Integrated Scalable System for Smart Energy Management

Submitted by

Ahmed Al-Adaileh

In partial fulfilment of the requirements for the Doctor of Philosophy degree

Kingston University London, United Kingdom
Faculty of Science, Engineering and Computing
School of Computing and Information Systems

London, December 2021

Acknowledgements

I want to express my extraordinary thankfulness and gratitude to my supervisory team Professor Souheil Khaddaj and Dr Eckhard Pfluegel; you have both been great mentors for me. Thank you for empowering my research and for continually being supportive and helpful. Your recommendations on both research, and career have been inestimable. A warm thanks go to Rosalind Percival and Marie Withers, who supported me unendingly with all the assets and information.

My most substantial appreciation and much gratitude go to my significant other, my wife, Alia, who upheld me with strength and vitality to achieve the best in education and career and relinquished her time as well as offered broad tolerance and commitment to raising our both kids remarkably. My delightful girl Nour, my son Jawad have been so understanding during the PhD research years, that I can't be grateful enough for their collaboration and penances for their dad's time – thank you for your adoration and prayers.

Last but not least, finishing this research would not have been conceivable in the event that I hadn't got huge assistance and backing from my mom, Fatmeh, who constantly dreamed big of me. You were, are and will consistently be my motivation to exceed expectations throughout everyday life. I wish my dad was with me today to see my achievement, yet I realize his blessings will consistently be with me. I thank my dearest siblings for remaining by me and for being supportive without requesting it and consistently offering unlimited companionship and coaching all of these years.

Table of Contents

Acronyms and Abbreviations

List of Figures

List of Tables

Abstract

The planet's reserves are encountering vital challenges and suffer inequitable consumption. The outcomes of the prostration of natural reserves have started affecting every single organism on the globe. Energy is a critical key factor in this aspect because a considerable part of the destruction is triggered by utilising the planet reserves to produce power in diverse forms. The increasing environmental awareness in humans' minds, and the rapid development of smart concepts, home automation technologies in both hardware and software fields, played an essential role in speeding up the progress to apply smart energy management which is needed to revert the situation to its appropriate track by focusing on two main divisions: firstly, producing clean and renewable energy and secondly, reducing the loss of the total generated energy. This research will concentrate on the second approach by proposing, implementing and evaluating a contemporary integrated, scalable, smart energy management framework that assists in reducing the energy consumption in the household sector, covering a range of single households till huge communities and big organisations with thousands of appliances. A number of correspondent strategies and policies which utilise a set of observed and predicted system entities are applied to keep meetings the most relevant quality attributes such as integrability, scalability, interoperability and availability. IoT concepts are applied in this context to connect conventional household appliances to a farm of microservices that implement predictive analytics techniques to reduce energy consumption by applying two main strategies; appliance substitution based on the energy consumption and creating automatic schedules to run appliances based on predictions. A case study is presented on two sample appliances within the household to illustrate the framework validity and deliver percentage figures of the saved energy. Additionally, the framework offers a number of possibilities to provide relevant third parties such as local energy providers, apparatuses' manufacturers, or pertinent government offices with various appliances' operational behaviours under real-life conditions.

1. Introduction and Background

1.1. Overview and Background

Our modern civilisation is highly dependent on all types of energy to sustain and grow further. The solid direct proportional relationship between the modern lifestyle and the amount of consumed energy had begun when earlier human beings started burning woods to produce fire for heating and cooking purposes and ended up constructing a state-of-art, sophisticated and most technologically advanced nuclear power stations to obtain kinetic energy and electricity. This energy gluttony has brought the mostly consumed fossil-based energy resources to its limits and put the environments and surroundings under tremendous pressure that can be observed in the rapid decrease of the forest spaces, the extinction of many organisms, the pollution and dramatic change of the expected climate patterns. Unfortunately, due to the rapid increase of our planet's populations, the rising of living standards, and the slow-moving development in the renewable energy resources sector led to putting more pressure on our traditional resources in the way that according to scientific researches our civilisation may suffer from massive challenges soon.

Delivering a solution to this problem can be done by following two approaches; firstly, accelerating the efforts to solve all problems that prevent the maximum production of clean and renewable energy. Secondly, using the available energy in the way to achieve the golden rule to consume the energy most efficiently while still matching the desired comfort levels. This thesis will concentrate on the second approach by proposing an integrated, scalable framework for smart energy management that targets reducing the energy consumption at the households, buildings and organisations levels.

The research topic of this thesis is evolving around the reduction of the amount of the energy consumed in the household sector by applying a number of smart technologies, namely Internet of Things (IoT), Machine Learning (ML), Artificial Intelligence (AI), and cloud-based Microservices. This topic is addressed by designing and implementing an integrated, scalable system for smart energy management engineered using IoT and cloud-based microservices, assisted by machine learning analytics to gather, process, analyse and predict energy usage habits for an extended period to optimise energy consumption in household sector.

Research questions:

1. What are the mechanisms that can be utilised to manage the energy consumption in household sector?
2. What are the most relevant strategies/policies that should be applied to achieve the maximum energy savings in this field?

1.2. Problem Description and Challenges

Achieving a satisfying level of energy savings by applying smart energy management systems is a very challenging mission [1]. The reason resides in the nature of this highly complex and continuously aggregated environment. There are numerous challenges. firstly, the majority of currently used appliances are legacy appliances that do not support any smart technology and do not allow adding any built-in modules to convert them into smart ones. This problem necessitates applying additional external adapters and actuators to deal with these kinds of appliances, which ultimately adds more complexity and additional costs. Secondly, the lack of standards among the newly built smart devices. Too many different protocols, developed by different vendors, are used in this field, such as ZigBee, Z-Wave and EnOcean [2]. Since direct compatibility among them does not exist, smart home systems must be built around one protocol, or customised communication bridges must be built between the controllers to ensure having two different hardware farms interacting efficiently and adequately. Thirdly, Since the whole concept is relatively new, it is observed that the acceptance and the contribution of people to these kinds of approaches are still not enough. As a result, the willingness to invest in establishing such smart management systems still did not reach acceptable and encouraging levels. The people's consciousness of the necessity of such systems plays a crucial role in the improvement and growth of the whole industry. Fourthly, like any new technology, the average prices of all related hardware and services are still not affordable by most consumers. This adds an obstacle towards establishing the system in the majority of middle-class consumers' facilities. Fifthly, the accuracy of the data collected from some smart home sensors is still not reliable and have the potential for improvement. Finally, in the market, there is a lack of reliable open-source platforms that support the smart home concept. Open-source concept opens great avenues for all developers and manufacturers to closely work together, which may lead to having common standards, cheaper products and services, and finally a friendly environment that is suitable for further development.

1.3. Energy Management Systems' Purpose and Importance

The continuously increasing costs, the need for more energy and the negative influence of energy generation processes on the environment has led to the development and application of energy management systems to monitor, control, decrease energy consumption in many aspects. The desired management systems must prove their ability to reduce energy consumption while maintaining equal levels of consumers' comfort and meeting energy demand. Moreover, on a global scale, there is an urgent demand to save energy in order to decrease and possibly revert the environmental destruction that occurred to the planet [3] [4]. However, it is not only the environmental impact that has led to using energy management rather the limited fossil fuels dependency. Since human civilisation cannot be sustained without having enough energy, and due to the fact that the energy obtained from fossils will not remain forever, the need to replace this energy resource with another, unlimited resource becomes mandatory. Due to many technological challenges, the energy obtained from renewable resources is still significantly below the required levels. For instance, according to Pineda and Tardieu [5] by the end of 2017, only 11.6% of the EU's required electricity has been covered by wind power generation. Here comes the role of the energy management systems to make it possible to reach the same level of comfort using less energy, which can be obtained from renewable resources. The next section summarises the aims and objectives, which are intended to be achieved in this work.

