



Human-centric algebraic machine learning

Open source experimentation tools - Final

Deliverable D6.6

WP6 - System and Tools



This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 952091.

Consortium



ALGEBRAIC AI



Champalimaud
Foundation



EPROSIMA
The Middleware Experts



German
Research Center
for Artificial
Intelligence



TECHNISCHE UNIVERSITÄT
KAISERSLAUTERN

uc3m



Project

Title: ALMA: Human Centric Algebraic Machine Learning
 Acronym: ALMA
 Coordinator: Proyectos y Sistemas de Mantenimiento SL (eProsima)
 Grant agreement: 952091
 Call: H2020-EIC-FETPROACT-2019
 Program: Horizon 2020
 Start: 01 September 2020
 Duration: 54 months
 Website: <https://alma-ai.eu>
 E-mail: alma@eprosima.com
 Consortium: **Proyectos y Sistemas de Mantenimiento SL (eProsima) (EPROS)**, Spain
Champalimaud Foundation (CF), Portugal
ALGEBRAIC AI (ALG), Spain
German Research Center for Artificial Intelligence (DFKI), Germany
Inria (Inria), France
TU Kaiserslautern (TUK), Germany
FIWARE Foundation e.V. (FF), Germany
Technical Research Centre of Finland (VTT), Finland
Carlos III University (UC3M), Spain

Deliverable

Number: **D6.6**
 Title: **Open source experimentation tools - Final**
 Month: 54
 Work Package: WP6 - System and Tools
 Work Package leader: Proyectos y Sistemas de Mantenimiento SL (eProsima)
 Deliverable leader: Algebraic AI
 Deliverable type: DEM: Demonstrator, pilot, prototype
 Dissemination level: Public (PU)
 Date of submission: 28/02/2025
 Version: v2.2
 Status: Finished

Version history

Version	Date	Responsible	Author/Reviewer	Comments
v1.0	08/06/2024	EPROS	Raúl Sánchez-Mateos Lizano	D6.4 - First Draft
v1.1	01/07/2024	ALG	Nabil Abderrahaman	Add toolkit content
v2.0	25/02/2025	ALG	Nabil Abderrahaman	Update tools content
v2.1	25/02/2025	EPROS	Denisa Alexandru	Review AML-IP tutorials and future work
v2.2	25/02/2025	EPROS	Raúl Sánchez-Mateos	Final review and proofreading

Executive summary

Algebraic Machine Learning (AML) is a relatively new machine-learning technique based on algebraic representations of data. Unlike statistical learning, AML algorithms are robust regarding the statistical properties of the data and are parameter-free. The aim of the EU-funded ALMA project is to leverage AML properties to develop a new generation of interactive, human-centric machine learning systems. These systems are expected to reduce bias and prevent discrimination, remember what they know when they are taught something new, facilitate trust and reliability and integrate complex ethical constraints into human-artificial intelligence systems. Furthermore, they are expected to promote distributed, collaborative learning.

This report corresponds to *Deliverable D6.6 - Open source experimentation tools - Final* of the ALMA project, and introduces a collection of Open Source tools and guides for AML. The AML Toolkit contains documentation, tutorials, tools for working with AML embeddings, and the AML-IP platform for distributed AML training. All the content presented in this document is open source and can be accessed online in

`aml-tools.readthedocs.io`.

Contents

Executive Summary	3
Contents	4
Acronyms	5
1 Introduction	6
1.1 Access to AML Toolkit	6
1.2 Partners involved	6
2 Open source experimentation tools	7
2.1 AML Toolkit Overview	7
2.1.1 Installation	7
2.1.2 AML Description Language Tooling	7
2.1.3 AML Hardware Toolkit	8
2.1.4 AML Integrating Platform Framework	8
2.1.5 AML Dashboard	8
2.1.6 Tutorials	8
2.1.7 Demos	9
3 Conclusions	10

Acronyms

AI	Artificial Intelligence.
AML	Algebraic Machine Learning.
AML-DL	AML Description Language.
AML-IP	AML Integrating Platform.
ML	Machine Learning.

1 Introduction

The ALMA research and innovation project aims to change the Artificial Intelligence (AI) paradigm by creating a new approach to Machine Learning (ML) known as AML. AML is a novel ML technique based on the algebraic representation of data. The goal of the EU-funded ALMA project is to leverage the properties of AML to develop a new generation of interactive, human-centered machine learning systems.

Here we introduce the AML Toolkit, a set of tools and guides for AML Description Language (AML-DL) and AML Integrating Platform (AML-IP). It contains documentation, tutorials, and demos to create, debug and analyse AML-DL embeddings, and to compute distributed AML models using AML-IP distributed platform following different strategies. All the content presented in this document is Open Source and can be accessed online.

1.1 Access to AML Toolkit

The AML Toolkit can be accessed in the following address:

`aml-tools.readthedocs.io`

The AML Toolkit is kept online so updates and fixes can be implemented with great agility. We also make use of `www.readthedocs.io`, which has a wide community and its use is wide-spread among the open source community.

The documentation is organised into sections to facilitate the navigation, and ordered in such a way to facilitate the natural learning curve for new users.

1.2 Partners involved

Team Leader:

- Proyectos y Sistemas de Mantenimiento SL (eProsima) (EPROS), Spain

Partners:

- ALGEBRAIC AI (ALG), Spain

2 Open source experimentation tools

AML Toolkit is a set of software tools that allows users to develop, train, and deploy Algebraic Machine Learning (AML) models. These tools are designed to be used by researchers, developers, and industry professionals who want to implement AML models in their applications. AML Toolkit aims to provide users, developers, and researchers with the documentation, guides, and examples needed to start working with AML.

