Smart2B
Smartness to existing Buildings

UPGRADING SMARTNESS OF EXISTING BUILDINGS THROUGH INNOVATIONS FOR LEGACY EQUIPMENT

Deliverable 3.2

User/Building Knowledge Extraction, AI4EU-based Algorithms and Orchestration

Deliverable Lead: EB

Deliverable due date: 31/08/2022

Actual submission date: 11/11/2022

Call identifier: H2020-LC-SC3-2018-2019-2020

This project receives funding in the European Commission's Horizon 2020 Research Programme under Grant Agreement Number 101023666.

Document Control Page	
Title	User/Building Knowledge Extraction, AI4EU-based Algorithms and Orchestration
Editor	EB
Related WP	WP3
Contributors	Marco Martellacci (EB), Veronica Munaro (EB), Bernardo Starita (EB)
Creation date	03/06/2022
Type	Report
Language	English
Audience	☒ public ☐ confidential
Review status	☐ Draft ☐ WP leader accepted ☒ Coordinator accepted
Action requested	☐ to be revised by Partners ☐ for approval by the WP leader ☐ for approval by the Project Coordinator ☐ for acknowledgement by Partners

Partners

Table of Contents

List of Figures

Abbreviations and acronyms

Abbreviation	Definition
KG	Knowledge Graph
IDE	Integrated Development Environment
ML	Machine Learning
MIMO	Multi-Input Multi-Output
INT-ENER	Intelligence building knowledge layer
HLUC	High Level Use Case
McLS	Multi-criteria Load Scheduling

Revision history

Version	Author(s)	Changes	Date
1.0	Marco Martellacci	Proposed ToC	28/06/2022
2.0	Marco Martellacci	Internal Review	18/09/2022
3.0	Bernardo Starita	Internal Review	19/09/2022
4.0	Bernardo Starita/Veronica Munaro	Internal review	23/09/2022
5.0	Jesus Sanchez (OdinS)	Internal Review	27/9/2022
6.0	Iakovos Michailidis (CERTH)	Internal Review	30/9/2022
7.0	Enerbrain team	Addressed reviewers comments	10/11/2022
8.0	Nuno Mateus (EDP)	Final revision	11/11/2022

DISCLAIMER

Executive Summary

Task 3.3 "Development of the User/Building Knowledge Extraction, AI4EU-based Algorithms and unified orchestration of building actions" in Smart2B intends to create a building intelligence knowledge layer designed to extract relevant building's features, such as information about the building's energy consumption, and to describe the links among data in the form of a Knowledge Graph.

As part of the Task 3.3, a unified translation will also be implemented for managing the high-level actuations received from the Task 4.1 "Implementation of multi-criteria and multi-level asset management services" services of WP4 to convert into low-level commands with the specific format over smart devices and local management systems in the building sites.

The goal of the present deliverable is setting the foundations for driving all the developments to be performed within the Task 3.3 throughout the entire project. In particular, it presents the outcomes achieved during this first stage of the project (until M12). In addition, the deliverable presents the approach which will be followed to realize these developments, by also proposing an architecture to support these developments.

The outcomes of this task will be exploited during the second part of the project to complete the implementation of the proposed software components and their integration within the Smart2B architecture. Indeed, at a second stage of the project, the partners involved within Task 3.3 will work to provide an implementation adhering to the proposed conceptual model's specifications, and this implementation will also be validated within the pilots, thus allowing to demonstrate the correctness of the overall proposed approach.

It should be noted that significant outcomes achieved to reach both goals will be also shared through the publication of the various AI resources into the AI4EU repository (datasets, Apps, software components and libraries, AI models in the energy sector), in order to cooperate with the AI4EU community and actively contribute to its activities.

1. Introduction

1.1. Objectives & Scope

The goal of the present deliverable is setting concrete foundations for driving all the developments to be performed within the Task 3.3 throughout the entire project. In particular, the deliverable presents the approach which will be followed to realize these developments focused on the three following major contributions:

1. The development of a component detecting, recognizing, and identifying a clear and accurate energy consumption pattern of a building (the so-called unique energy signature) by leveraging specific algorithms. In order to achieve this goal, the authors will investigate the current state of AI4EU resources and identify assets or services that can be reused as valid starting points of this activity.
2. The development is a component to translate high-level requirements from Task 4.1 management services into building actions.
3. Extract and update a Knowledge Graph and made it available in Elasticsearch.

This deliverable presents the outcomes achieved to conceive and design the two first mentioned components during this first stage of the project. These outcomes will be then exploited during the second part of the project to complete the implementation of the three components and their integration within the Smart2B project.

1.2. Relation with other tasks

Task 3.3 is interacting with several Smart2B work-packages and tasks, as follows:

- **WP2 – Development of Smart2B Devices & Building Interface**

  T2.1 - Development of novel cost-effective IoT sensors and actuators: the higher-level architecture developed within this work package will benefit from the environmental information provided by the smart devices developed in T2.1 in the form of inputs to the building signature identification procedure.