1.4. Aim and Objectives

This work mainly aims to propose an integrated and scalable system for smart energy management, using IoT and cloud-based microservices, assisted by big data analytics capable of gathering, processing, analysing and predicting energy usage habits in order to optimise energy consumption.

The previous bird's-eye view might be broken down into these objectives:

1. Research and analyse current solutions related to smart energy management frameworks.
2. Review and compare machine learning algorithms and techniques to assess their capacity to support the solution being proposed.
3. Propose a new integration system architecture for smart energy management

4. Provide a detailed description of an implementation of the proposed framework, in a selected household environment which consists of a combination of conventional and smart equipment.
5. Using a real-life case study, apply various testing and validation techniques to ensure the correctness, robustness and suitability of the proposed system

1.5. Background, Analysis, Costs and Environmental Impact

The first step to solving any problem is to describe it adequately. Delivering a proper description can only be obtained by measuring, observing, monitoring and recording all related constants, facts and statistics. This approach is illustrated in the next sub-sections delivering some global statistics, followed by national figures from the European Union, UK, China and the USA.

1.5.1. Worldwide Energy Consumption

The total worldwide energy consumption shown in Terawatt per hour (TWh) per year is explained in Figure 1.1 [6] which illustrates the source of energy divided into different sections, including traditional biofuels, coal, crude oil, natural gas, nuclear and other renewables. The picture carries alarming and shocking figures because it illustrates a rapidly increasing energy demand and shows a small percentage of the energy obtained from renewable and clean resources. Starting from the 1950s, the energy demand started progressively getting bigger and bigger until it reached very high levels by the end of 2019. This sudden explosion in the energy demand has been confronted with producing energy from traditional, unclean resources such as fossil fuel and coal, which can have a devastating impact on the environment.

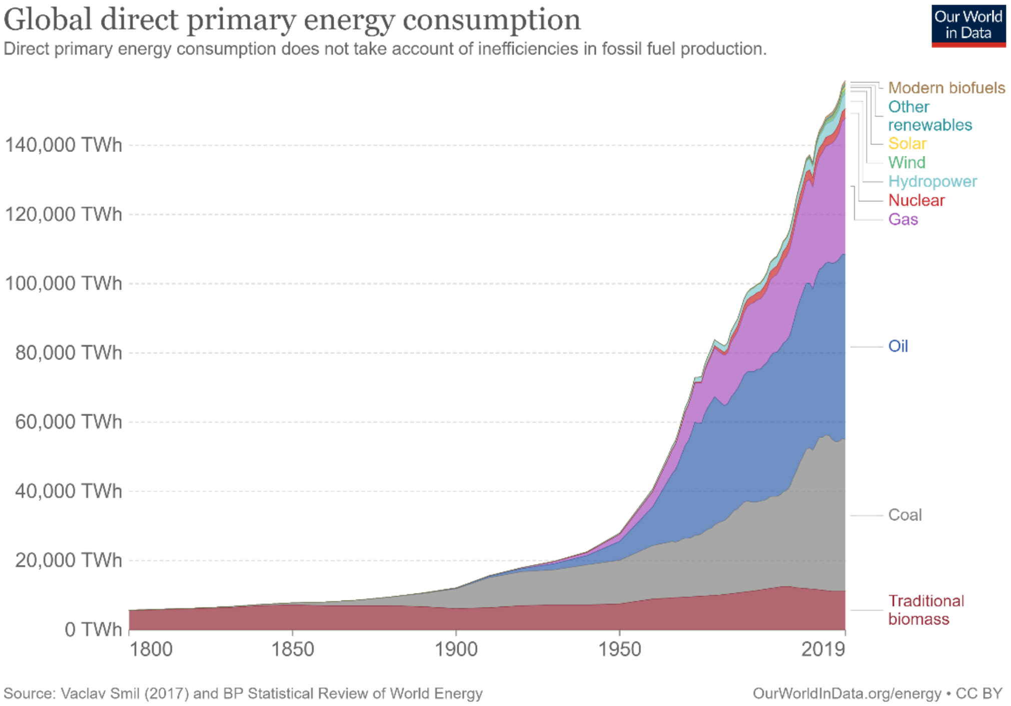


Figure 1.1: Worldwide primary energy sources. [6]

In the next section, we will see that the picture is not different in the EU compared to the global one.

1.5.2. Electricity Generation in Europe

In Europe, the situation was not different compared to worldwide statistics. Figure 1.2 Obtained from the Eurostat [7] illustrates this fact. The figure covers the EU electricity production and consumption in 2020 and shows that the total energy generation increased to compensate for the increased demand. 4.4% of electricity was generated from solar, 13.3% from wind, and 42.8% comes from conventional thermal sources.

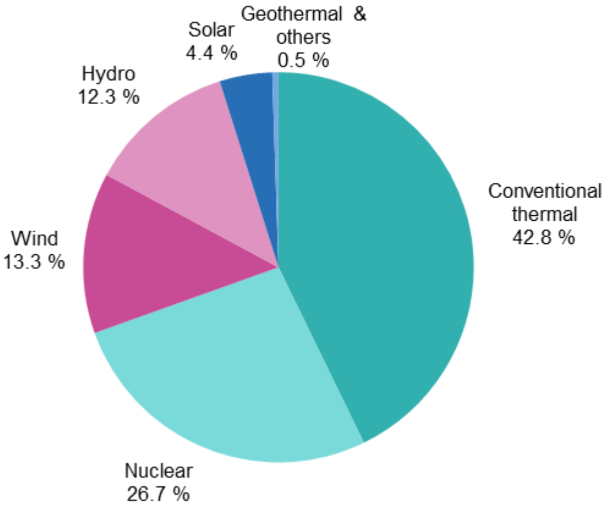


Figure 1.2: EU electricity production by source, 2020. [7]

1.5.3. China's Electricity Mix

In China, unfortunately, the picture is a bit darker in both dimensions: energy consumption increased dramatically, and renewables did not develop at the same rate. Figure 1.3 [8] shows some features illustrating these facts.

