ARTIFICIAL INTELLIGENCE BASED TRANSFORMATION PROJECTS-THE ROLE OF DATA SCIENCES (RDS)

Antoine Trad

IBISTM, Courbevoie, France

ABSTRACT

The Applied Polymathical/Holistic Mathematical Model for Integrating Data Sciences (AHMM4IDS) supports Enterprise's transformation projects (simply Project). The AHMM4IDS uses various Mathematical Models (MM), that abstract, incorporate, and integrate Data Sciences (DS), AI-Subdomains, Information Communication System (ICS) components with Project's transformed resources. Transformed resources can be services (and artefacts), success factors (or calibration variables), business processes (and scenarios), mixed-methods, AI-Models, and adequate Enterprise Architecture (EA) Models (EAM). MMs, mixed-methods' based services, artefacts, and EAMs can be used to establish set of DS Patterns (DSP) that include DS technics/capabilities, data-platforms' access (and management), algorithms-functions, mapping concepts, unbundled services; to model and implement Decision Making Processes' (DMP) related infrastructure, data-storage(s), components-models, and end-users' integration. The integration of DSPs enforces and automated DMPs, Project's validity-checking, and Gap Analysis (GAPA); which all need adapted interfaces to access Enterprise, Project, Data-storage(s), ICS, EAMs, pool(s) of Artificial Intelligence (AI) services, and other types of resources. On the other hand, DSPs communicate with other, by using Project's and AI components; and can use also various medias-types formats, like the eXtensible Markup Language (XML) format, and many others. Imported (or exported) DSs' contents and structures are combined with other Project's artefacts and components, to deliver DSPs for various AI-Subdomains.

KEYWORDS

Data Sciences, AI-Subdomains, Polymathical mathematical models, Business and common transformation projects, Enterprise architecture, Artificial intelligence, Qualitative and quantitative research, and Critical success factors/areas.

1. INTRODUCTION

AI-Subdomains include BigData, Machine Learning (ML), Deep Learning (DL), and other. The AHMM4IDS is used for Project's AI-Subsystems' integrity-checking, algorithms' integration, GAPA, financial analysis, risk-management, and many other types of strategic DMP related operations. DSPs use MMs (A AHMM4IDS is a set of MMs) and hence In-memory DataSets (IDS) that can be interrelated with mixed-research method(s). These methods are based on mainly qualitative and associated quantitative methods. The RDS takes a transformative enterprise-wide view and not just DS' usage in a specific case of statistics; it also promotes DSPs which support the central reasoning-engine for the qualitative Heuristics Decision-Tree (HDT). Knowing that Projects are complex and have high rates of failure [1].

David C. Wyld et al. (Eds): SIGML, CNSA, MoWiN, CCSEIT, NIAI, AIAP – 2025
pp. 01-21, 2025. CS & IT - CSCP 2025 DOI: 10.5121/csit.2024.150201

MMs and hence the AHMM4IDS, support AI by aligning AI-Subdomains, Projects, EAMs, AI-Models (AIM), and other artefacts. In this article the focus is on DS, related DSPs, and their relations with other AI-Subdomains, services' architectures, interface-variables, adapted mixed-methods, and domain functions, EAMs, data-platforms… AI-Subdomains like (R)DS, ML, DL, DP, and other, are the fundaments of an efficient DMP that constitutes the basis of an enterprise Decision-Making System (DMS). As already mentioned, DSPs are used for validity-checking, GAPA, and optimization activities, and DMS activities. Before reading this article, the valuable reader can consult In-House Implement (IHI) Polymathic Transformation Framework (IHIPTF) related guides, and Projects fundamental works, like: 1) The IHIPTF Guide [2]; 2) The IHIPTF Glossary [3]; 3) A related syllabus [4]; 4) The AHMM4PROJECT [5]; and 5) The AI based Projects [6]. A Project is a set of Critical Success Areas (CSA) where a CSA corresponds to a DSP; and the first CSA is the Research and Development Project (RDP).

2. THE RDP

2.1. The Research Question

This article and RDP's Research Question (RQ) is: "Which RDS and related AI-subdomains features, and structural inter-action are needed to support AI-based Projects and Entity's sustainable evolution(s)?" The RDP's Polymathic Resources and Literature Review (PRLR) uses AI literature, IHIPTF's knowledge DataBase (DB), Conceptual Proof of Concepts (CPoC), and previous articles repository, and gives advantages to the authors' relevant works and professional consulting-projects, like:

- The Polymathic set of MMs or the AHMM4PROJECT(s) [5].
- Deep Learning Integration for Projects (DLI4P) in Projects [22].
- Machine Learning Integration for Projects (MLI4P) [7].
- The Business, Societal, and Enterprise Architecture Framework: An Artificial Intelligence, Data Sciences, and Big Data-Based Approach [8].
- An Applied Mathematical Model for Business Transformation and Enterprise Architecture: The Holistic Organizational Intelligence and Knowledge Management Pattern's Integration … [9].
- Enterprise Transformation Projects: The Polymathic Enterprise Architecture-Based Generic Learning Processes (PEAbGLP) [10].
- The Business Transformation and Enterprise Architecture Framework-The Applied Holistic Mathematical Model's Persistence Concept (AHMMPC) [12].
- An Applied Mathematical Model for Business Transformation and Enterprise Architecture-The Holistic Mathematical Model Integration (HMMI) [13].
- Using Applied Mathematical Models for Business Transformation [14].
- The Polymathic approach for Projects that use a Meta-Model [15].
- Applied Holistic Mathematical Models for Dynamic Systems (AHMM4DS) [16].
- … and others.

2.2. The Set of MMs for RDS

The Project, ICS, AIMs, EAMs, and other, are in fact sets of MMs and IDSs. The RDS uses the most relevant MMs which are [5]:

- MM for U4MP (MM4UP) which supports unbundling.
- MM for Factors (MM4FC) which structure and checks Factors.

- MM for GAPA (MM4GP) which supports GAPA's estimations.
- MM for PRWC (MM4PR) which supports PRWC's estimations.
- MM for Expectations and Constraints (PEC) (MM4PE) which structures PEC.
- MM for Polymathic Enterprise MtM (PEMtM) (MM4PM) which is base for PEMtM.
- MM for Methodologies and ICS (MM4MD) which supports and checks OOM/UM, Archimate, and other.
- MM for MDTCAS (MM4MT) which supports and checks MDTCAS.
- MM for APDs (MM4AD) which supports and checks APDs and its problem-types.
- MM for Intelligence (MM4IN) which supports and checks AI and decision-making.
- MM for AHMM() (MM4AH) which constructs the for a specific topic .
- And others.

2.3. The AMM4IDS

The RDS needs holistic organizational intelligence and knowledge management capabilities offered by the AHMM() [9]. Which restructures (or transforms) an Entity in the optimal manner, by applying AI/DS modules and DSPs. That needs a concept that is based on standards, mapping-mechanisms, and interoperability. DSPs respect standards, and methodologies to support AI based Projects. Transforming Entity's legacy ICS and DMPs into an AI based atomic service-oriented environment.

2.4. RDP's Hypothesis

This RDP uses hypotheses (or assumptions and even constraints) that are in fact linked to Project's CSAs and the RDS corresponds to the main CSA and RQ. RQ's real-scope and feasibility, and RDP's credibility, depend on the perceived hypotheses (and assumptions). Where RQ's main hypotheses depend on the following Project's transformational-activities (and phases) successful finalizations:

- Project's start, budget, contracting, goals, AI/RDS vision (and roadmap) developments were hammered and documented.
- There is the needed a sufficient level of political support and AI/DS skills/experiences.
- Disassembling strategies were implemented to offer AI/DS services and The IHI methodology is the Methodology, Domain, and Technology Common Artefacts Standard (MDTCAS) artefacts.
- The Entity and Project have successfully implemented MDTCAS, Factors Management System (FMS), Polymathic Rating and Weighting Concept (PRWC), HDT, and In-House Implement (IHI) Polymathic Transformation Framework (IHIPTF).
- The Project uses the DB Centric Concept (DBCC) (or DB first) for data operations [17]; and the DB Centric Implementations (DBCI) [18].
- The Entity privileges IHI solutions and AI-Subdomains' integration.
- The Project's team has the needed skills for AI/DS modelling and development.
- The Project (and Entity) has implemented the AHMM() or AHMM4IDS (and its MMs) fundaments and interactions [5].
- A hypothesis (assumption) corresponds to a CSA, which are selected using the PRLR.