- **WP3 - Development of Smart2B Platform & APIs**

  T3.2 - A knowledge graph including the data curated, annotated and provided by T3.2 (integrated with the feedback provided by the users/actors through user-centric services in WP4) will be developed.

  T3.4 - The Knowledge Graph developed will provide the inputs to Task 3.4, thus enabling the development of semantics-based APIs to allow SPARQL queries from WP4 services.

- **WP4 – Development of Smart2B Services**

  T4.1 - A holistic unified translator will be implemented within the T3.3 to convert the high-level actuations received from Task 4.1 management services of WP4 into low-level commands with the specific format over smart devices and local management systems in the building sites.

  WP4 services (all tasks) - will provide relevant inputs to the INT-ENER component to the realization of the Knowledge Graph.

1.3. Structure of the deliverable

The document is structured as follows:

- Chapter 1 - Introduction**:** presents the deliverable’s objectives and relation with others project tasks.
- Chapter 2 - Related works: reports a state of the art mainly focused on Algorithms and models for User/Building knowledge extraction.
- Chapter 3 - Intelligence building knowledge layer (INT-ENER): shows the approach followed during the project to realize the Intelligence building knowledge layer. In addition, this chapter also report the requirements elicitation of the component and to conceive its architecture, and describes the Knowledge Graph role.
- Chapter 4 - Translator of the building actions: illustrates the approach followed to realize the Translator. In addition, this chapter starts to explain structural requirements of this component.
- Chapter 5 – Conclusions: draws the conclusions and future developments of the work.

2. Related works

Since the concept of User/Building Knowledge Extraction is quite new and it bases are still being founded, it can be interesting to take a step back and examine its fundamentals, by analyzing which concepts are being focused on and which are the trends the industry seems to have been following. In this regard, this chapter begins with the review of the fundamentals related to the concept of User/Building Knowledge Extraction focusing mainly on Algorithms and models for User/Building Knowledge Extraction available on the AI4EU platform. The idea behind this analysis of the state of the art within AI4EU, reported in the following section, is to exploit and reuse open artificial intelligence algorithms and other resources, which can be taken as reference to drive the future developments conducted within Task 3.3 during the Smart2B project. The second part of the chapter analyzes two others relevant topics for the development within Task 3.3: the generation of knowledge graphs within the scope of Smart Buildings and methods for extracting building energy consumption patterns.

2.1. Algorithms and models for User/Building Knowledge Extraction available on the AI4EU platform

The study on the state of the art presented in this section takes into account the resources offered by the H2020 AI4EU platform [6] and by other existing on-demand platforms.

In particular, the focus is on resources such as:

- Data analysis algorithms (e.g. deep reinforcement learning) to identify the unique signature of the user/building in terms of energy consumption, indoor comfort and air quality;
- Data models for KG, with a particular focus on the concepts of User/Building, Smart Building, Energy;

2.1.1. Analysis of Available Resources on AI4EU

This section reports the analysis of the resources available on the portal AI4EU. The adopted approach for this analysis includes two steps:

STEP 1: search for potential reference on the catalog of the portal AI4EU [6], leveraging a set of 'ad hoc' keywords which are specified in the field "Search Keywords" of the catalog's page reported in Figure 1. The result of the search is a set of found resources.

STEP 2: empirical evaluation of the list of resources to select the ones that appear more suited for this study.

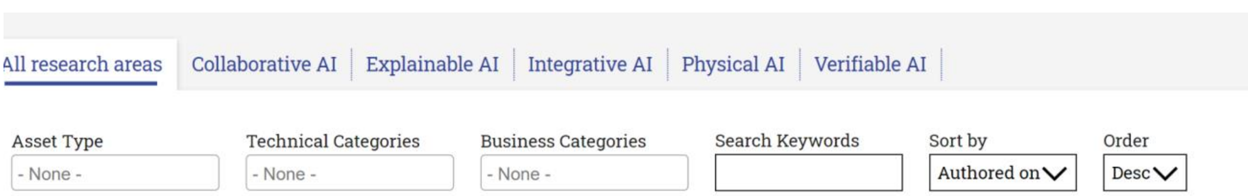

Figure 1: AI4EU search engine

In the first attempt to find the resources available on the portal AI4EU the keyword "signature" has been used, however, this search has not produced any results.

In the second attempt, the keyword "energy" has been used. The list of results achieved by applying the query based on this keyword are reported in the following. In the following list, the results are grouped by asset type and each result is represented by its name and a corresponding brief description.

- **Dataset:**

1. Portuguese Transmission System Aggregated Load Time Series and Encoded Time Covariates [14]

   Aggregated load time series of the Portuguese TSO (Transmission System Operator) for 2018 and 2019 (15 minute resolution) accompanied by encoded time covariates.

2. NVE-Wind-Power-Production-in-Norway [15]

   Hourly wind power production data in MWh/h from 69 wind farms in Norway.