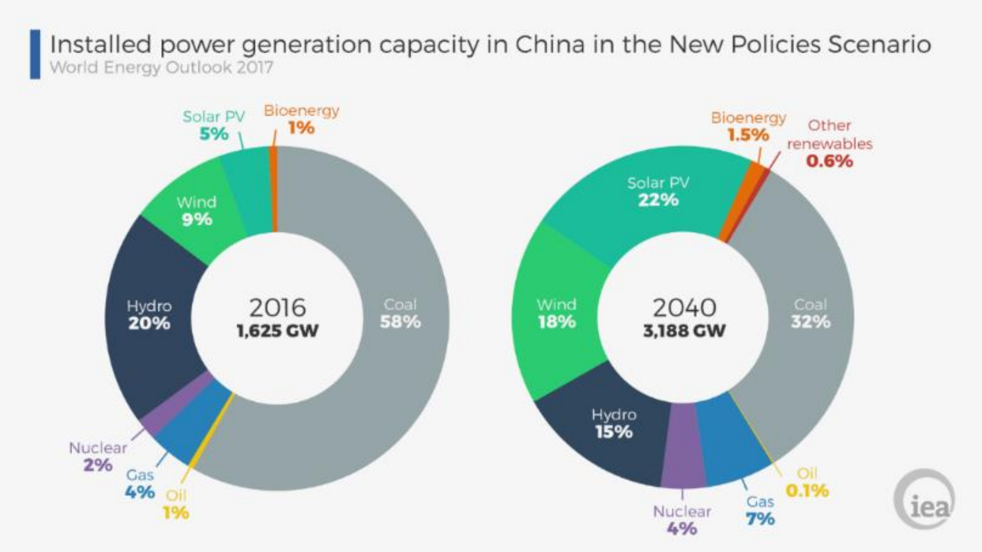


Figure 1.3: China's energy consumption in 2016 with a prognosis of 2040 [8]

Still, an encouraging trend is seen in the reduction of using unclean energy sources such as coal and replacing it with another clean resource such as solar and wind [9]. Completing the scene requires having a look at the power usage percentage in different sections, particularly the residential one. The following section shows some figures from different countries worldwide.

1.5.4. Worldwide Household Electricity Consumption

The differences in energy consumption worldwide are shown in Figure 1.4 [10] which undoubtedly reveals that there is a massive difference in energy consumption between countries. For instance, an average household in a country like Canada consumes about 20 times more than the energy consumed in an average household in India. These figures make it clear that any energy management system should concentrate on the household occupants of countries with high energy consumption rates, considering them as a central element of the system and taking their behaviours and habits under the loupe.

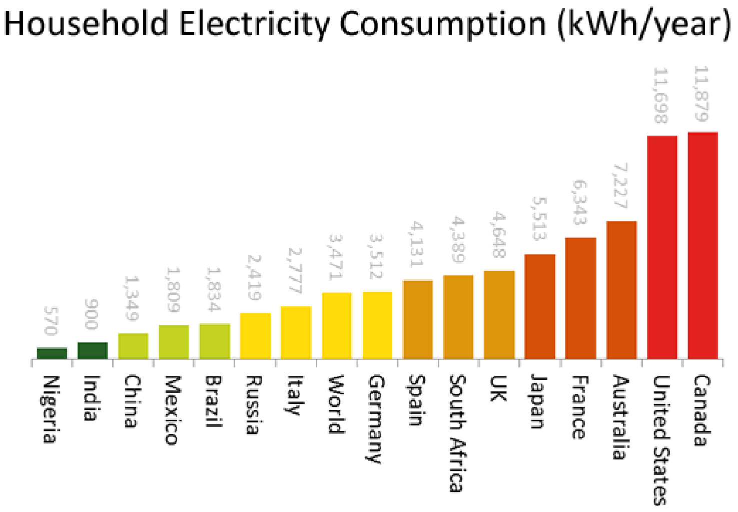
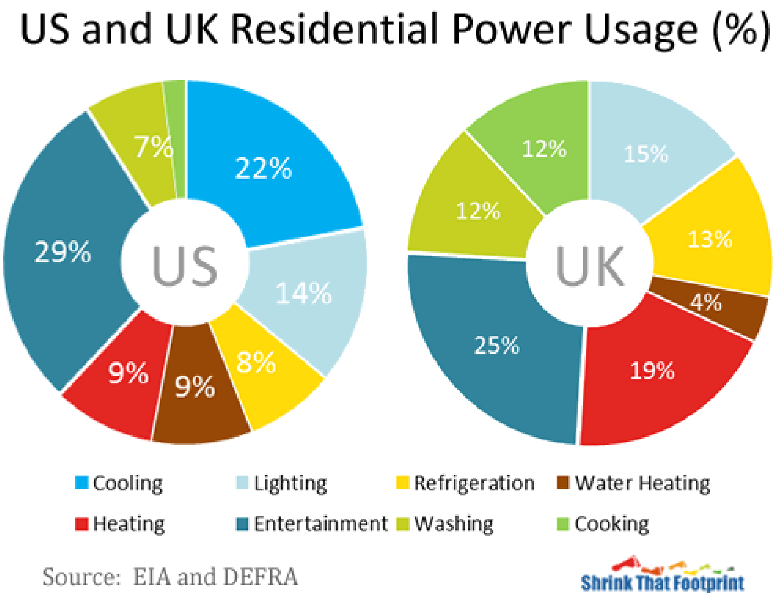


Figure 1.4: Selected Household Electricity Consumption Rates (in kWh per Year) [10]

1.5.5. Residential Power Usage Percentage in the USA and UK

In order to fully understand the residential section power usage, two countries, namely the US and UK, are considered. The power usage is illustrated in Figure 1.5 [11], from both countries, which shows that entertainment, heating and lighting are the most consuming energy in the household. It gives an indicator on which direction should be the efforts concentrated while developing and designing energy management systems.



Source: EIA and DEFRA

Figure 1.5: The USA and the United Kingdom Household Electricity Usage. [11]

A quick look at the percentages of energy consumption in household, transportation and manufacturing segments in the U.S. (Figure 1.6) [12], reveals that these sectors are occupying the highest energy consumption successively; 36% for industrial section, 35% for the transportation from all kinds, 17% for the residential uses and 12% for the commercial activities.

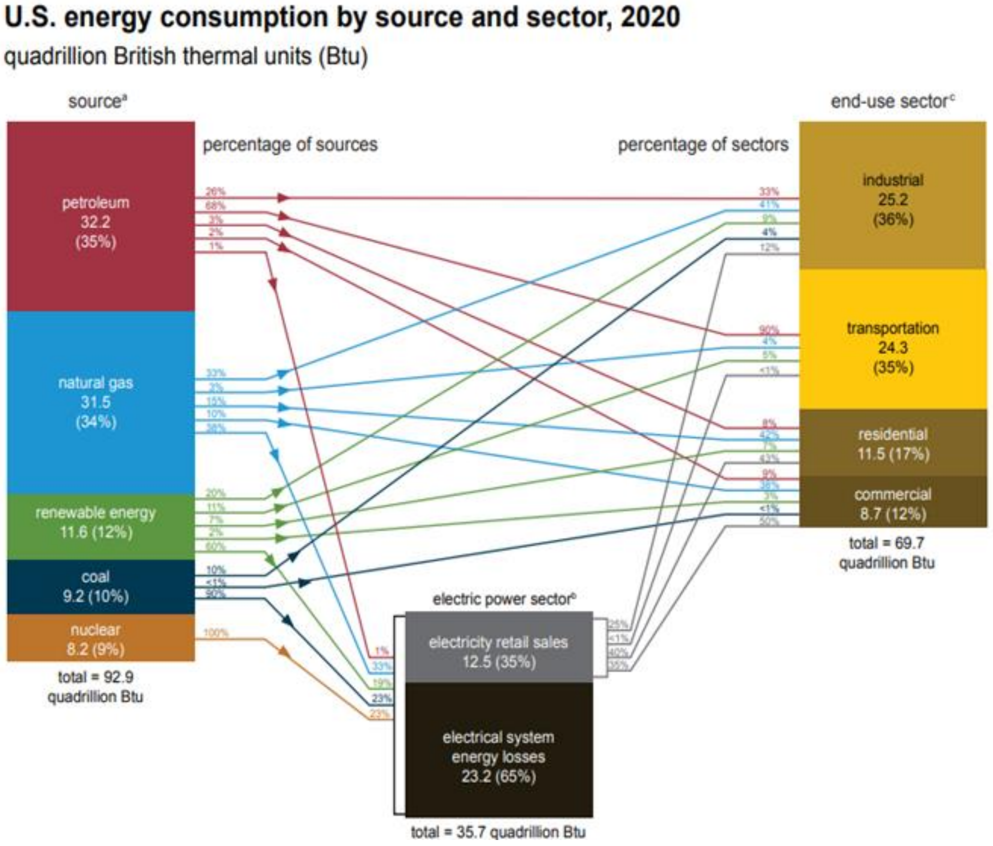


Figure 1.6: U.S. energy consumption by source and sector, 2020 [12]