2.5. The PRLR and FMS

This RDP localized an important research-gap that is due to: 1) There isn't anything similar to the RDS, AHMM4IDS, and MMs, and IHIPTF; 2) eXtremely High Failure Rates' (XHFR) and their possible identification; 3) There isn't a mixed method similar HDT that includes the Quantitative-Qualitative Research Mixed Model (QQRMM); 4) A concrete Weightings-rating (Wgt) based PRWC and FMS that are related to AI/DS, GAPA, MMs, ICS and IHIPTF; 5) The use of CSA Decision-Tables (CSA_DT) to qualify Project's CSAs; and 6) A structures approach to AI-subdomains and the DS' integration. As shown in Figure 1. GAPA is applied to all CSAs, but in this article only one GAPA/CSA_DT will be presented. The AHMM4IDS supports DSs and DSPs by inter-relating MMs for Factors (MM4FC), where Critical Success Factors (CSF), and Performance Indicators (KPI) are used for Project's and DSs basic-evaluations, and Optimization-Functions (OF). The AHMM() (and MMs) support DS' MetaModels (MtM) and a Polymathic Enterprise MtM (PEMtM). The PEMtM perform Entity's validity-checking and also uses the FMS and its underlying Factors that offer [19,20]:

- They can be used in Natural Programming Languages (NLP) scripts.
- The FMS incorporates CSAs, CSFs, KPI, and ICS VARiables (Var) (simply Factor).
- A CSA maps to a set of CSFs (and Project's resources), and a CSF is a set of KPIs.
- The Team manages and tunes the initial-sets of Factors.
- A KPI maps to a unique requirement and problem-type.
- CSFs are used for solving problem-types, in Decision Making System (DMS)/Knowledge Management System (KMS) (simply Intelligence), and other.
- FMS ensures that: 1) A CSA maps to an Entity APD (or a common functional-domain); 2) A CSF maps to a set of requirements (and directly linked problem-types); and 3) A KPI maps to a ICS' item-variable or Var.

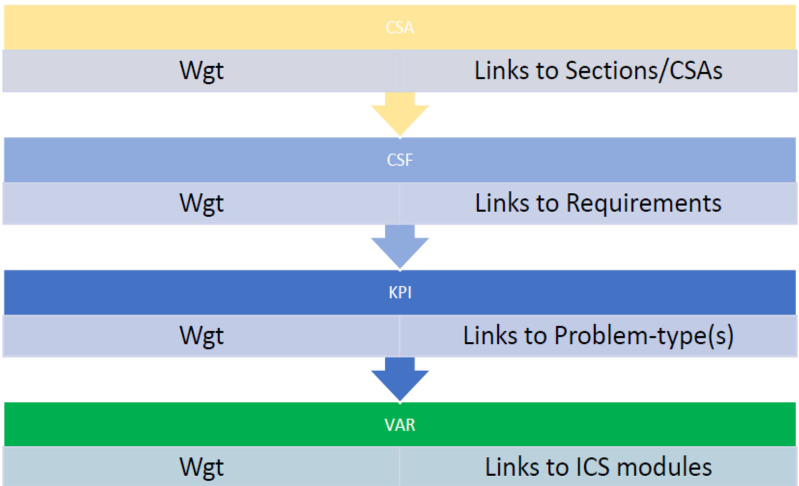

Figure 1. The evaluations for IHIPTF and AHMM() that process CSA_DTs.

2.6. The PRWC, CSA_DT's Evaluation, and GAPA

The PRWC interacts with the FMS and Intelligence to offer [21]:

- An Entity standard for CSA-DTs' evaluation method like the Decision-Making Notation (DMN).
- CSAs are evaluated using CSA_DTs, where a CSA corresponds to a Project topic.
- CSA_DTs deliver RDP's Phase's 1 evaluations which constitute Project's DT (Prj_DT).
- The AHMM4IDS and MM4PR are supported by the PEMtM.
- Is used to evaluate Project's integrity and used by DSPs.

- Uses the FMS that includes: Factors like VARs that are accessed by DSPs.
- GAPA evaluates Projects, DS' modules, and other components statuses; by using HDT based Intelligence to eliminate gap(s).
- The PEMtM enables GAPA's processing in all Project's phases which are synchronized by the Transformation Development Method (TDM).
- The TDM uses The Open Group's Architecture Framework (TOGAF) and its Architecture Development Method (ADM).
- GAPA uses Factors like CSFs which can be: 1) Project's resource; 2) Disassembling outcomes; 3) DSPs' evolutions; 4) PRWC's outcomes; 4) TDM's synchronization; 6) AI/DS' outcomes; and 7) Use of KPIs to link VARs to concrete components.

3. ICS, EAMS, DBS, AND OTHER

3.1. Introduction and Distributed ICS

The AI Generic and Basics Constructs (AI-GBC) is common layer for all AI-Subdomains, where an Entity and a Project can implement an IHI AI-GBC, which is the largest and critical part of AI-Subdomains' integration and/or implementation. Building a robust AI-GBC for transformed ICS (or modern applications) ensures scalability and reliability for AI-Subsystems. The AI-GBC: 1) Includes the hardware, software, data-access mechanisms, and networking components, which are essential for transforming, implementing, deploying, and managing Projects and AI-Subsystems; 2) It is the backbone of AI-Subsystems, that enables AI-Subdomains like, ML algorithms to process huge data-volumes and to generate data-insights; 3) Supports the integration of patterns like DSPs; 4) Avoids just using commercial products that cause a hairball; and 5) Offers a robust infra-structure and necessary resources for AI-Subdomains, so Entities can integrate complex algorithms like ML and BD for data-insights and data-driven DMPs. The different and extensive processing of AI-Subdomains requirements and AIMs, is strongly related to large IDS (used in launching complex computations). The use of various types of massive IDSs is a considerable challenge. Distributed ICS (DICS) and related computing-processing, need parallel processing-steps which enables AI-Modules to be scalable and supports the demands for massive data-driven environment [23,26].

3.2. DICS and High-Performance Computing (HPC) Systems

As shown in Fig. 2, the AI-GBC recommends Graphics Processing Unit (GPUs) and Tensor Processing Units (TPU) usage, [23,24,25] because:

- Graphics based processing technologies have enforced AI computing capabilities.
- GPUs offer new possibilities in complex domains like AI and were originally created/designed for graphics and video-rendering.
- But they are also suitable for accelerating AI-Subdomains computations, where their architecture is based on hundreds of cores which can handle thousands of threads simultaneously. That makes it exceptionally capable for parallel-processing requirements of ML algorithms.
- As it is designed for parallel processing/multi-threading, the GPU can used for various types of applications, whereas TPUs are Application-Specific Integrated Circuits (ASIC) used for ML and DS.
- TPUs are designed to support and accelerate the processing of TensorFlow's framework, which is mostly used for DL; and they are optimal for high-throughput used by matrix-operations that are dominant in Nural Networks (NN) processing-calculations, which offers efficient alternative to GPUs for specific AI-Subdomains.

- TPUs are AI-accelerators, which are optimal for large AIMs training, in a variety of use-cases (or APDs), like chatbots, code-generation, media content-generation, synthetic-speech, vision-services, recommendation engines, personalization-models…
- Entities use DICS (or Clouds) based TPUs for IDSs for real-time DA and DS tasks.
- Clouds' services and APIs enable Entities to adapt to AI-Requests, ensuring optimal-performances and robust Projects.