3. Aggregated data of radiation measured in Asturias [16]

   Dataset contains radiation data measured by FAEN (Fundación Asturiana de la Energía) in a series of locations and during a period from 2008 to 2018.

- **As a service:**

4. Short-term load forecasting model for TSOs (LightGBM) [17]

   A forecasting service for predicting the Portuguese aggregated electricity load time series (15-min resolution, 24hr forecasting horizon). The service makes use of the LightGBM algorithm.

5. Load Forecasting Databroker [18]

   A databroker service used for loading time series to forecast machine learning models.

6. SARIMA load forecasting model [19]

   A seasonal ARIMA load forecasting model for boiler rooms in district heating networks (AIExperiments Asset).

7. Prosumer Electricity Load Clustering Model for Demand Response Applications [20]

   A Kmeans clustering model for prosumer's electricity load categorization meant for demand response applications.

- **Docker Container:**

8. DIDA Platform [21]

   The DIDA (Digital Industry Data Analytics) Platform is an OSS Digital Manufacturing Platform, aiming to become a reference implementation for any Industry 4.0 Data Analytics need, enabling the development of applications in several industrial domains.

Analyzing the obtained results, it should be underlined that the performed search has found in the AI4EU's catalog assets of type "Dataset", "As a service", and "Docker Container", while it has not found assets of type "Executable", "Jupyter Notebook", "Library", and "ML model".

During Step 2, the evaluation of each resource found at Step 1 has been conducted based on the asset description reported on the same AI4EU portal. The main considerations that emerge from this assessment are that resources number 4, 5, 6, and 7 seem interesting for the Task 3.3 development and thus their potential should be taken into account in the second part of the project, during the implementation of the proof of concept, while the other resources do not seem useful for the continuation of the work. In addition, it should be noted that, even the first attempt finalized to search resources on "signature" demonstrated the lack of useful resources, some ML models included in the list of resources selected at Step 2 are focused on the prediction (e.g., the numbers 4 and 5 of the list above) and for this reason they can be taken as reference for this work.

Given the fact that AI4EU is a live, constantly evolving, open repository of relevant tools, the above-described application of the search and subsequent selection approach of valid resources on AI4EU will be followed by a future activity that is planned to be conducted in the second part of the project and that will consist in a more detailed evaluation of the selected resources, with the final aim to potentially reuse some parts of them.

2.2. Knowledge graphs within the scope of Smart Buildings

In the field of Smart Building, the different involved devices adopt a broad range of advanced technologies. However, these technologies are mostly closed systems with a limited ability to interact and exchange data, as consequence the data provided by the building's devices are domain-specific and not interoperable [22]. To contribute to enhancing interoperability among hardware and software components of the smart building, various generic solutions have been proposed in literature. One of the most interesting proposed solutions to solve the interoperability issue within smart environments is based on the Knowledge Graph, as reported for example in [23, 24]. "Knowledge Graph" term represents a graph-based structure that includes real-world entities and their relationships, providing in this way detailed information about every examined concept. Such approach is based on the idea that machines can understand the meaning of the exchanged information, thus contributing to enhancing their semantic interoperability.

Based on the above considerations, the works within Task 3.3 intend to exploit a solution based on KG, thus contributing to investigate its promising potential within the Smart2B case studies. In this context, the KG can also help to extract new building's knowledge thanks to reasoning tasks that allows to infer, by means of rules, logical consequences from a set of already asserted facts [25].

2.3. Building energy consumption patterns

Several approaches can be exploited to assess the energy usage of building components. All these approaches can be classified into two main groups: calculated and measured [26]. The first allows to determine the performances in the design phase, while the second enables to evaluate the performance in real conditions. It should be noted that the second is more precise and accurate, but it often requires measuring several different parameters within the building [27]. A method included in the second group that provides for example the possibility to estimate the total thermal transmittance of the building envelope is the **energy signature**, by estimate the overall power loss related to the temperature difference. This method, whose specifics are described in the following section, is adopted as a reference for the developments within Task 3.3.

2.3.1. Building's unique energy signature

The unique energy signature of a building is a model that describes the correlation between external temperature and general consumption. This information can be useful to validate the optimal control strategy from a high-level point of view and it is necessary to provide a significant input to the on-board safety control intelligence developed in Task 2.3. Figure 2 shows an example from literature [12]. The consumption in kWh is collected and represented as a function of the external temperature in °C and °F for weekends and weekdays. The fit of this data is a broken line with different steepness for the two cases. This is called the building's signature and it is unique for each system. The steepness of the correlation suggests the quality of the control strategy, as it gives an indication about the entity of the consumption and the responsiveness of the controller to temperature variations.