After considering some figures related to energy consumption, in the next section, some facts related to the costs and the environmental impact of the energy will be presented. This should give an idea about the tremendous amount of money wasted, and the degree of environmental damage.

1.5.6. Electricity Prices Worldwide

Measuring electricity prices in countries is essential for energy management systems. This is because implementing these systems requires investment. In countries where the electricity price is low, it could be that the invested budget to establish and maintain an energy management system may be higher than the saved costs, however, in some countries, this investment will return its benefits during a short amount of time. A quick look at Figure 1.7 [13] gives an idea about the electricity price for the household sector worldwide. According to this chart, Italy has the highest rate (21 US Dollar Cents Per kWh); however, in Sweden, this rate is only 8 Cents per Kwh. So, we can read that implementing energy management systems in countries such as Italy and Germany, where the case study takes place as we'll be seen in the implementation chapters 5 and 6, may be profitable and worthwhile.

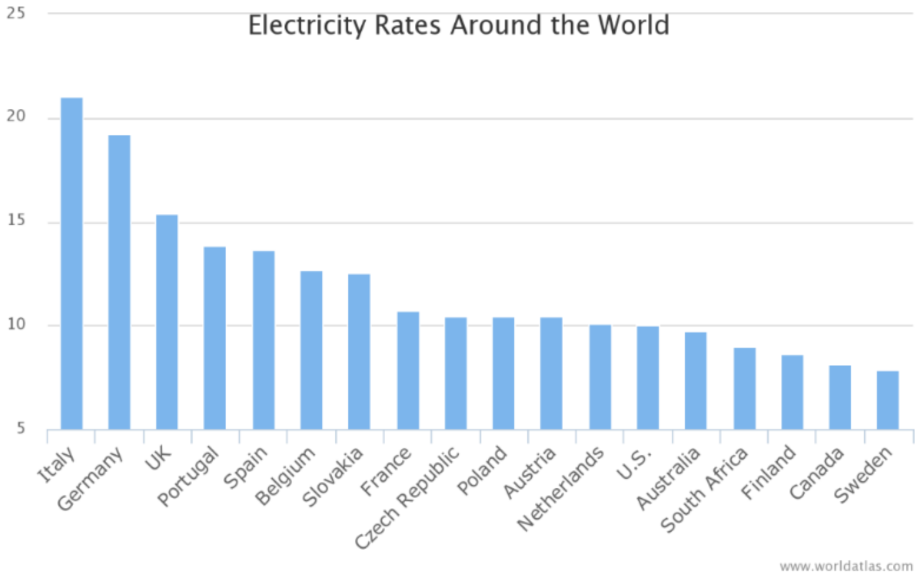


Figure 1.7: Electricity rates around the world [13]

1.5.7. Electricity Power and CO_2 Emissions

Now let us look at some figures related to the environmental aspect. The numbers illustrated in Figure 1.8 [14] shows the countries highlighted according to their Carbon Dioxide emissions resulting from the production of electricity and heat measured by % of the total fuel combustion.

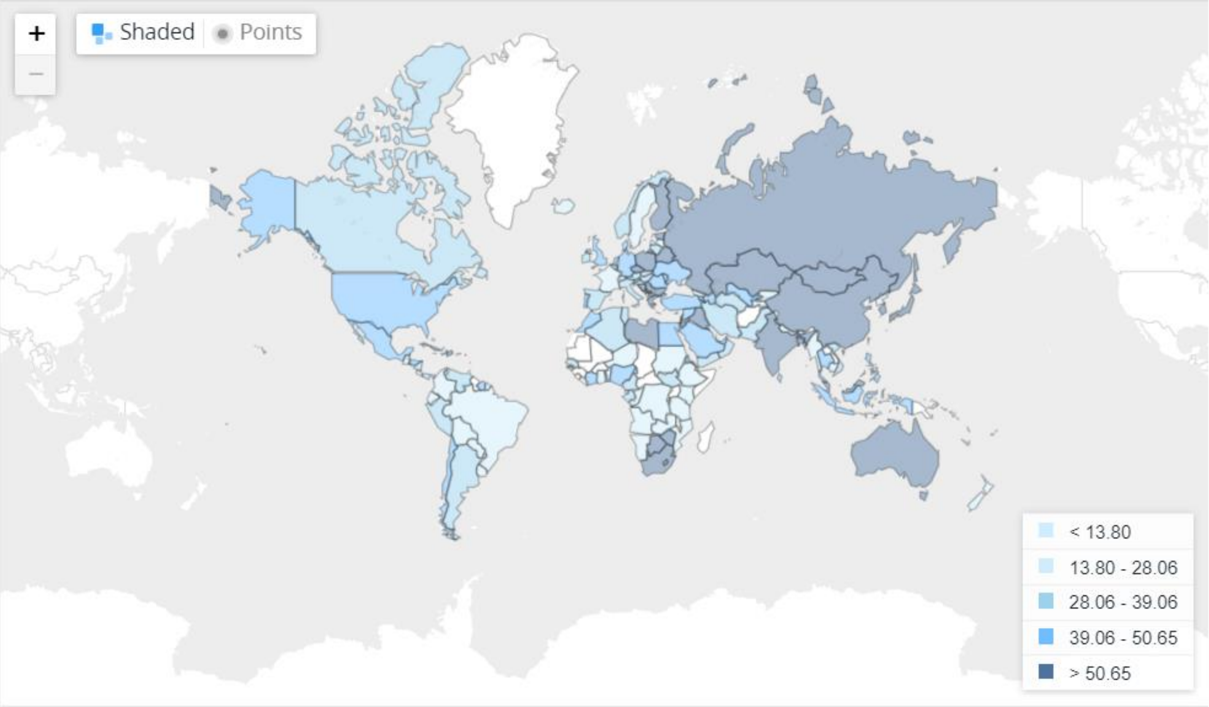


Figure 1.8: CO_2 emissions resulted from the production of electricity and heat [14]

So, from the environmental point of view, implementing and maintaining energy management systems must be considered mandatory in those countries which are dark highlighted on the map. Despite the remarkable challenge to switch from using conventional energy sources to renewable and clean ones, there is a considerable movement towards this type of energy in the last decades.

1.5.8. Renewable Energy Production

According to [15] the global renewable energy generation over the long-term illustrated in Figure 1.9 [16] shows that starting from 1965 till nearly 2020, hydropower was dominating the renewable energy market. However, starting from the year 2000 the graph begins showing an increase in the total amount of energy generated by different renewables such as solar PV, wind, and other renewables.