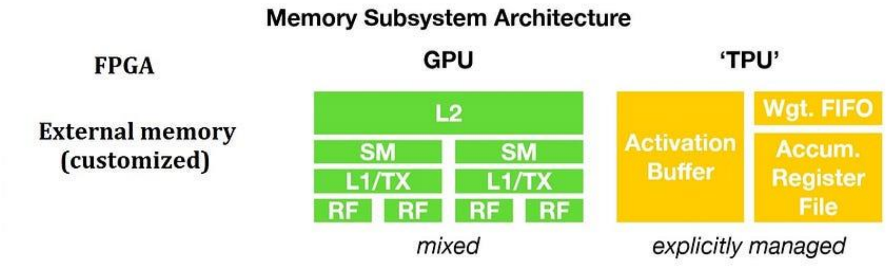

Figure 2. The ICS processing setup.

3.3. The Setup of High-Performance Data Storage Management Solutions

The AI-GBC recommends high-performances Data Storage Solutions, [23] based on:

- Distributed File Systems (DFS) like the Hadoop Distributed File System (HDFS) and Google Cloud Storage (GCS) are crucial for managing and coordinating large/huge IDSs across ICS' multiple nodes.
- HDFS splits (or breaks) large/huge files into smaller blocks, which are distributed and can be replicated across ICS-nodes, which support scalability, and high-throughput.
- It persists-stores data (or IDSs) near to computation-processing nodes, enhancing performances for various types of tasks like batch-processing.
- High-Speed Solid-State Drives (SSD) are important for data-intensive applications, like in AI-Subdomains, due to extremely fast read/write speeds.
- SSDs have outperformed legacy Hard Disk Drives (HDD) in speed, which is crucial vital for fast data-processing, application-launches, and real-time DS and Data-Analysis (DA) activities.

The AI-GBC recommends Data Management, based on [23]:

- Data-ingestion and preprocessing are important steps in the data-pipeline, which ensure that raw-data is transformed into a request data-format that is ready for AI-Subdomains like DA, DA, ML…
- Extract, Transform, Load (ETL) pipelines, that are shown in Fig. 3, are applied to manage data-flows from multiple data-sources.
- In the extraction-phase, data is gathered from different data-sources like DBs, APIs, and system-files. In this data-transformation phase, the data is formatted, including Quality of Data (QoD) topics like missing values, duplicates, and inconsistencies.
- The loading phase is responsible for storing the processed-data in a Data Warehouses (DW) or DB, to be accessible by DS.
- ETL pipelines ensure data-integrity and efficient DS operations.
- Data labelling tools are used for implementing annotated DataSets, which are important for supervised learning; the mentioned tools enable tagging and categorizing data, like images, text, or audio, providing la-bels for training ML models.

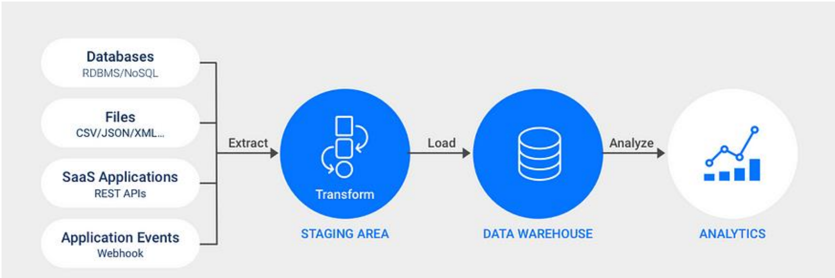

Figure 3. The ETL pipelines [23].

3.4. The DB Centric Strategy and Use of IDSs

In AI there are various streams and probably the most important ones are:

- A DBCC and DBCI are Data (or DB) oriented approach which is the most popular concept in AI-Subdomains, especially in APDs like business, marketing, logistics [17, 18]… Such an approach is known Quantitative methods and statistics.
- A heuristics approach (like the proposed HDT) uses an Actions-Research (AR) engine is and experiences oriented, which is adapted to PEAbGLPs.
- Enterprise oriented which a mixture, which is combination of the two previous approaches, and is represented by the AHMM4IDS, QQRMM, IHIPTF, and Entity AI Concept (EAIC); which need an AI generic and basic constructs strategy.

3.5. Design for AIMs and AI-Subdomains Architectures

AIMs and AI-Subdomains that define a specific AI-Architecture is considered in the context of the Entity's EA-Models and has the following characteristics [39]:

- EA or Unified modelling language can be used for these purposes.
- Based on the Project's requirements and problems, specific algorithms can be selected to design AI-Architecture integration and interactions.
- The selected algorithms can include rule-based learning, DL, and NLP.
- AI-Architecture affects performance in an important manner, so there is the need to tweak different AI-Platform configurations to find the most effective one.
- Various learning technics are very efficient for different APDs.
- NLP models like transformers would be better for managing complex contextual relationships.

4. AI-GBC Advanced Topics

4.1. The Setup and Use of IDSs

IDS is the basis of using data artefacts and IDSs' main characteristics/features, parts, components, or modules are [27]:

1. They enable AIMs training and learning.
2. They can have different and formats depending on the applications of AI-Subdomains, which can range from images, binary, text, complex sensor-io...
3. The QoD and quantity of IDSs are important CSFs in estimating AIMs.

4. QoD must ensure that used data is errors-free, otherwise IDSs will be full of biases, or irrelevant information; which will cause the AIM to deliver erroneous predictions.
5. QoD needs: 1) Data cleaning of inaccuracies and inconsistencies; 2) Data labelling and tagging with correct-labels for supervised learning; and 3) Data augmentation helps AIMs and EAMs to improve generalization.
6. Quantity of data improves predictions (and prediction-accuracies), because, more data/IDSs means better AIMs, but that can prove itself wrong… Data quantity has to correspond to the AIM's problem-space, which means that it should have enough variation and samples of different classes/outcomes needed for AIM's PEAbGLPs.
7. IDSs preparations are important for AI-Subdomains like ML includes: 1) Irrelevant-data identifying and discarding; 2) Duplicated data detection; 3) Noise data filtering; 4) Incorrect data-types correction; 5) Missing-values corrections; 6) Multi-collinearity improvements; 7) Outliers are managed; and 8) Unacceptable format are discarded.

4.2. The Use of AIMs

An AIM is a program (or logic) that autonomously supports specific business tasks, in an automated manner. Like HB, it learns, solves problems, and makes predictions. It does learn from experience like the HB, but it acquires knowledge from massive IDSs and applies mathematical-techniques and algorithms to derive insights. An example case, is an AIM that is used to compare pictures of tele-phones and PCs/laptops, using training on labelled images of both. To find differences, the AIM analyses the inputted images to detect patterns, like size, key-board, used building materials, and screen's design. When the AIM is highly trained, it can be used for decision making for new objects, as shown in Figure 4. AIMs can be used for different Apart from this image recognition task, you can apply AI models to different workflows; that includes Apart from this image recognition task, you can apply AI models to several workflows; that include NLP, anomaly-detection, predictive-modelling, and forecasting… These include natural language processing (NLP), anomaly detection, predictive modelling and forecasting, and robotics. to several workflows. These include natural language processing (NLP), anomaly detection, predictive modelling and forecasting, and robotics [39].

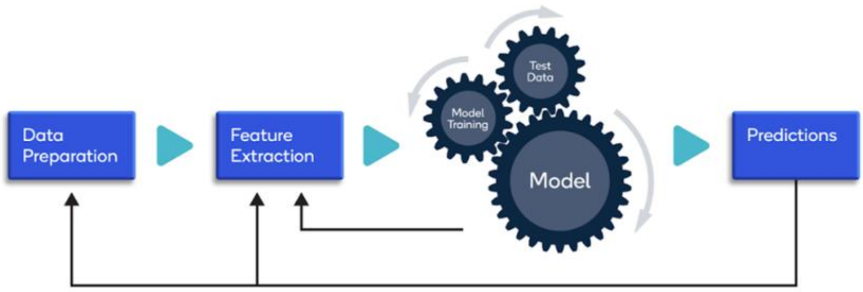

Figure 4. An AIM [39].