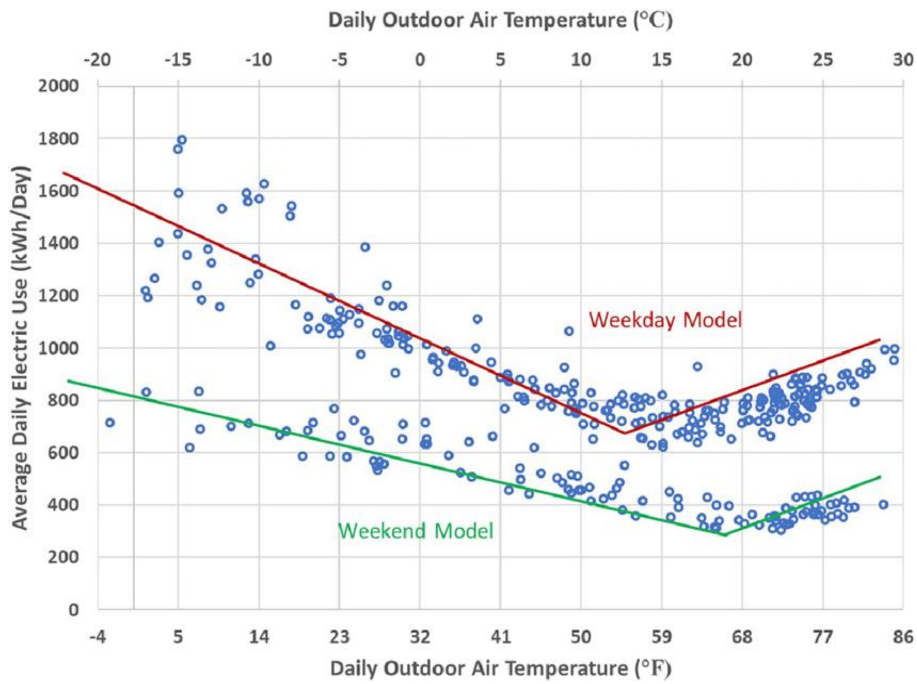

Figure 2: Building's energy signature for weekend and weekdays (from [12]).

2.3.2 Types of ML to predict energy signature

In order to predict building characteristics based on energy signatures, it is possible to rely on supervised or unsupervised learning approaches. For this reason, this section focuses on and reviews these two different ML techniques, and then briefly analyzes the different software solutions supporting these techniques.

- **Supervised learning**: It is a technique that aims at building a model from labeled training data, which allows to make predictions about unavailable or future data. So, the supervision technique leverages a set of samples where the output signals are already known, and the used learning algorithms can be divided into:
  - Classification algorithms, if the labels have a discrete size;
  - Regression algorithms, if the labels have a continuous dimension.
- **Unsupervised learning**: In unsupervised learning, as opposed to supervised learning, the data is unlabeled. The most used technique for this approach is clustering, which allows to aggregate within groups (called clusters) data on which there is not a prior knowledge of group membership.

Software Solutions supporting supervised and unsupervised techniques

The choice of frameworks dealing with machine learning is very wide (e.g. TensorFlow, Scikit-learn, H2O, Apache Spark, Apache Flink). Within Task 3.3 it is restricted to the field of open-source projects, the scope of this choice. As reported in Section 3.3.2, the final choice of software solutions for the development of the proof of concept fell on the following solutions, i.e. **Spark** [4] and **Jupyter** [5], which are both considered very valid by several significant studies [8, 9, 10].

Spark [4] is not just a machine learning framework but a unified platform for parallel data processing oriented to the world of big data. Spark's ML library contains several machine learning algorithms ranging from supervised and unsupervised to recommendation algorithms.

Jupyter Notebook [5] is an application based on the client-server model that allows the creation and sharing of web documents in JSON format, which follow a pattern and an ordered list of input/output cells. These cells offer space for codes, markdown texts, mathematical formulas and equations or multimedia content. Processing works on a web-based client application that must be started with a standard browser. The Jupyter documents created can be exported as HTML, PDF, Markdown or Python documents or alternatively they can be shared with other users via email, Dropbox, GitHub or your own Jupyter Notebook.

The two main components of Jupyter Notebook are a set of different kernels (interpreters) and a dashboard. Kernels are small programs that process specific requests in a language and react with related responses. A standard kernel is IPython, a command line interpreter that allows you to work with Python. Over 50 kernels provide support for other languages like C++, R, Julia, Ruby, JavaScript, CoffeeScript, PHP or Java. The dashboard serves on the one hand as a management interface for individual kernels and on the other hand as a center for the creation of new documents, Notebooks or to open existing projects.

3. The intelligence building knowledge layer (INT-ENER)

The INTelligencE buildiNg knowlEdge layeR (INT-ENER) is a layer that takes care of the extraction of information about the building in terms of energy consumption and of the generation of a Knowledge Graph from curated and annotated data in Task 3.2 "Development of Data Models, Automatic Semantics, Filtering and Storage" and the inputs from WP4 and WP5. The Knowledge Graph will be stored in Elasticsearch along with the energy consumption information and made available to the Platform API developed in T3.4.

Information such as technical description of devices, encoding/decoding of commands, onboard intelligence algorithms are handled to enable artificial intelligence. The layer will constitute the engine used by the API developed in T3.4 to expose information upstream (feeding WP4 and WP5 frameworks) as such it constitutes a key element of the Smart2B platform.