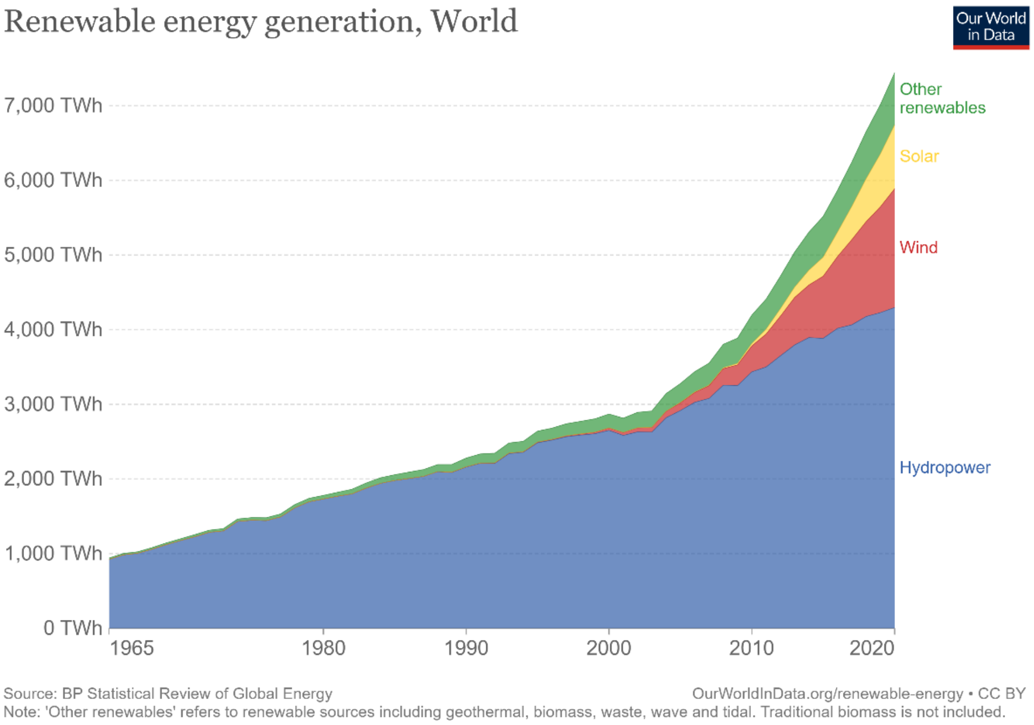


Figure 1.9: Global renewable energy generation [16]

The previously mentioned facts and figures draw a sharp illustration of the negative impact of the energy segment on the global environmental and financial situation. This has motivated and urged governments and organisations to establish, sharpen and promulgate a variety of laws and regulations to regulate the energy sector in order to decrease consumption, and ultimately reduce CO_2 emissions and make further steps toward restoring a clean and sustainable environment.

1.6. Law and Regulations

The importance of the issues related to energy efficiency forced countries and organisations to provide a statutory and legal cover to organise and control most of the actions taken in this area. All law materials aim at one thing: increasing energy efficiency. This section will cover some of these laws and regulations

1.6.1. ISO 50001 - Energy Management Standard

Undoubtedly the proper and efficient energy consumption assists countries and organisations to save money, protect resources and reduce the climate change speed, or even stop it. The International Organisation of Standards has offered a dedicated standard called ISO 50001 to assist and support organisations to develop energy management systems properly and effectively. Organisations that already make use of existing, famous standards such as IOS 9001 and ISO 14001 [17], could easily integrate this new standard into their existing efforts to enhance and maintain high quality and improve their environmental management because ISO 50001 is already implicitly integrated and used in those standards. According to [17], the ISO 50001:2018 standard offers a framework to:

- Grow and maintain strategies and policies to achieve higher degrees of using energy efficiently.
- Meet the strategies and policies by fixing targets and objectives.
- Use the data to increase the degree of understanding of used energy so that proper decisions can be made.
- Properly track and record results
- Evaluate how far policies and strategies work
- Repeatedly enhance and upgrade energy management outlines.

1.6.2. Energy Labelling

An energy consumption labelling sticker was established by the European Union. It should be available on all white goods and lighting bulbs when sold or rented. EU's Energy Labelling Directive contains a detailed description of how to use it and what should it contain. These labels give a clear picture of the energy efficiency on a scale from D (least) to A+++ (highest) energy efficiency, so it is easy for customers to distinguish between high-energy-demanding and low-energy-demanding appliances. Figure 1.10 [18] shows an example of labelling.

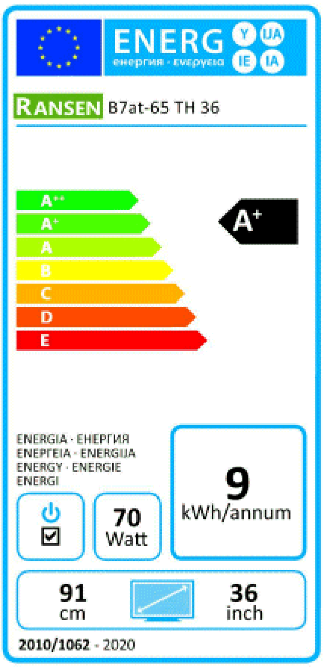


Figure 1.10: Example EU Label [18]

In Europe, almost every device carries this labelling that matches at least the minimum energy level (D). However, those kinds of devices are not achieving any economic profit, so they are gradually disappearing from the market. Due to using this labelling, it is expected to achieve a total saving of 175 MT of oil by 2020, which is equivalent to approximately €465 per household. Energy labels can be created via designated online tools [18].

1.6.3. Ecodesign

Ecodesign guidelines mean that all goods produced by manufacturers and companies need to match customers' needs while consuming the lowest amount of resources and having the lowest negative environmental and social impact. All EU Ecodesign Directives are offered and managed by the EU commission. All newly designed, or already existing systems, products, services or processes are entitled to apply these directives to ensure its compatibility with surrounding environments and reduce its possible damage. Some of its principles include the following:

- Use materials that have the lowest negative environmental impact.
- Minimise the number of used materials during the production process.
- Minimise the number of used resources during the production process.
- Minimise pollution and waste.

Although Ecodesign has too many advantages, it may suffer from some shortages, such as the implementation costs, the risk resulting from dealing with new materials and processes.

1.6.4. Energy Star

According to [19], the Energy Star can be defined as a U.S. environmental protection agency voluntary program that assists organisations, industry and persons financially by saving money and protecting the environment by enhancing energy efficiency. Energy Star's (Figure 1.11 [19]) primary purpose is providing easy, simple and trustworthy sources to make the right decision to choose the most energy-efficient and environmentally friendly products when buying or renting them. American citizens have purchased over 300 million products carrying this label in 2107.

Figure 1.11: EU Energy Star Label [19]

1.7. Thesis's Structure

This thesis consists of seven chapters. The following is a summary :

Introduction and background – All information related to the background of the research, the description of the problem and the motivation behind initiating this work can be found in this section. Numbers and statistics are included to illustrate the size of the problem. It also contains a list of aims and objectives intended to be achieved. A sub-section is included to mention the contribution of lawmakers and legislators to assist in solving this issue. Finally, it contains this summary overview of all sections.