As shown in Figure 5, supporting an AIM needs [39]:

- Identifying the problem to be solved and defining goals to be achieved.
- Data preparation and gathering DataSets that reflect the workflows. Where data can be structured, unstructured, static, or streaming, and inconsistencies must be removed.
- Execute AI-Architecture tasks.
- Training, validation, and testing data splitting into three DataSets as follows:
- Training DataSet, which can be up to 70% of the total DataSet.

- Validation IDSs use 15% of remaining data for validation, and AIM's enhancements.
- Testing DataSet, reserves the final 15% to evaluate the AIM performances.
- AIM training uses backpropagation to incrementally tune its internal parameters; and requires important ICS-resources and efficient frameworks like PyTorch.
- Hyperparameter tuning of batch-size, learning-rate, and regularization methods keeps the balance between underfitting and overfitting.
- Model assessment by using validation DataSets, to evaluate the AIM's effectiveness. Various Factors (or Metrics) like accuracy, precision, recall, and F1-score will provide insights in AIM's performances.
- Testing and deployment use testing DataSets and the AIM has to to meet defined use cases; and if the results are satisfactorily, then deployment processes are initiated.
- Continuous evaluation and enhancements are supported by the applied AIMs to adapt to the transformation of data-patterns. Received reports helps in understanding AIM's performances and how to make needed adjustments to keep it relevant.

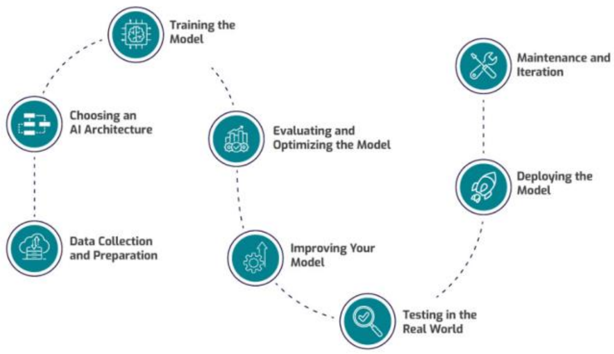

Figure 5. The creation of an AIM [39].

4.3. Needed Support

Supporting an AIM needs [27]:

- AIM's training process includes improve PEAbGLPs to offer predictions or decisions based on inputted IDS, that includes various steps.
- The selection of the Right-Model step, offers an algorithm that corresponds to the IDS and the problem to be solved.
- The Preparation step, splits the IDS into training, validation, and test sets of data.
- The Feeding step, inputs IDSs into the AIM in batches, during the training-phase.
- The Backpropagation step, adjusts AIMs by PRWC (weightings) that is based on predictions' errors.
- Validation steps, use validation IDSs to tune AIMs' hyperparameters.
- Testing steps, evaluate AIMs' performances on unseen IDSs to ensure generalization.
- AIMs' training faces many challenges, and overfitting, underfitting, and ensuring AIM's interpretability are barriers that AI-Engineers face [27]:
- Overfitting is the case AIM learns too much, from the training IDS too well, which results in including noise and outliers, and causes poor performances on new IDSs.
- Underfitting is the case when the AIM is very simple to capture the underlying-trend(s) in the IDSs.
- Interpretability refers to the ability to understand AIMs' made decisions, which is determinant.

To overcome these challenges, AIMs can use best practices [27]:

- Regularization: Techniques like dropout or L1/L2 regularization can prevent overfitting.
- Cross-Validation: Using different parts of the data to train and validate the model helps in assessing its performance.
- Feature Engineering: Selecting and transforming the right features can improve the model’s learning ability.
- Model Explainability: Tools and techniques that help explain the mod-el’s decisions can build trust and aid in debugging.

4.4. The Setup and Use of Services/API

AI-Subdomains use specialized services, APIs, and AI As A Service (AIaaS) which cover a wide-spectrum of required AI-Functions, like understanding hu-man-language(s), recognizing objects in images and videos, learning from data, understanding speech, analysing sentiments, suggesting personalized advices-recommendations, and other. The AIaaS offers [11,28,29,30]:

- The AIaaS platform ecosystem is optimal for fast commoditization in cloud-services. Cloud-vendors offer a range of standardized, pre-configured AI-Services. Like AI-services that supports the deployment of chatbots; financial-services, fraud-detection…
- Services-Oriented Architecture (SOA) and MSA which accompanied the decline of legacy monolithic ICS architectures. The emergence of SOA and MSA enable the implementation of cloud-services, like AIaaS.
- Legacy monolithic ICS architectures, an application is a set of unit, whereas, SOA and MSA support applications composed of atomic Blocks, which supports the implementation of AIaaS-Functions.
- Commoditization of AI-Functions in AIaaS offers: 1) Optimizing ICS-infrastructure costs; 2) ‘pay as you go’ model; 3) Reduction of special-ists’ costs; 4) ‘out of the box’ pre-trained AI models can be used; and 5) Integration with other DICSs.
- Supports various APDs like image-recognition as shown in Figure xxxx that enables a user to identify objects by taking a photo.
- There are many AI-Frameworks that deliver AI-Services and for various subdomains like Machine Learning as a Service (MLaaS).

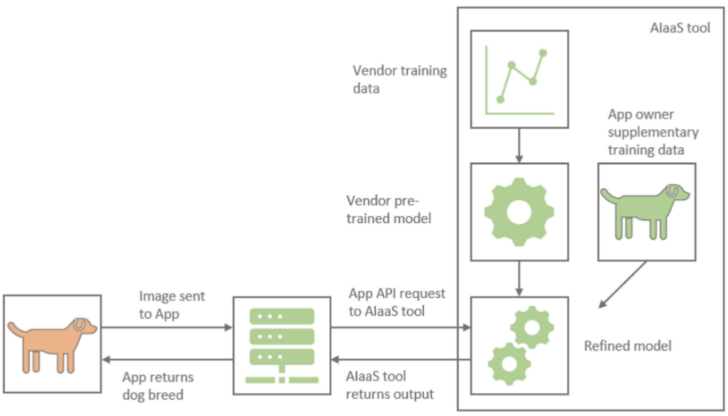

Figure 6. An AIaaS based application that recognizes objects [28].

5. LEARNING APPROACH

5.1. Generic Learning Approaches

There is a strong interaction between Projects, AI-Subdomains implementation and Entity PEAbGLPs (ELP), and therefore the Entity has to implement a Polymathic Enterprise Architecture-Based Generic Learning Processes (PEAbGLP). The PEAbGLP manages all ELPs' that includes ML, DL, DS, traditional-legacy LPs and other. The PEAbGLP proposes an IHI concept for a generic and transcendent EAIC. EAIC generic approach means that it supports and interfaces with all AI-Subdomains. The EAIC uses a Polymathic transformation framework that is specialized in Projects. Projects have XHFRs, and added to this complexity, AI-Subdomains implementations and related products can force a siloed-integration implementation approaches, which are the main reason for XHFRs. The EAIC ensures Entity's sustainability, just-in-time decision-making, and operational efficiencies [10].

5.2. AI-Learning Subdomains

There various types of AI-LPs types: 1) Artificial Narrow Intelligence (ANI); 2) Artificial General Intelligence (AGI), and Artificial Super Intelligence (ASI). Where AI-LPs can be classified to its ability to function like the HB; and the main differences are [8,31]:

- ANI is used for simple AI-Subdomains and applies algorithms implemented by developers, which is equivalent to the implementation of a predefined-task, and it does not include autonomous-reasoning; like the AI for Voice (AI4Voice) commands, offer users appropriate responses, used in APDs such as fraud-detection, machine-translation…
- AGI is the case where AI can reason and offer decisions like the Human Brain (HB), where the level of precision is still to be defined. AI based problem-solving which HBs cannot solve like scientific research or mathematical-reasoning.
- ASI is in initial phases, and it is supposed to mimic HB's intelligence but for the moment AI is much more an enhanced statistics field.