This component will constitute the bridge between the Platform API developed in T3.4 and the knowledge generated from the data lake found in the Smart2B Storage component developed in T3.2. The component will be composed of routines that guarantee to keep updated the Knowledge Graph by checking for changes in the data lake registry and, if any, insert them appropriately in the knowledge graph. At the same time, the component must answer the query arriving from the platform northbound APIs (interface to WP4 and WP5). Therefore, the component must be up and running on the Smart2B platform to be responsive on both sides. The component will be developed in Python by exploiting Graph databases engines or graph libraries [2].

Knowledge extraction will be performed by analyzing the data collected from buildings in terms of consumption and environmental parameters to identify the so-called unique energy signature of building/building area. The energy signature is a correlation between consumption and external temperature, which is described in depth in subsection 2.3.1.

The INPUTS of the component are the following:

- Data provided through the Smart2B Elasticsearch (e.g. Weather data, building data);
- Machine Learning model;
- Data from WP4 services.

The OUTPUTS of the component are the following:

- The energy consumption at different scales (office/room, floor, buildings) and different time horizons (1h, …, 24h). This use case will use statistical and machine learning technologies to perform the different forecasts. This output is provided by CalcSignature service, as specified in the following section;
- Translation into low-level signals of the high-level commands specified by services of WP4;
- Knowledge Graph.

A complete view of the interactions (and the corresponding exchanged data) between INT-ENER and the other Smart2B system components is represented in Figure 3.

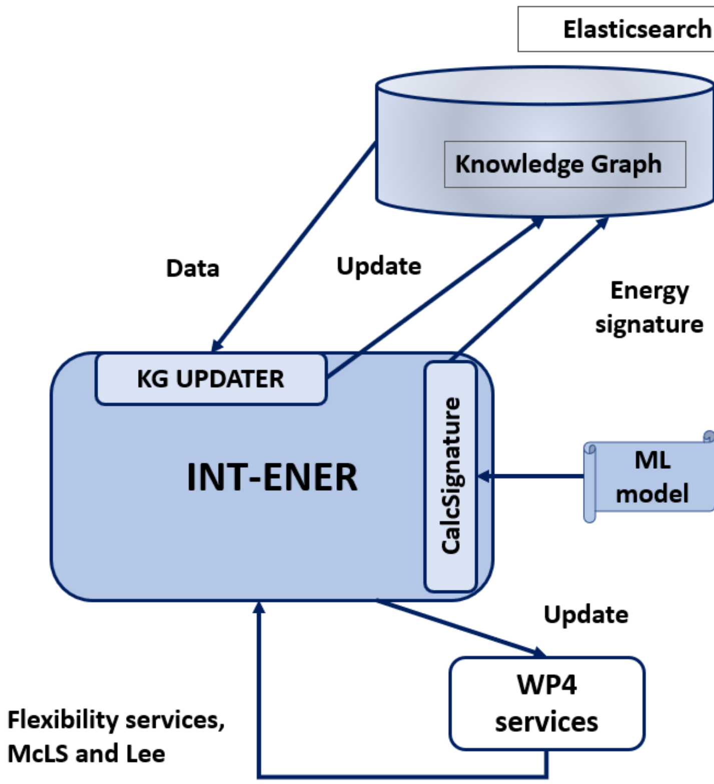

Figure 3: The proposed architecture for INT-ENER

As represented in Figure 3, the model used to extract the signature can be considered a further input of CalcSignature. This feature allows the stakeholders to change the ML model in case some condition is changed. In particular, it will become essential if the proposed solution will be integrated in future in a Digital Twin of the smart building [13]. The latter is an integrated multi-physics, multiscale replica of the physical building that uses the best available physical or simulation models, sensor updates, data history, etc., to mirror the life of its corresponding twin.

3.1. The approach followed to realize INT-ENER

The approach followed to realize INT-ENER includes different steps as shown in Figure 4. We start with a state of the art (main outcomes are reported in Chapter 2) and the elicitation of the requirements. These steps are followed by design and development. The final step is the evaluation performed in two different case studies. In this regard, the first preliminary case study, proposed by Enerbrain, provides a valid context to assess the results of the signature calculation, thus allowing to validate the software component against the functional requirements/specifications. On the other hand, the second case study, proposed and analyzed within the Smart2B project, will test the integration of INT-ENER with other components of the Smart2B architecture and in particular with the Storage Component, the NGSI-LD Broker and the Edge Layer.