These tools cover the entire process of developing AML applications, from data collection and preprocessing, to generating embeddings for problem description, validating those embeddings, training the model using an AML engine, training using a distributed strategy, performing inference using the resulted AML model, and validating the results.

This framework is developed as part of the ALMA project. The aim of the EU-funded ALMA project is to leverage AML properties to develop a new generation of interactive, human-centric machine learning systems. These systems are expected to reduce bias and prevent discrimination, remember what they know when they are taught something new, facilitate trust and reliability and integrate complex ethical constraints into human-artificial intelligence systems. Furthermore, they are expected to promote distributed and collaborative learning.

The AML Toolkit can be accessed at:

`aml-tools.readthedocs.io`

2.1 AML Toolkit Overview

Below we present a brief description of the sections that can be found on the AML Toolkit online documentation.

2.1.1 Installation

Installation instructions on Linux.

2.1.2 AML Description Language Tooling

AML Description Language (AML-DL) specification, command set and examples to give the user the capability to describe elaborated embeddings with a concise code, and to establish a methodology to build embeddings for complex problems into semilattices and other algebras with idempotent operators.

Architecture and usage of the AML-DL Interpreter, a software tool that transforms human-readable semantic embeddings, written in AML-DL, into AML input data for the AML Engine. The AML Toolkit also provides a set of examples of how to use the AML-DL Interpreter.

AML-DL Consistency Checker user and developer guide, examples of usage, and consistent and inconsistent embedding catalog. The AML-DL Consistency Checker is a software tool to study the consistency of algebraic semantic embeddings into semilattices. Given a semantic embedding expressed using the AML-DL, it can determine if the embedding is consistent or not. If the embedding is not consistent the AML-DL Consistency Checker indicates the potential primary and secondary sources of the inconsistency.

AML-DL Debugger integrates the AML-DL Interpreter together with the AML-DL Consistency Checker. It provides a user interface and allows the user to dynamically create, evaluate, and debug AML embeddings. Its modular architecture brings the opportunity to add new functionality in the future.

2.1.3 AML Hardware Toolkit

AML Hardware Toolkit, a specialized hardware accelerator for AML to improve performance and energy efficiency compared to general-purpose CPUs and GPUs. The architecture is a SIMD processor with parallel cores, each optimized for AML tasks and connected to High Bandwidth Memory (HBM) for high data throughput. It features a custom instruction set for efficient operation control and a PCIe interface for seamless integration with existing systems, enabling faster computation and reduced power consumption.

2.1.4 AML Integrating Platform Framework

AML Integrating Platform (AML-IP), a communications framework in charge of data exchange between AML nodes through local or remote networks. It is designed to allow non-expert users to create and manage a cluster of AML nodes to exploit the distributed and concurrent learning capabilities of AML. Thus, AML-IP is a communication framework that makes the transport protocols abstracted from the user, creating a platform that communicates each node without requiring the user to be concerned about communication issues. It also allows the creation of complex distributed networks with one or multiple users working on the same problem. It also facilitates communication with the Fiware Context Broker through a Fiware Node for managing inference data. This integration provides mechanisms to read data from a context broker entity, request inference, and send the results back to the context broker.

2.1.5 AML Dashboard

AML Dashboard, a user-friendly tool for training, evaluating, and applying AML models to several datasets. It supports data collection, model training, and performance comparison with other ML approaches. The dashboard integrates seven key interfaces: Dataset Interface for data collection, Training Interface for model training, Model Exploration Interface for comparison, Real-Time Exploration for live testing, Model Fetching for retrieving trained models, Context Broker interaction for inference data handling and AML-IP Management for node control and monitoring.

2.1.6 Tutorials

MNIST multiclass classifier with AML: Complete walk-through of an AML example: data preprocessing, embedding, training, model analysis and inference.

AML-IP Collaborative Learning: Full example using AML-IP for collaborative learning. Nodes can communicate their local statistics and share their atomizations based on that information.

AML-IP Workload Distribution: Full example using AML-IP for distributing the computing load. The computation of a model is split among several computing nodes, which return their results to the main node.

Manage data with the AML Dashboard: Full example using AML-Dashboard for the data collection step of an AML model training process.

Train a model with the AML Dashboard: Full example showcasing the training process of an AML model using the AML Dashboard.

Fetch a model with the AML Dashboard: This tutorial showcases the fetching process of an AML model using the AML-IP Collaborative Learning Scenario integrated in the AML-Dashboard.

Make predictions with the AML Dashboard: Full example showcasing the process of making predictions using an AML model through the AML Dashboard, covering both real-time and batch prediction workflows.

Debug an AML-IP network with the AML Dashboard: Full example showcasing the debugging process of an AML-IP network using the Monitor Network State Scenario integrated in the AML Dashboard.

Send data to the Context Broker with the AML Dashboard: Full example showcasing the process of sending data to the Context Broker and retrieving inferences using the AML Dashboard.

Manage AML-IP nodes with the AML Dashboard: This tutorial showcases the process of managing AML-IP nodes (creating, stopping and deleting) using the AML Dashboard.

Parallelize AML training with AML-IP: This tutorial showcases the process of parallelizing the training of an AML model using AML-IP nodes in the context of the Workload Distribution Scenario.

2.1.7 Demos

Collection of demos on how to use the AML Toolkit to create and verify AML embeddings.

Create an AML Embedding: Creation of a complex embedding using AML-DL.

Check consistency: Detailed example of how to verify the consistency of an AML embedding.

3 Conclusions

The ALMA project has developed a comprehensive open-source toolkit to make AML more accessible and efficient. These tools support the entire AML development process, from data collection and preprocessing to embedding generation, validation, model training, inference, and result validation. The AML Toolkit equips users, developers, and researchers with the necessary documentation, guides, and examples to get started and effectively implement AML in their projects.