In AI related ELPs and more specifically the author's PEAbGLP supports various AI-Subdomains by offering like [7,31]:

- AI-Machines are today are the base of ELPs, where machines which DICS based, interpret, process, and analyse IDSs to diagnose or solve problems. It is mainly a Quantitative approach in which massive IDSs are inputted to DICSs to be processed.
- ML has different types of LPs, which are to: 1) Supervised; 2) Unsupervised; and 3) Reinforcement LPs.
- Supervised ML (SML) LPs are based on labelled data-samples; where machines executed specialized tasks on samples that have been labelled, such samples are known as Training-Data for AI (TD4AI), like for example in on image-recognition.
- SML's image-recognition the main concept is based on idea that a machine determines if the image belongs to a specific category and object. Once the machine learned to recognize object's image from TD4AIs, then it will be fed with new images to tune the knowledge related to the object; and at the end used to make predictions.
- Unsupervised ML (UML) LPs are based learning by comparing objects. UML is different from SML, because the machine will not use labelled samples to execute tasks like analysis. By using comparison, that depend on the characteristics of the objects' different elements, like in the case of image-segmentation. Image-segmentation's goal is to

classify-group similar image-objects depending on their attributes, without having pre-defined solutions, like online products' search.

- Reinforcement ML (RML) LPs are based learning from concrete user-experiences and interactions with the concerned environment(s) and APDs, where LPs correspond to HBs' way of reasoning. In this case the machine explores DMS options and offers feasible recommendations (or rewards) or penalties, depending on the selected Factors. And the objective is that the machine improves and persists LPs with the most rewards and therefore, improves its future DMS performances.
- DL LPs are based on NN, which is a subcategory of ML that in-turn uses Artificial NNs (ANN) to solve-process more complex IDSs and deeper (more aggregated) AIMs. DL efficiency exceeds ML, because it is capable of analysing more complex DataSets using deeper with sophisticated NNs based processing. Like in the case of an image-recognition the DL LP analyses the shapes.
- And afterwards iteratively, it tries to understand object's constructions from various viewpoints. So, DL LPs, develop a Virtual NN (VNN) from large IDSs to offer solutions.
- The VNN explores a number of layers, and data becomes hard to analyse.
- NLP based AI (NLP) is founded on scientific methods related to learning from NLP to implement machines capable of communicating, and developing businesses and their Business Processes (BP).
- Other Subdomains based AI (OS) like robotics, expert systems, fuzzy logic (and other) LPs include: 1) Robotics that uses AI on robots; 2) Expert systems that offers HB expertise to develop a well-defined reasoning-thinking capacity; 3) Fuzzy logic-based AI for machines/computers to understand topics, problems, and solution which are not just binary 0 or 1 ("true" or "false"). They are applied for machines to offer decisions that are close to HB's decisions made in complex situations, like in medical systems.
- ML LPs support the evolution of ML oriented Projects. ML supports DMS, AI-Subdomains (like DS, BD…).
- AI-Subdomains like DS, BD, and other need a Project and Entity strategic roadmaps which defined AI based transformation processes to implement an Entity DMS where the role of DS and DS and the needed AI-Cartography and DSPs [8,32].

6. AI-CARTOGRAPHY AND DSPS

6.1. AI-Cartography

AI-Cartography needs to define the real relations and interactions between various DICS and AI-Subdomains, especially AI-Components, ML, DS, and DL. The mentioned AI-Subdomains are inter-connected to analyse large IDSs. AI-Components, ML, DS, and DL are related, but they are applied for different goals and have different methods of practical operation [6,8,32]:

- AI-Subdomains need a generic concept to support the DMS.
- A generic concept thar needs a Project to transform the traditional business environment into an automated one.
- AI-based business environment control and monitoring.
- The role of BD, DS, and data-modelling techniques are essential for the DMS.
- ELPs for DL and ML are based action research approach can unify all AI-Subdomains.
- As already mentioned, DS deals with IDSs using environments, and technics to extract hidden-patterns, meaningful-insights, and to offer optimal decisions.
- DS automated processes includes: gathering, organizing, and analysing IDSs. And can be considered a Polymathic (interdisciplinary approach) which merges different fields of DICS, scientific processes (and methods), and statistics.

- For mining BD (associated with DS), DS uses various techniques, tools, and algorithms.
- DS uses mathematical statistics and ML to handle voluminous IDSs.
- In ML, statistical-methods are used by DCIS to enhance their LPs, without being implemented by software packages. And it focuses on making algorithms learn from the imputed-data, collected-insights, and persisted predictions/decisions on data.

As already mentioned, ML have 3 LP types: Supervised; Unsupervised; and reinforced; and can be related by [32]:

- The Polymathic RDS can put together capabilities from various APDs and AI-Subdomains ML, quantitative statistics, and visualization; to deliver APD-valuable insights from voluminous IDSs, supporting robust DMS' processes in various areas, like DICS/technology, scientific-research, and usual business oriented APDs.
- ML is a subset of both DS, and AI, which utilizes algorithms and statistical models to process IDSs, that in turn support ELPs to be enriched without hard-coding.
- AI is a wider concept, and it focuses on creating DCISs which execute operations which are commonly done by the HB which has intelligence, reasoning, learning, and problem-solving capabilities.
- DS is the foundation for ML and AI that incorporate IDSs for applied AIMs, and to learn from them. It integrates algorithms from ML and includes concepts from legacy traditional APDs' expertise, statistics, and mathematics to implement solutions.
- AI is a valuable resource for DS related fields, because it generates data-insights. The main difference with AI, lies in DS's comprehensive approach to data-collection, preparation, and analysis, transcending only algorithmic or statistical tools-facets. Where ML and AI focuses on algorithms' implementation in Projects.
- An ML case is predictive-maintenance, which is used to predict future results (or outcomes) based on historical-data. Predictive-maintenance includes analysing IDSs from sensors to predict equipment's possible failure(s). ML algorithms are applied to find or identify patterns in IDSs that become problems, and enables maintenance engineers to automate corrective actions before that problems occur.
- NLP is the combination of the mentioned concepts and it involves analysing human language(s) to extract insights and their meaning. DS is used to collect and prepare DataSets, and ML is used to develop the needed algorithms, and AI is used to support the NLP subsystem.

6.2. Cases of Interaction

Cases in which AI, DS, and ML interact can be [32]:

- The Recommendation systems, is an AI case, where it represents an algorithm that delivers personalized recommendations based on IDSs. DS participates in the collection and analysis of user IDSs, while ML is used to implement algorithm(s) that supports the mentioned recommendation system. A concrete case of the recommendation system powered by AI, is Amazon's personalized product recommendation-algorithm. The mentioned system applies ML techniques and algorithms to analyse user-behaviours' information, including historical-transactions, product-ratings, and browsing history, to finally provide personalized-recommendations/suggestions to end-clients.
- The Fraud detection system uses DS and ML to analyse large data amounts and to identify (hidden) patterns and errors/anomalies that can include fraudulent actions. ML algorithms identify such patterns and anomalies, while DS collects and prepares IDSs for DA. Like in the case of PayPal which utilizes a ML mixed-methods and DS to analyse

huge quantities of transactions' information to find fraudulent-actions. Such a concept identifies patterns and errors/anomalies in IDSs that could indicate fraudulent-actions, like atypical expenditure habits or suspicious Internet Protocol (IP) locations.