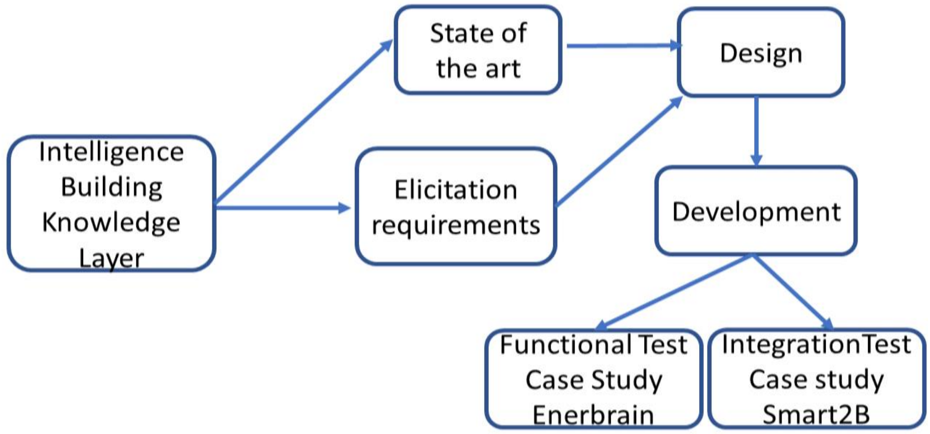

Figure 4: Approach to conduct the INT-ENER 's implementation

3.2. Requirements and Specifications of the INT-ENER services within real case studies

3.2.1. The CalcSignature service

Among the major implemented services, the INT-ENER includes the CalcSignature service which will predict individual profiles for user energy consumption and production. This result will be made available in the online storage to other Smart2B components.

As described in the introduction of Chapter 3, inputs of this service are the monitoring data of the individual buildings, as well as other data (e.g., weather data).

At this stage, the innovation introduced lies in the possibility to deep dive into the consumption analysis in detail, both from a spatial and from a timing point of view. The objective of this task is not only to identify building's properties, but to enable a *prediction* of future use. As a result of this phase, we expect to be able to identify **exclusively from data** relevant system's features, such as the energy signature. Our starting point is the dataset collected from an Enerbrain's building. We will explore different approaches and evaluate the best-performing one.

3.2.2. Case Studies and elicitation requirements

For the analysis, implementation and subsequent evaluation of INT-ENER we refer to two different case studies. The first is proposed by Enerbrain while the second case study is the one analyzed within the project and will therefore be analyzed when the required inputs are provided by the various partners.

3.2.2.1 Case study proposed by Enerbrain

In order to understand the strategy to be employed, it is necessary to rely upon a significant dataset to develop it and test it. To this end, Enerbrain tapped into its own expertise to extract data from an Italian shopping mall with a similar behavior to that expected from Smart2B pilot's buildings. It includes the following fields:

1. **Id_building**: number that identifies the building for Enerbrain
2. **Area**: portion of the building under control (room)
3. **Type** (vh = heating, vc = cooling, se = shutters enabled, ve = ventilation enabled): operative mode, it depends on the season and on the actuators type
4. **Temperature setpoint [°C]**: desired daily internal temperature profile

5. **Relative humidity setpoint**: desired relative humidity
6. **CO_2 setpoint [ppm]**: desired daily CO_2 concentration
7. **Internal temperature [°C]:** instant building internal temperature measured by eSense
8. **External temperature [°C]:** instant outside measured temperature
9. **Command**: control action sent to the actuators
10. **Bypass**: operative mode during which the commands from existing BMS are sent directly to the actuators, skipping the EB algorithm
11. **Start of comfort [n]**: date and time when the comfort setpoint must be guaranteed (during building operative hours), in epoch
12. **End of comfort [n]**: date and time when the comfort setpoint is not mandatory (night, early morning), in epoch
13. **Energy consumption** [kWh]: average energy spent with a 5 minute sampling time
14. **Timestamp [n]**: current date and hour, expressed in epoch

At the first stage we do not take into account humidity and CO_2, but we consider only temperature.

Part of the adopted dataset will be used in the pilot to train and build the model that calculates the signature, while another part will be used to test the obtained model. Indeed, in general, to train any ML model, the dataset must be broken down into data for training and test data. The first subset is used to train the model, while the second subset is provided as data input to the model to perform predictions which are in turn compared with the expected values.

For the analysis of this dataset, it is adopted the most common subdivision used in literature, which provides that the percentage chosen for the training set is **80%** while for the testing set it is **20%**.

3.2.2.2 Case study set in the context of the Smart2B project: User-centric Energy Profiling & Energy Forecasting

This part will be clearer as soon as the needed dataset will be generated and provided by the specific case study in the context of the Smart2B project. The goal is to complete it for the next deliverable for Task 3.3. foreseen for the end of the project.

3.3. Overview of the software architecture

3.3.1. Calculation of the signature: selection of the ML model

In line with the proposed approach, we train a model by applying a learning algorithm to a set of meteorological observations and internal data collection, which are labeled with energy consumption to let the algorithm determine the expected energy consumption, based on the data collected. Considering that energy consumption is expressed in the dataset as a continuous value (e.g. in KWh), the search for the software solutions to be adopted to apply the ML method must focus on solutions that support supervised regression algorithms. When selecting a valid ML solution, it is also important to account for the type and shape of the inputs that the model is expected to receive, which has an impact on its update strategy. Indeed, it should be noted that, once the forecast model is defined, the updating of the model itself could be performed either by passing as input a continuous data stream (in the form of data in motion) or passing an aggregated set of data at defined intervals (batch mode).