Energy Management Systems (EnMS) – A literature review chapter reviews a number of researches accomplished in this field. It describes and categorises a number of existing conventional, smart and modelling based energy management frameworks, and addresses the list of challenges faced in this field.

Technologies and Techniques – It provides a literature review related to all technologies and techniques that exist in this field and used to implement the related framework, including microservices, cloud computing, IoT concepts, data analytics and mining technologies, with a detailed description of various modelling analysis such as classification, regression, association, clustering, time series and their correspondent algorithms.

Integrated Scalable Smart Energy Management Framework – The proposed framework is illustrated and explained in detail. Components' overview is included, and some data workflows are demonstrated. All related functional requirements and quality factors or non-functional requirements are presented in this chapter.

Implementation and Evaluation – This chapter provides a detailed description of the implementation of the proposed I3SEM integrated framework in a selected household environment, including all experimental settings and assembled sensing and cameras systems. Various applications, software and algorithms and a data subset are evaluated in this chapter to provide a proof-of-concept for the fully implemented system, which is presented in the next chapter using the entire dataset.
Case Study – This chapter consists of a detailed explanation of two main experiments designed to implement the proposed framework on two sample household appliances: the refrigerator and the immersion water heater. It contains also the final results achieved to reduce energy consumption.
Conclusion and Future Plan – A conclusion, including the future work plan, is presented in this chapter.

1.8. Summary

As seen from previous figures and statistics, the world is moving towards reducing the use of the traditional resources to generate energy, and is attempting to answer the demand for energy by investing in renewables of all types; wind, solar PV, geothermal or hydropower. It has been recorded that this approach is moving faster in some countries, such as the UK, USA and Germany, but goes a bit slower in other countries such as China.

As shown in previous statistics figures, the household is considered one of the sectors where energy is consumed at the most. Statistics also reveal that efforts should be concentrated in these main directions: 1) encourage obtaining energy from renewable resources and 2), use the available energy most efficiently. Both directions have been supported by governments and governmental organisations by establishing and promulgating different laws and regulations to control the energy sector aiming at minimising the harms and negative effects on the environment, and by encouraging stakeholders to switch to renewable resources.

Due to the increasing energy awareness among the household occupants regarding the energy consumption impact on the environment and their budget. Moreover, the rapidly developing smart techniques which can be utilised almost in all environments. The opportunity to invent new systems to deal with this situation has arisen as never before, due to the massive efforts being taken to build, design and propose a number of Energy management EnMS systems which have a substantial influence on the energy efficiency, environment and economy.

As discussed in next chapter, all the reviewed frameworks cover different aspects of smart home management systems. They are pivoted around the same repeated idea of establishing a mesh network by adding hardware devices that have networking capabilities, together with having sensors and actuators, and finally designing a core management system to deal with the various network's nodes and data. Some of the proposed architectures and frameworks added more off-the-shelf units, such as microservices, simulation techniques, cloud solutions, emerging new techniques and technologies. However, all of them share a set of drawbacks and challenges; including lack of integrated architecture, lack of scalability, restrictions of the applicability on legacy and modern and smart environments, besides side issues related to the security, mobility management and proper stakeholders engagement and management.

The framework proposed in chapter 4, provides a comprehensive and solid architecture to bypass these downsides by offering a unique structure that divides the framework into three main zones, and presenting a number of relevant generic components, moreover, utilising appropriate state-of-art and modern paradigms, besides offering essential approaches related to the context-sensitive analysis, detection and probability generation, predictive analysis and the alerting messaging. The division of the framework into two main zones combined with a gate zone, improves several quality-driven aspects related to scalability, enhanced encapsulation, performance and interoperability. The client zone is implemented and physically resides in the household site, and external APIs providers. This approach shifts a remarkable part of processing power from the central processing units in the cloud to the client and offer more privacy and enhanced security to deal with sensitive data within the correspondent household without the need to transfer it to the cloud, where only filtered, anonymous constraints are processed centrally. The cloud gate stands in front of the cloud zone to facilitate a number of relevant characteristics related to security and performance, by authorising, load-balancing incoming requests and caching responses. The cloud zone is the core and central unit of the architecture comprised of all necessary components to process and analyse the gathered data and deliver responses to all upcoming requests. Shifting shared units from the client zone to the cloud has a direct impact on enhancing scalability, modularity and performance.

2. Energy Management Systems (EnMS)

2.1. Introduction

Energy Management System is a set of systematic procedures that aims to persistently improve energy management and reach higher energy efficiency degrees [20]. One of the major aims of any energy management system is reducing the amount of consumed energy while still offering the same or even an increased level of service and comfort, by providing a set of fully and semi-automatic tools, processes and techniques to maximise the consumers' engagement to continuously control the consumed energy. This approach can also be defined as energy efficiency. Reducing the amount of consumed energy lead not only to a decrease in cost, also to the reduction of greenhouse gas (GHG) emissions. In the industrial sector, applying energy management systems has the possibility to decrease energy consumption by 20% [21]. This percentage has a substantial environmental impact as it is known that the industry causes 26% of the global carbon dioxide emissions [21].

This chapter starts with a review of some meta operating systems and middleware designed and currently running in this field. This is followed by a description of the characteristics of large systems since the proposed SEMS is considered a large, extendable sophisticated platform capable of managing an unlimited number of devices, sensors, protocols, components and off-the-shelf modules, which matches certain characteristics such as compatibility, expandability, interoperability, integration and standardisation. The answer to the basic question related to the advantages of developing and maintaining an energy management system architecture is emphasised in section 2.4. Going a bit deeper and touching some data-related aspects including data challenge, data acquisition and data storage have been done in section 2.5. Having done all these pre-sections, a detailed review and grouping of a number of existing energy management frameworks designed in the field of energy management systems will be introduced in section 2.6. Groups are split into three main categories: high-level frameworks, low-level frameworks and modelling and data predictions frameworks. Then it will be followed by a review of a number of smart energy management systems frameworks which were grouped into four main categories: smart homes frameworks, IoT integration in smart buildings management systems, analytics-based approaches and modelling, simulation and forecasting concepts. Important to mention that reviewing EMS followed by SEMS was done for many reasons; firstly, to show the historical evolution of energy management systems by integrating the smart technologies, which provides an indirect justification why to design and propose a new smart framework, rather than proposing an ordinary EMS without smart technology. Secondly, to provide a base of comparison to mention the advantages and disadvantages so the newly proposed framework can attempt to avoid the disadvantages and concentrate on the positive parts of these systems and enhance them. Finally, to provide a reference for potential comparison between the proposed framework and the frameworks done so far, which assists in emphasising the uniqueness and the added-value of the proposed framework illustrated in chapter 4.

2.2. Review of Meta Operating Systems for Context-Aware Systems

Meta operating systems are established on top of an operating system allowing coordinating heterogeneous systems, devices, applications, processes and nodes to communicate with each other in real-time [22]. This section contains a brief overview of some well-known meta operating systems that may be considered to build large systems consisting of different nodes. By nature, smart energy management systems may vary in scale from a specific, capsulated system designed to serve a particular aim, to comprehensive systems serving multi-purposes, and offering a wide range of services. This large type of SEMS consists of different nodes and systems that work together and interact mutually based on the middleware architecture that manages the dependencies and organise the interactions among different nodes. In the literature, there are several middleware approaches for energy management systems used for distributed environments which include: **The power-aware middleware** - used for exchanging data on Mobile Ad hoc Networks called Transhumance [23] [24] [25]. **The hierarchical framework** – called Global Resource Adaptation through CoopEration (GRACE), offers integrations on different levels [26], as illustrated in Figure 2.1 [26]. **Middleware for Energy-awareness in Mobile Devices** – that consists of six different components aiming to manage the energy in mobile devices: resource manager, ML application classifier, power estimator, policy manager, messaging service, and processing engine.