- The NLP based chatbots which are designed to simulate conversations with people (clients). AI-Subdomains support chatbots, like ChatGPT, which uses ML algorithms, and NLP techniques, to understand Natural Language Queries (NLQ), to provide personalized responses. The implementation of chatbots, forces AI-Developers to start with collecting and preparing large training IDSs. These IDSs are used to train ML models that can analyse and interpret Natural Language Text (NLT). When the models were trained, then they are integrated into chatbot(s) to provide intelligent and precise answers or solutions.
- General Electric's (GE) predictive maintenance platform which uses ML, which collects sensor IDSs from its equipment and analyses them to predict their statuses. The mentioned platform uses advanced ML's algorithms to detect patterns in sensors' IDSs. It is designed to learn from historical-data, to improve predictions' accuracy.
- The mentioned AI-Subdomains overlap and there are redundancies, which each Subdomain has a unique role in solving problems. The interdependence of Subdomains DS, ML, and AI leads to the use of adapted environments. Where Machine Learning Algorithms (MLA) play a central role.

6.3. Machine Learning Algorithms

There are different types of MLAs [33,40,41,42]:

- SML Algorithms (SMLA), consists of a target (or outcome) variable (also known as a dependent variable), which is used to predicted from a given set of predictors (independent variables). SMLAs for classification and regression include generating a function/module that maps input-data to corresponding outputs. Training is applied until the AIM achieves the defined or sufficient-enough level of precision. The most popular and used SMLAs are: 1) K-Nearest Neighbours (KNN) is a SMLA classifier, which uses proximity to create classifications (or predictions) on the grouping of individual data-points. It is the common classification/regression classifiers [34]. 2) Regression uses a set of mathematical-methods to predict a continuous outcome (y) based on the value of (1 to n predictor) variables (x). Linear regression is the most common regression analysis method because it is simple to use in predicting and forecasting activities [35]. 3) Logistic regression is a statistical algorithm applied to binary-classifications, used to predict results (or outcomes) that comes from 1 of 2 offered possibilities (yes or no, True or False, Spam or Not spam…). It is used to predict continuous-values for classification problems [36]. 4) Decision Trees (DT) are a non-parametric SML used for classification and regression to create an AIM which predicts the value of the defined target-variable by learning from Decision-Rules (DR) deducted from DataSets. And a DT Classifier (DTC) is a ML prediction mechanism which generates rules, like the "IF Rule THEN…" clause [37,45]. 5) Random Forest is a popular, dynamic and simple ML algorithm that offers, without exaggerated parameter configuration-tuning, optimal results. It is used for classification and regression functionalities [38]. 6) Linear Discriminant Analysis (LDA) is an algorithm for classification and dimensionality reduction; it is used to IDSs that contain large number of features (like image-data). The reduction of the number of useful features is essential to achieve a clear classification process [46]. 7) And other…. Each SMLA uses different types of IDSs, problem-type, or requirement.
- UML Algorithms (UMLA) acts on unstructured and unlabelled data, in which there are no target (or outcome) variable for predicting purposes. UMLAs for clustering and data-

mining identify hidden-patterns (or possible structures) in data. Using the mentioned patterns, data-points are grouped by using "similar-characteristics", that auto-generates a function for mapping input-data to clusters (or groups). This iterative process continues until the applied AIM identifies meaningful patterns in IDSs. The most popular SMLAs are:1) K-Means clustering used for clustering-problems used in ML or DS contexts, where it groups unlabelled IDS into different clusters. K defines the number of pre-defined clusters, created and as if K=2, creates 2 clusters, and for K=3, creates 3 clusters…2) Hierarchical Clustering, used to group the unlabelled datasets into a cluster and also known as Hierarchical Cluster Analysis (HCA). It develops the tree-hierarchy of clusters (a dendrogram). Applies two approaches: a) Agglomerative is a bottom-up concept, which starts with defining data-points (as single clusters and merging them until one cluster is left); and b) Divisive is the reverse of the agglomerative algorithm because it is a top-down concept. 3) Principal Component Analysis (PCA), it is a statistical method that transforms observations of correlated-features into s set of linearly uncorrelated features, with the support of orthogonal-transformation process. The new transformed-features are PCAs used for exploratory data analysis and predictive modelling. It is a technique to draw strong patterns from the given dataset by reducing the variances. Each of these UMLAs serves different types of data/IDSs and problem-type, requirements, making these algorithms widely applicable across various APDs, AI-Subdomains, and other fields, such as customer-segmentation, anomaly-detection, and pattern-recognition.

- RML Algorithms (RMLA), consists of reinforcement ELPs' algorithms, where the DICS supports continuous-training (by trial & error) to offer recommendations and/or decisions. ELPs learn from past-experience and (try) to capture optimal knowledge used to offer precise decisions. RMLA's are: 1) Markov Decision Process (MDP), is a stochastic dynamic process, that originates from Operations Research (OR) and is used to model interactions between ELP's and the Entity's environment, DICS, BPMs and other. MDPs are divided into two categories: a- Value-based; and b- Policy gradient-based. In this framework, the interaction is characterized by states, actions, and rewards. 2) Value-based algorithms learn from state's value(s) of the Entity's environment (and related artefacts), where the value is given by the expected-rewards to finalize the task. 3) Q-Learning is model-free, off-policy algorithm, which focuses on providing recommended action(s) to take and in specific APD's contexts to offer optimal solutions. It uses Q-Tables in which possible solutions are persisted (for different state-action pairs in the environment). It also contains Q-Values that are updated after actions' execution. 4) Deep Q-Networks (DQN), are deep Q-Networks, which operate like Q-Learning algorithm, but DQN is based on NNs. 5) State-Action-Reward-State-Action (SARSA) is on-policy algorithm that uses current-actions from the current-policy used to learn and from where the values are deduced. 6) Policy-Based algorithms update the policy to optimize the reward/solution path(s); and there are different policies gradient-based algorithms.

7. COMPLETE PROPOSITION AND PROTOTYPE-THE CPOC

7.1. Preparing DICS' Platform

The CPoC's first step is to prepare the optimal RDS platform that includes:

- An IHI Cloud based DICS.
- DICS that offers HPC capabilities for enforced AI-Computing and accelerating AI-Subdomains processing, to support parallel-processing for ML algorithms.

- Support for AI-Accelerators, which are optimal for AIMs' training …
- HPC support for DBs based on DFSs that in turn enable scalability, fault-tolerance…
- Use DB centric concepts and design for AIMs and AI-Subdomains.
- All RQ's hypotheses are fulfilled and the Entity is transformation ready.

7.2. Preparing IDSs

The CPoC is a concept that is based on patterns that support IDSs for this undertaking and the CPoC is mainly based on [44,45]:

- The Iris Flower IDS (or Fisher's Iris IDS, IFIDS), is a multivariate IDS which was used for multiple-measurements in taxonomy related problem-types.
- It uses MLA's LDA to process IFIDSs and to quantify the morphologic variation of Iris flowers (of related "n" species).
- The IFIDS consists of 50 samples for each of 3 species of Iris (Iris setosa, Iris virginica and Iris versicolor), as shown in Figure 7.
- The selected "m" or 4 features were used and were measured for each data-sample: Length and width of sepals and petals, in centimetres.
- The combination of 4 features used by LDA-Model compares the species.
- These IFIDS can be used in AI-Subdomains basic constructs.

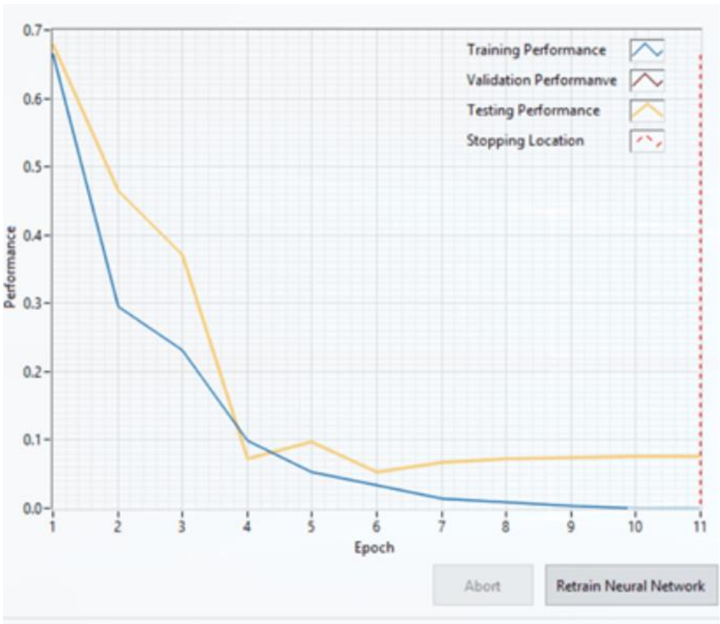

Figure 7. Scatterplot of the IFDS [45].