3.3.2. Selection of the solutions for ML

For the implementation of CalcSignature, the choice fell on the following software solutions: **Spark** [4] and **Jupyter** [5], which have been already introduced in Chapter 2.

Spark provides different ways to treat the data flow regardless of whether the flow is batch or continuous, and therefore it allows the manipulation of data when reading the original data, when preparing the data set to the learning component, and in the next phase to transmit the processing result to other storage or processing systems (such as text files, database systems, etc.). Furthermore, the use of structured APIs makes the programmer's job easier, allowing him to work on a complete data management system. Nevertheless, another Spark's important feature is that its engine exploits distributed computing and therefore all operations, from the manipulation of the data to its processing, can be performed on a single machine as well as on a cluster of machines.

The implementation of CalcSignature will be conducted using Python as the Spark access API. In this regard, tools integrated into Python such as the graphing modules for the representation of informative data, and the Pandas module for data analysis will be used.

The choice of the IDE for the management of the pyspark modules was focused on JupyterLab under Anaconda. JupyterLab is a flexible tool that allows the integration of the modules for creating graphics within the script, exploiting insert comment blocks formatted in the Markdown markup language. The installation of Spark can be ready to use in a few minutes thanks to Anaconda, which is a platform for Data Science with Python. Through Anaconda it is possible to build one or more development environments for Data Science within which installing all the packages necessary for development. The environment also includes Conda which is a package and environment manager that simplifies the installation of additional packages by checking dependencies, updating already installed packages or downgrading in case of incompatibility between packages.

3.3.3. Knowledge Graph Updater

Through Knowledge Graph Updater component a Knowledge Graph is built, continuously updated and made available to other Smart2B services on the online storage component. The Graph will be realized according to NGSI-LD linked data model and populated based on the data received from the online storage.

Task 3.4 to develop the appropriate semantics-based APIs to enable SPARQL queries from potential northbound services (e.g., WP4 services).

3.3.4. Exposed API

Since Task 3.4 aims to providing an interoperable access to the resources and operations of the Smart2B platform it is essential to support this goal by starting to provide the basis to allow to access to the INT-

ENER. In this regard, the data behind main services\ functionalities of the INT-ENER can be queried through the NGSI-LD Broker.

3.4. Notes and Deployment considerations

3.4.1. Distribution on AI4EU

The study of the art reported in Section 3.1.2 demonstrated the lack of models to calculate the signature within the AI4EU platform. In order to contribute to overcoming this issue, a future work can be the distribution of the Smart2B layer proposed in this Chapter on the AI4EU platform.

4. Translator of the building actions

A holistic unified translator will be implemented for managing the high-level actuations received from Task 4.1 management services to convert into low-level commands with the specific format (e.g. directly communicating with the NGSI-LD Broker following the achieved data model presented in Smart2B D3.1 FIWARE-based harmonization, data annotations (Task 3.2), filtering (Task 3.2) and storage (Task 3.2) [11]) over smart devices (i.e. those located in WP2) and third party gateways or Building Management systems (e.g. HomeAssistant, OpenHAB) in the building sites.

4.1. Requirements and Specifications of the translator

The translator acts as a transmission element between the requirements specified by the services in Task 4.1 and the actuators, through a machine learning application. Every device contains an edge computing module where an optimal control computes the necessary physical commands to be executed by the actuator, the core element of such a controller is the cost function, which determines the undertaken overall regulation strategy. The translator can simply transmit the cost function parameters provided by the service layer of Task 4.1 but can potentially intervene as an interface between the control action by translating the control strategy into control actions at eNode level. This second functionality will be developed depending on the development of Task 4.1.

5. Conclusions

This deliverable presents Smart2B approach, developed in “T3.3 - Development of the User/Building Knowledge Extraction, AI4EU-based Algorithms and unified orchestration of building actions”, to create a building intelligence knowledge layer on Smart2B platform, designed to extract relevant building’s features, such as information about the building’s energy consumption, and to describe the links among data in the form of a Knowledge Graph. In addition, the unified translation component, that will act as manager for the high-level actuations received by Smart2B services WP, converting it into low-level commands for smart devices and local management systems in the building sites, was also described in this deliverable.

Finally, and as result of the research performed at AI4EU repository, that highlight and identify a lack of related resources, Smart2B will contribute to AI4EU community through the publication of the several AI resources including datasets, Apps, software components and libraries, AI models in the energy sector.

Future developments within Task 3.3 will address the implementation and evaluation of the INT-ENER according to the proposed approach reported in section 3.2, and the realization of the Translator component according to chapter 4. For all the developments, it essential to continue investigate the potential of the resources selected from AI4EU which were classified as potentially interesting for activities of T3.3 as reported in Chapter 2.

6. References

[1] Smart2B D1.1 Specifications Smart2B.

