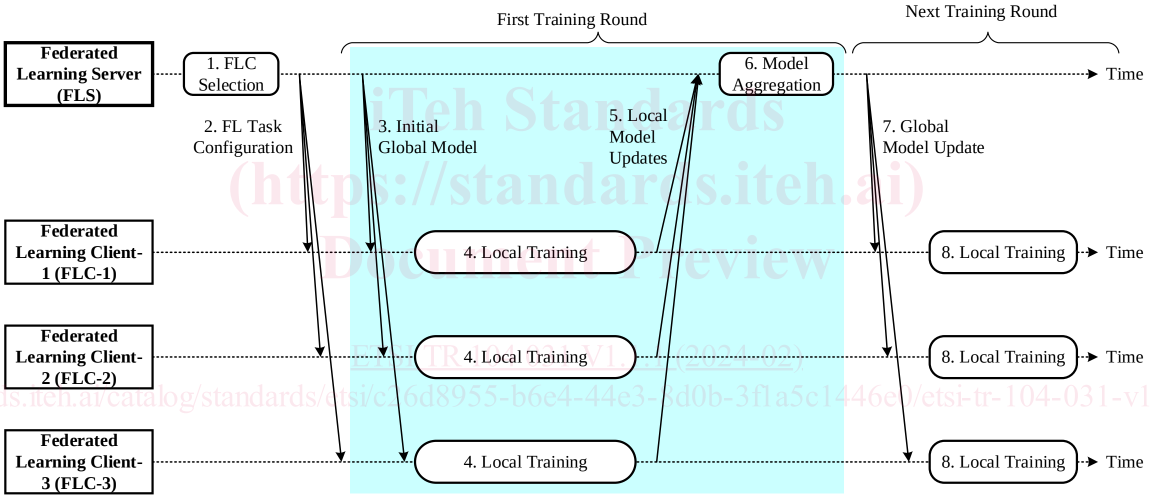

Figure 5.2.1-1: Federated learning

Several advantages of federated learning include:

1) improved data privacy-preservation since training data stays at FLCs;

2) reduced communication overhead since it is not required to collect/transmit training data to a central entity; and

3) improved learning speed since model training now leverages distributed computation resources at FLCs.

However, FL needs to transmit model updates among AI agents (i.e. between the FLS and FLCs), which introduces security issues and additional communication overhead compared to centralized machine learning. Also, FL requires data at all FLCs follow an Independent and Identical Distribution (IID) (i.e. IID-data) in order to achieve a good learning performance. In addition, FL inherits some potential security issues and threats such as data poisoning and model poisoning attacks.

There are many real scenarios where FL is used to learn a global AI model without collecting training data from end devices and/or protecting data privacy. For example, an end device such as a smart phone nowadays has more sensing capabilities (e.g. camera, sensors), which generate different types of data streams. A global AI model can be learned from these data streams from different end devices. Instead of collecting such data streams from end devices to cloud, each end device performs local training independently and reports its local model updates to an FLS, which aggregates model updates received from multiple end devices and produce the global AI model.

5.2.2 Transfer Learning (TL)

Transfer Learning (TL) has attracted much attention in recent years [i.8] due to the fact that many AI tasks are more or less relevant. In general, TL uses the source AI model trained from the source domain as a starting point to train the target AI model in the target domain. For example, the model parameters or the model structure of the pre-trained source AI model is transferred from the source AI agent to the target AI agent and is used by the target AI agent to train the new target AI model in the target domain (see Figure 5.2.2-1).

There are several TL methods such as Pre-train and Fine-tune, Domain Adaptation, Domain Generalization, and Meta-learning. Taking the Pre-train and Fine-tune method as an example, the first step is to find a pre-trained AI model that can be used for the new problem. It is important to choose a pre-trained AI model carefully.

EXAMPLE: An AI model for riding a bicycle cannot be used for training a self-driving car model and maintain trust.

After a pre-trained AI model is determined, there are usually several approaches to transfer and leverage the pre-trained AI model, such as:

- To remove the output layer of the pre-trained AI model and use it as a feature extractor for the new training data from the target domain.
- Another approach is to transfer the structure of the pre-trained AI model, but re-train all the weights with the new training data from the target domain.
- A different approach is to re-train specific layers but reuse other layers of the pre-trained AI model. For example, for a neural network model, some lower layer of the network (which are used to identify the underlying features of various objects such as boundaries and shapes) can be reused as they are, while only some higher layers (which are used to identify advanced features such as the specific appearance of the face) will be retrained.

TL requires communications between AI agents (i.e. the AI model knowledge transfer from the source AI agent to the target AI agent), which introduces security issues. In addition, since TL relies on the pre-trained AI model, it inherits potential threats to the pre-trained AI model (e.g. data poisoning and backdoor attacks to the pre-trained AI model).

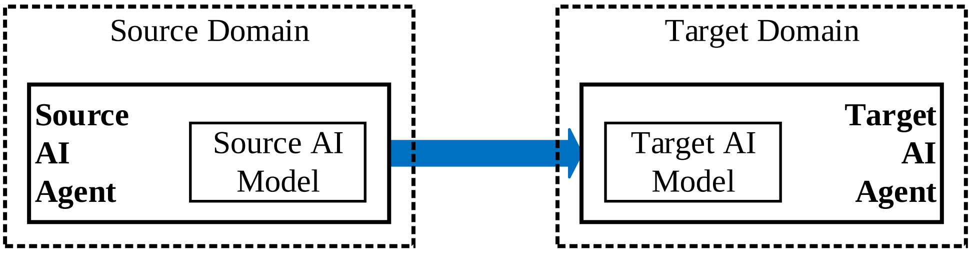

Figure 5.2.2-1: Transfer learning

5.2.3 Multi-Agent Reinforcement Learning (MARL)

Different from traditional supervised or unsupervised machine learning, Reinforcement Learning (RL) is a continuous machine learning paradigm, often used to solve sequential decision-making problems. An RL agent keeps interacting with the environment to gain real experience (i.e. samples); in the meantime, it keeps learning from the gained real experience an optimal policy, which is usually used to control or influence the environment and in turn new real experience will be generated.