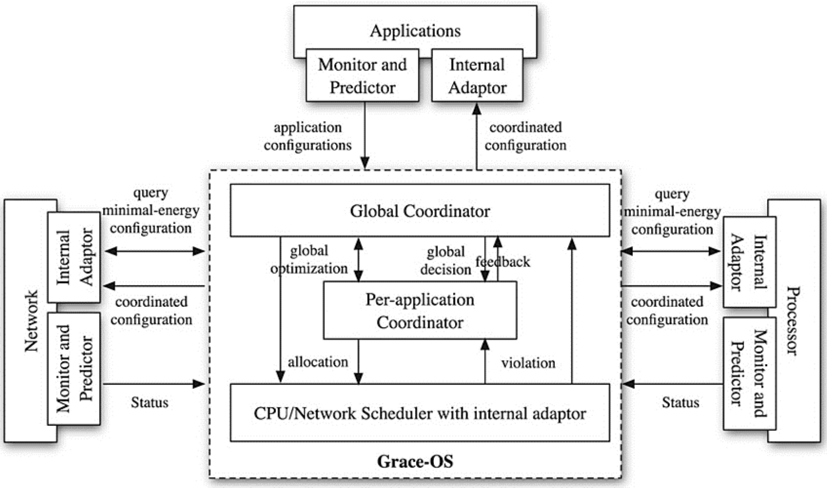

Applications
Monitor and Predictor
Internal Adaptor
application configurations
coordinated configuration
Network
Internal Adaptor
Monitor and Predictor
query minimal-energy configuration
coordinated configuration
Status
Global Coordinator
global optimization
global decision
feedback
Per-application Coordinator
allocation
violation
CPU/Network Scheduler with internal adaptor
Grace-OS
query minimal-energy configuration
coordinated configuration
Status
Processor
Internal Adaptor
Monitor and Predictor

Figure 2.1: The GRACE system. [26]

CasCap – context-aware power management system which consists of three major components: mobiles, internet, and clones (cloud services that can be used by mobiles to offload processing) [27]. **DYNAMO –** As illustrated in Figure 2.2 [28], a multi-layer architecture that aims to optimise energy consumption in mobile devices. It consists of four different levels: hardware, operating system, middleware and an application [28].

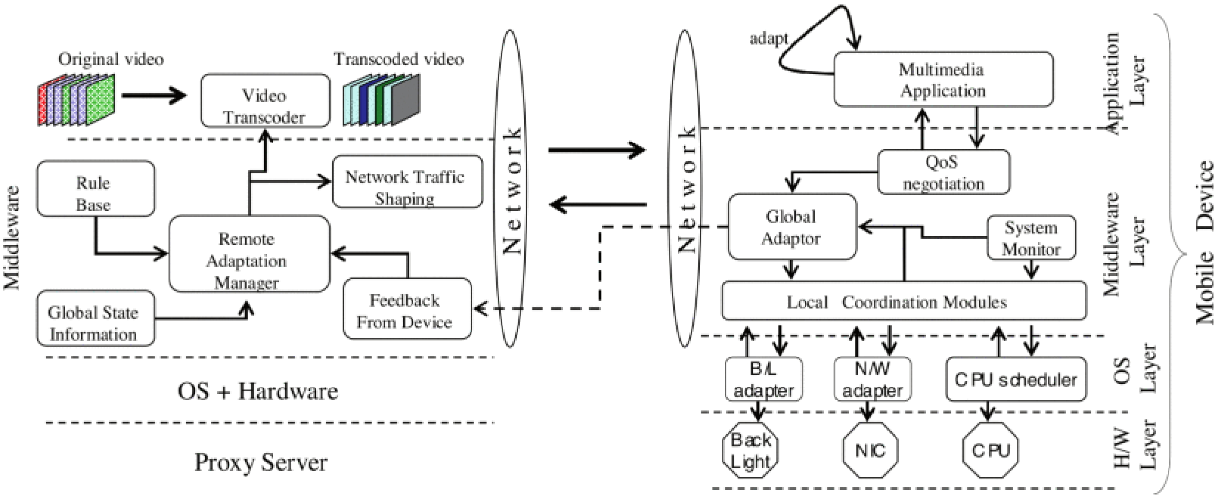


Figure 2.2: Architecture of the End-to-End Cross-Layer Adaptation Framework [28]

Power-Aware Reconfigurable Middleware (PARM) – assists in saving energy in low-power devices by reconfiguring the distribution of components and migrating them. The architecture is illustrated in Figure 2.3 [29]

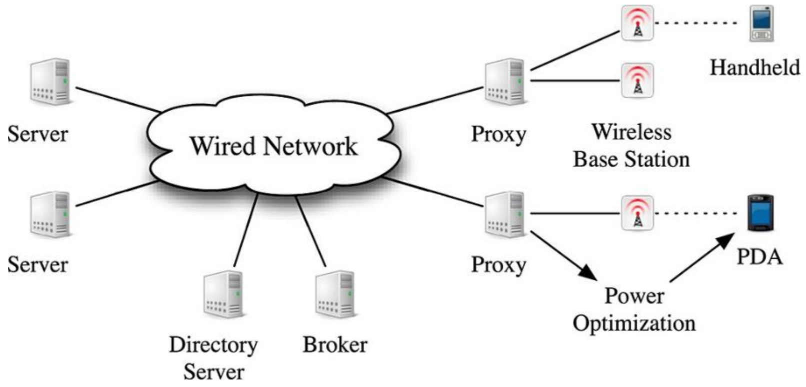


Figure 2.3: PARM Example Architecture [29]

Energy Centric Operating System (ECOSystem) – is a framework based on a new abstraction level of the energy currency that aims to manage the energy consumption on the operating system level by following three dimensions: time, tasks and devices. The framework is illustrated in Figure 2.4 [30]. **SANDMAN Middleware –** is an energy-efficient middleware based on the BASE platform which is a minimal communication middleware that follows the peer-to-peer principles. **SleepServer approach –** allows managing energy on the desktop personal computers by using various proxy servers and virtual local area networks.