[2] Smart2B D1.2 Prototypes of Smart2B devices, Final Version and technical report.

[3] Smart2B D1.4 Description of the Use Cases.

[4] Spark. [On line]: https://spark.apache.org/.

[5] Jupiter Notebook. [On line]: https://jupyter.org/.

[6] Catalog AI4EU. [On line]: https://www.ai4europe.eu/research/ai-catalog.

[7] Holzinger, A., Kieseberg, P., Weippl, E., & Tjoa, A. M. (2018, August). Current advances, trends and challenges of machine learning and knowledge extraction: from machine learning to explainable AI. In *International Cross-Domain Conference for Machine Learning and Knowledge Extraction* (pp. 1-8). Springer, Cham.

[8] Meng, X., Bradley, J., Yavuz, B., Sparks, E., Venkataraman, S., Liu, D., ... & Talwalkar, A. (2016). Mllib: Machine learning in apache spark. *The Journal of Machine Learning Research*, *17*(1), 1235-1241.

[9] Pentreath, N. (2015). *Machine learning with spark* (p. 338). Birmingham: Packt Publishing.

[10] Pimentel, J. F., Murta, L., Braganholo, V., & Freire, J. (2019, May). A large-scale study about quality and reproducibility of jupyter notebooks. In *2019 IEEE/ACM 16th international conference on mining software repositories (MSR)* (pp. 507-517). IEEE.

[11] Smart2B D3.1 FIWARE-based harmonization, data annotations and filtering storage.

[12] Assessing the Validity, Reliability, and Practicality of ASHRAE's Performance Measurement Protocols (ASHRAE Research Project 1702) - Scientific Figure on ResearchGate. Available from: https://www.researchgate.net/figure/Building-A-daily-electric-energy-use-vs-outside-air-temperature_fig3_330096320 [accessed 6 Oct, 2022]

[13] Khajavi, S. H., Motlagh, N. H., Jaribion, A., Werner, L. C., & Holmström, J. (2019). Digital twin: vision, benefits, boundaries, and creation for buildings. *IEEE access*, *7*, 147406-147419.

[14] Portuguese Transmission System Aggregated Load Time Series and Encoded Time Covariates. [On line]: https://www.ai4europe.eu/research/ai-catalog/portuguese-transmission-system-aggregated-load-time-series-and-encoded-time.

[15] NVE-Wind-Power-Production-in-Norway. [On line]:

https://www.ai4europe.eu/research/ai-catalog/nve-wind-power-production-norway

[16] Aggregated data of Radiation measured in Asturias. [On line]:

https://www.ai4europe.eu/research/ai-catalog/aggregated-data-radiation-measured-asturias

[17] Short-term load forecasting model for TSOs. [On line]:

https://www.ai4europe.eu/research/ai-catalog/short-term-load-forecasting-model-tsos-lightgbm

[18] load-forecasting-databroker. [On line]:

https://www.ai4europe.eu/research/ai-catalog/load-forecasting-databroker

[19] SARIMA load forecasting model. [On line]:

https://www.ai4europe.eu/research/ai-catalog/sarima-load-forecasting-model

[20] Prosumer Electricity Load Clustering Model for Demand Response Applications. [On line]:

https://www.ai4europe.eu/research/ai-catalog/prosumer-electricity-load-clustering-model-demand-response-applications

[21] DIDA Platform. [On line]:

https://www.ai4europe.eu/research/ai-catalog/dida-platform

[22] Costantino, D., Malagnini, G., Carrera, F., Rizzardi, A., Boccadoro, P., Sicari, S., & Grieco, L. A. (2018, May). Solving interoperability within the smart building: A real test-bed. In *2018 ieee international conference on communications workshops (ICC Workshops)* (pp. 1-6). IEEE.

[23] Stavropoulos, T. G., Vrakas, D., Vlachava, D., & Bassiliades, N. (2012, June). BOnSAI: a smart building ontology for ambient intelligence. In *Proceedings of the 2nd international conference on web intelligence, mining and semantics* (pp. 1-12).

[24] Bakakeu, J., Schäfer, F., Bauer, J., Michl, M., & Franke, J. (2017). Building Cyber-Physical Systems–A Smart Building Use Case. *Smart cities: foundations, principles, and applications*, 605-639.

[25] Horrocks, I., Patel-Schneider, P. F., Boley, H., Tabet, S., Grosof, B., & Dean, M. (2004). SWRL: A semantic web rule language combining OWL and RuleML. *W3C Member submission*, *21*(79), 1-31.

[26] Belussi, L., Danza, L., Meroni, I., & Salamone, F. (2015). Energy performance assessment with empirical methods: Application of energy signature. *Opto-Electronics Review*, *23*(1), 85-89.

[27] Nordström, G., Lidelöw, S., & Johnsson, H. (2012). Comparing energy signature analysis to calculated U-values in wooden houses in a cold climate. *WIT Transactions on Ecology and the Environment*, *165*, 411-419.