7.3. Basic Constructs

This prototype's main characteristics are [45]:

- Uses a DT and Working DTC (WDTC) are ML-Prediction system that generates rules.
- Uses an ML-Library which is a customized library for DT, which carries out the following tasks: 1) Implementing functions which split training-data into small DataSets depending on their disorder; and 2) Implementing source-code that calls/uses splitting-functions to create a DT from IFDS and processes/computes a predicted-class.

- Setting-up the CPoC with n=50 data items that are a subset of the data-store.
- The objective is to predict object-types (tagged with 0, 1 or 2).

7.4. The Execution-Setup

This PoC's setup has the following steps [45]:

- There isn't a standard format for WDTC and the Project must define its IHI version as shown in Figure 8 below.
- Each WDTC node contains 6 values defined in a Node class:
  1) The List<int> is the nodes' collection.
  2) The root-node's nodeID has the value: "0". The source-rows are the rows of the entire 30-item dataset with the values: [0, 1, 2. . . 50].
  3) The splitCol and splitVal attributes determine where the source-rows in the node are split in 2 subsets (having small average Gini-impurity). Small-impurity reflects a higher-level of homogeneity that is optimal.
  4) The selected splitCol is set to "2" and splitVal to "3.3", that implies that rows with column [2] has the values strictly-less than "3.3", are shifted (or assigned) to the left-child (or side) of the root-node, and the same with values greater than or equal to "3.3" are shifted/assigned to the right child of the root-node.
  5) The classCounts is an attribute collection that has the numbers of classes related with source-rows in nodes. Root-node's classCounts contains numbers of every class that is associated with source-rows. As all (50) rows are contained in the root-node, and 10 are left of each of the three classes, the classCounts collection that contains [10, 10, 10…].
  6) The predictedClass attribute is used for the prediction which is associated with the actual/current node.
  7) root-node's predictedClass is a class that maps/corresponds to classCounts' highest value. As all 3 class-counts is equal, implies that the predicted class is a tie between classes 0, 1, and 2. In cases of ties, PoC's WDTC randomly selects the 1st of the tied classes, which implies that the predicted class is "0".

```
public class Node
{
  public int nodeID;
  public List<int> rows;
  public int splitCol;
  public double splitVal;
  public int[] classCounts;
  public int predictedClass;
}
```

Figure 8. The class Node structure.

7.5. The Execution-Conditions

This PoC's execution conditions, and constraints [45]:

- The use of splitting (and disorder) scripts creates WDTC.
- The WDTC is ML based and predicts and then generates rules like IF Budget < 10000.0 AND Project_Status >= 2.0 THEN RiskFactors = 10 (which is high).

- The WDTC is ML must be customized and standard-library for DTs offer a huge set of complex functions. Therefore, it is better to use an IHI DWTC.
- For this PoC 30 (training/reference data items) to implement the IHI WDTC which has 7 nodes; which enables 100% accuracy.
- The PoC proves by predicting the class/species of a new (unkown iris-flower with sepal and petal values of (6.0, 2.0, 3.0, 4.0).
- The WDTC predicts that the class is 0, corresponding to the rule: IF (column 2 < 3.1) AND (column 0 >= 5.1) THEN class = 0.

7.6. The Execution-Prediction Process

This PoC's prediction process [45]:

- In the case of external tools or libraries WDTC, and specifically the environments that apply recursions, makes it complex and difficult to define the set of rules that offer predictions.
- PoC's WDTC ignores this complex topic and bypasses using sets of rules (as string-collections) in prediction processes.
- Sets of rules for WDTC's root-node is: IF (any value in all column of the item to be predicted, THEN it is anything.
- The PoC labels this condition using the snippet-string: IF (*) THEN […].
- In some simple cases the root-node is sufficient.
- The prediction_accuracy of a minimal WDTC is the comparison-etalon value; like for a collection of 30 items includes 22 class_1 items, 4 class_0 items, and 4 class_2 items, which means that root-node's class-counts is [4, 22, 4].
- The WDTC predicts class_1 for all items and related prediction_accuracy is 22 / 30 = 0.73 (rounded). The result that corresponds to the prediction_accuracy of a root-node is achieved by simple guessing.
- Therefor a credible WDTC offers prediction_accuracies that is better (>) than its minimal root-node version.

8. CONCLUSIONS

DSPs support reasoning-engines for Projects which are complex. The RDS aligns AI-Subdomains, Project components, EAMs, AIMs, and other artefacts. In this article the focus is on DS, other AI-Subdomains, services' architectures, EAMs, DICS-platforms… AI-Subdomains like (R)DS, ML, DL, DP, and other, are the fundaments of a DMS. This articles main RDS conclusions are:

- The RDS takes a transformative enterprise-wide view and not just DS' or AI.
- IDSs' contents and structures are combined with AI-Subdomains and DICS artefacts to deliver DSPs to be used by a DMS.
- IDSs can be interrelated with mixed-research methods.
- An has to build a performant IHI DICS Platform to support AI-Computing and ML algorithms, which are optimal for AIMs' building and training.
- Apply a DB, AIaaS, and IDS centric concepts.
- Use ELPs that includes ML, DL, DS, traditional-legacy LPs that results in implementing a transcendent EAIC.
- EAIC generic approach means that it supports and interfaces with all AI-Subdomains.

- An AI-Cartography defines the relations between various DICS and AI-Subdomains, AI-Components…
- An RDS supports the interaction of AI, DS, and ML components.

ACKNOWLEDGEMENTS AND FUTURE WORKS

This article is a part of a series of research works related to AI, Projects, and EAMs. It is an intersection of the mentioned fields that are interconnected using a Polymathic methodology and framework.

The Author will in the future work on finding common AI algirythm that can halp AI based transformations.

REFERENCES

[1] Trad, A. Enterprise Transformation Projects-The Applied Polymathical/Holistic Mathematical Model for Enterprise's Business Process Modelling (AHMM4EBPM). Journal: WSEAS. (2024).

[2] Trad, A. The In-House Polymathic Transformation Framework (IHIPTF)-The Guide. IBISTM. France. EU. (2024). https://www.ibistm.org/docs/PRWC.pdf

[3] Trad, A. The In-House Polymathic Transformation Framework (IHIPTF)-The Glossary. IBISTM. France. EU. (2024). https://www.ibistm.org/docs/Glossary.pdf

[4] Trad, A. The In-House Polymathic Transformation Framework (IHIPTF)-The Syllabus. IBISTM. France. EU. (2024). https://www.ibistm.org/docs/Syllabus.pdf

[5] Trad, A. The Applied Polymathical/Holistic Mathematical Model for Enterprise Transformation Projects (AHMM4PROJECT). IBISTM. France. EU. (2024). https://www.ibistm.org/docs/PAHMMforProjects.pdf

[6] Trad, A. AI based Transformation Projects. IBISTM. (2024).

[7] Trad, A. Business Transformation and Enterprise Architecture Projects: Machine Learning Integration for Projects (MLI4P). Unmanned Aerial Vehicles and Multidisciplinary Applications Using AI Techniques. Pages: 188-240. IGI Global. USA. (2022).

[8] Trad, A. The Business, Societal, and Enterprise Architecture Framework: An Artificial Intelligence, Data Sciences, and Big Data-Based Approach. Handbook of Research on Applied Data Science and Artificial Intelligence in Business and Industry. Pages: 447-490. IGI Global. USA. (2021).

[9] Trad, A. An Applied Mathematical Model for Business Transformation and Enterprise Architecture: The Holistic Organizational Intelligence and Knowledge Management Pattern's Integration … Journal: Volume 11, Issue 1, Pages 1-25. Scholar articles, International Journal of Organizational and Collective Intelligence (IJOCI). IGI Global. USA. (2021).

[10] Trad, A. Enterprise Transformation Projects: The Polymathic Enterprise Architecture-Based Generic Learning Processes (PEAbGLP). Machine Learning and Data Science Techniques for Effective Government Service. Pages: 29-66. IGI Global. USA. (2024).

[11] Primo AI (2024). Platforms: AI/Machine Learning as a Service (AIaaS/MLaaS). Primo AI. https://primo.ai/index.php?title=Platforms:_AI/Machine_Learning_as_a_Service_(AIaaS/MLaaS

[12] Trad, A. The Business Transformation and Enterprise Architecture Framework-The Applied Holistic Mathematical Model's Persistence Concept (AHMMPC). WSEAS. (2019).

[13] Trad, A. & Kalpić, D. An Applied Mathematical Model for Business Transformation and Enterprise Architecture-The Holistic Mathematical Model Integration (HMMI). Journal: IGI Book. (2019).

[14] Trad, A., & Kalpić, D. Using Applied Mathematical Models for Business Transformation. IGI Complete Author Book. IGI Global. USA. (2020).

[15] Trad, A. POLYMATHIC APPROACH FOR ENTERPRISE TRANSFORMATION PROJECTS-IMPLEMENTING A BUSINESS META-MODEL THAT RESPECTS ENVIRONMENTAL SUSTAINABILITY (PETP-IBMMES). SCF. Journal Article. (2024).

[16] Trad, A. Applied Holistic Mathematical Models for Dynamic Systems (AHMM4DS). Journal: International Journal of Cyber-Physical Systems (IJCPS). Volume 3. Issue 1, Pages 1-24. IGI Global. USA. (2021).

[17] Trad, A. Business Transformation and Innovation Projects-A DataBase Centric Concept (DBCC). IGI Global. USA. (2024).

[18] Trad, A. Business Transformation and Innovation Projects-A DataBase Centric Implementations (DBCI). IGI Global. USA. (2024).

[19] Peterson, S. Why it Worked: Critical Success Factors of a Financial Reform Project in Africa. Faculty Research Working Paper Series. Harvard Kennedy School. (2011).

[20] Putri, N., & Yusof, S.M. Critical success factors for implementing quality engineering tools and techniques in malaysian's and indonesian's automotive industries: An Exploratory Study. Journal Proceedings of the International MultiConference of Engineers and Computer Scientists. Volume 2. Pages 18-20. (2009).

[21] Trad, A. Business, Economic, and Common Transformation Projects-The Polymathic Ratings and Weightings Concept for AI (PRWC4AI). IGI-Global. USA. (2024).

[22] Trad, A. Architecture Projects: Deep Learning Integration for Projects (DLI4P). Handbook of Research on Advancing Cybersecurity for Digital Transformation. Pages: 288-331. IGI Global. USA. (2021).

[23] Namala, Ph. Building Robust AI Infrastructure for Modern Applications-Ensuring Scalability and Reliability in AI Systems. Medium. (2024). https://phaneendrakn.medium.com/building-robust-ai-infrastructure-for-modern-applications-88b992a53c30

[24] Google. Tensor Processing Units (TPU). Google (2024). https://cloud.google.com/tpu?hl=en

[25] Intel. What is a GPU? graphics processing units defined. Intel. Intel. (2024)

[26] Storage, P. Parallel vs. distributed computing: An overview. Pure Storage (2022). Blog. https://blog.purestorage.com/purely-informational/parallel-vs-distributed-computing-an-overview/

[27] Cluster Protocol. Understanding Datasets and AI Model Training. Open-AI for all. Medium. (2024). https://medium.com/@clusterprotocol.io/understanding-datasets-and-ai-model-training-78cb9538f6bc

[28] Hackney, H. AI As A Service – A Simple Introductory Example. (2023). https://www.architectureandgovernance.com/artificial-intelligence/ai-as-a-service-a-simple-introductory-example/

[29] Amazon. Machine Learning on AWS-Deploy high-quality AI models as APIs'. Amazon. (2024a).

[30] Red Hat. 6 overlooked facts of microservices. Red Hat. (2021).

[31] Galeon. Types of Artificial Intelligence & learning methods: what you need to know. Galeon. (2024). https://www.galeon.care/en/types-of-artificial-intelligence-learning-methods-what-you-need-to-know/#:~:text=There%20are%20several%20branches%20of,unsupervised%20learning%20and%20reinforcement%20learning.

[32] Magnimind Academy. Connection Between Data science, Machine learning (ML), and Artificial Intelligence (AI). Magnimind Academy. (2023). https://magnimindacademy.com/blog/connection-between-data-science-machine-learning-ml-and-artificial-intelligence-ai/

[33] Analytics Vidhya. Machine Learning Algorithms. (2024). https://www.analyticsvidhya.com/blog/2017/09/common-machine-learning-algorithms/

[34] Wikipedia. k-nearest neighbors algorithm. (2024). https://en.wikipedia.org/wiki/K-nearest_neighbors_algorithm

[35] Kurama, V. Regression in Machine Learning: Definition and Examples of Different Models. The ultimate goal of the regression algorithm is to plot a best-fit line or a curve between the data. (2024). https://builtin.com/data-science/regression-machine-learning

[36] Franco, F. Logistic Regression Algorithm. (2023). https://medium.com/@francescofranco_39234/logistic-regression-algorithm-6451c7928375

[37] Scikit Learn. Decision Trees. (2024). https://scikit-learn.org/1.5/modules/tree.html

[38] Donges, N. Random Forest: A Complete Guide for Machine Learning. All you need to know about the random forest model in machine learning. (2024). https://builtin.com/data-science/random-forest-algorithm

[39] Airbyte (2024). Types of AI Models: Your Ultimate Guide. https://airbyte.com/data-engineering-resources/types-of-ai-models

[40] Javatpoint. K-Means Clustering Algorithm. (2024). https://www.javatpoint.com/k-means-clustering-algorithm-in-machine-learning

[41] Wikipedia. Markov Decision Process. (2024). https://en.wikipedia.org/wiki/Markov_decision_process

[42] Dasgupta, R. Reinforcement Learning: AI Algorithms, Types & Examples. Data Science & AI. (2023). https://www.opit.com/magazine/reinforcement-learning-2/

[43] Wikipedia. Iris flower data set. Wikipedia. (2024). https://en.wikipedia.org/wiki/Iris_flower_data_set

[44] ANS Center. ANNHUB: Making a Machine Learning Model to Classify Iris Flowers. ANS Center. (2023). https://www.anscenter.com.au/post/making-a-machine-learning-model-to-classify-iris-flowers

[45] McCaffrey, J. How to Create a Machine Learning Decision Tree Classifier Using C#. Microsoft. (2020). https://visualstudiomagazine.com/Articles/2020/01/21/decision-tree-classifier.aspx

[46] Gambella, C., Ghaddar, B., & Naoum-Sawaya, J. (2021). Optimization problems for machine learning: A survey. European Journal of Operational Research. https://doi.org/10.1016/j.ejor.2020.08.045. Sciencedirect. https://www.sciencedirect.com/topics/computer-science/linear-discriminant-analysis

AUTHOR

I am a holder of a PhD in computer sciences and actually I am a researcher, professor/lecturer and consultant at IBISTM. My research is related to Transformation Projects, AI, Polymathism, and Enterprise Architecture.