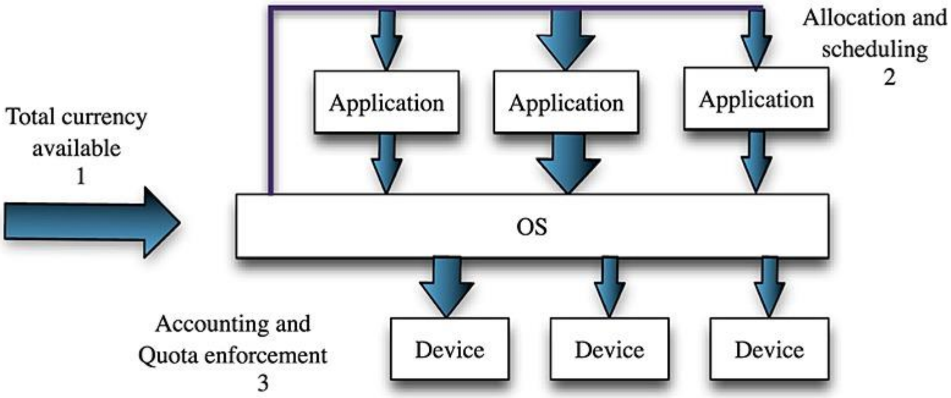

Total currency available 1
Allocation and scheduling 2
Application
Application
Application
OS
Accounting and Quota enforcement 3
Device
Device
Device

Figure 2.4: ECOSystem Framework [30]

GAIA – Meta-operating system [22] is an approach designed to support developing and running mobile software which runs in environments where users have the ability to interact with various devices concurrently. It consists of three main components: (1) Kernel: which contains all major management modules and a set of services such as context service, space repository. (2) Main application module. (3) All applications that run in the active space. The adding of the knowledge base has expanded the system in the way to describe entities and maintain cumulative ontologies. All different information related to the system nodes is stored in the central repository. The context is offered easily for both users and applications. Furthermore, the potential update of the system can be separated from the application itself. GAIA suffers from security issues because it does not have any basic modules dealing with security issues.

2.3. Characteristics of Successful Large Systems

Energy Management Systems are considered complex and large systems which are identified by the relationship among their various internal components. In this context, the architecture of the system plays an essential role when it comes to identifying the complexity and implementing related solutions. The environment where EnMS is implemented defines the exact deployment steps, which may vary from an environment to another. Every environment has its unique key factors and knowledge which should be considered during the design and implementation phases. Since the system is constructed in dynamically evolving and changing environments, continuous improvements, integrations, changing of functionalities and services are necessary to meet the evolving system needs. For this purpose, a large complex system should meet some essential characteristics to enable them to perform effectively, these are: being compatible, expandable or extendable, interoperable, integrable and standardissable, performant, reliable, scalable [31]. In the next sections, some of these characteristics are described in more details

2.3.1. Compatibility

This quality factor defines the ability of a system, application, module or component to continue functioning properly after updating any part (hardware or software) of the system. This goal can be achieved by defining the exact specification, functionality and boundaries of the software entity. Relying on interfaces and predefined standards may increase the compatibility of any software or architecture.

2.3.2. Expandability

Software is considered expandable when new functions can be added to the application without the need to re-implement all or a huge part of the software. Expanding the software may be needed to match changes in the type of use or user numbers. Also, it is maybe needed to match new functional or legal requirements. In the SEMS several changes may require a system to be expandable, for example: having new technologies, protocols, devices. Also, increasing the volume of the collected data from sensors, and increasing the number of connected devices.

2.3.3. Interoperability

This factor defines the capability of a system operating on different platforms, together with different external systems without suffering from any functional or performance restrictions. As indicated by the European Commission [32], ensuring the interoperability of energy management systems while integrating the demand response leads to activating the public-private partnership and increasing the effectiveness of implemented EMS. This factor plays an essential role in the SEMS approach because of the evolving systems and continuously changing platforms, techniques, and technologies in the field. Continuously new devices, applications, systems, products are invented, so the need for any entity to be interoperable is growing rapidly. Serving this purpose can be achieved by providing standardised architectures, applying interfaces, and providing clear boundaries with well-defined input/output interfaces.

2.3.4. Integration

The next step after ensuring the proper grade of interoperability is the ability of integration. This means how far applications or components are ready to perform collectively when deployed together within one system, or different systems within a larger system via predefined Application Programming Interfaces (APIs). Due to the nature of the EMS architectures which consist of a variety of systems, components, devices and modules, the integration is considered a must feature, where different layers of entities communicate with each other via APIs. Usually, the first layer consists of hardware (sensors, actuators, devices etc.), these have a direct connection via a middleware with gateway nodes, which is considered a master entity for further communication with services and entities in the cloud. Using middleware for communication is the main difference between integration and interoperability because integration requires using a middleware of any kind, however, interoperability requires data being exchanged among entities (components or systems) possibly in real-time. Moreover, interoperable systems do not only share data, they also processes and present those as their own data.

2.3.5. Standardisation

The major prerequisite to implementing and maintaining large EMS systems is following predefined standards. The need for standardisation comes from the fact that several software and hardware vendors deliver independent products that need to work together to deliver the desired system functionality. So, these vendors, who develop their products separated from each other must follow a set of rules and standards to allow integrability and interoperability. The absence of such standardisation may be considered as the main obstacle towards the rapid and effective development of any EMS. Commercial systems such as EMS will be deployed to a real-time platform, must match predefined characteristics because it is required to fulfil functional and non-functional requirements obligated by the market where it should be implemented. Mainly commercial systems must fulfil particular criteria; such as compatibility, flexibility, interoperability, data accuracy, and illustrates a high degree of expandability to enhance the scalability with proper resources management.

2.4. Advantages of Energy Management Systems Architectures

Previous sections described what criteria should be considered while designing a large and complex architecture for an EMS. However, designing a framework for EMS brings several additional advantages, such as;

- allowing decision-taking mechanisms and organisation services to consider the amount of energy consumed by various assets, or by different processes, to enable energy optimization on both local and global levels [33].
- Incorporating standards for exchanging, analysing and displaying energy data, and measuring the performance.
- Meeting the requirements of compatibility, expandability and interoperability to support further future developments and extensions.

2.5. General Data Challenges

According to Statista.com [34], the data and information volume which have been generated, caught, replicated and used worldwide in 2021 reached 79 Zettabytes (79 x 10^{21} Bytes). According to the latest forecasting scenarios, this number is subject to increase up to 181 Zettabytes (181 x 10^{21} Bytes) in 2025. Moreover, with developing and using new devices, sensors, and smart appliances, data volume and data variety and format is likely to accelerate even further. According to Heiss [35], several challenges came up with this enormous amount of data, in many fields such as business and industry, and science. In business, the challenge resides in merging data from various resources and protocols to extract useful information. This can be clearly seen while collecting and analysing customer data retrieved from social networks to derive some marketing strategies. In science, gathering the enormous amount of data in intense amounts, and ensuring the availability of this data for a limitless number of scientists all over the world is also considered a typical challenge for today. According to Heiss [35] in Germany, there are many platforms dedicated to dealing with the enormous amount of data obtained from various scientific fields, such as Data Life Cycle Labs (DLCL) [36] and Helmholtz Data Federation (HDF) [37]. Another challenge in the science field is archiving scientific data, or what is so-called (data preservation). This aspect becomes very important for that kind of data that cannot be repeatable, such as weather, or some geological measurements. In the next section data acquisition and data storage including the referencing to some data mining and machine learning techniques, will be discussed in detail.

2.5.1. Data Acquisition

Gathering data has moved from just collecting valuable data that is matching and serving the organisations objectives, towards gathering all kinds of data that represents the stakeholders’ interactions with an organisation. Enhancing user experience positively, and achieving reliable, robust and realistic strategic decisions in any business requires building a comprehensive knowledge generated by collecting and merging data from different origins; such as data collected from tracking users’ behaviours, the feedback which is given on the offered services, the mutual interactions and discussion occurred among users related to the given services or products, and finally, the users’ profile which fulfils the whole picture. Obviously, the huge amount of gathered data requires another scale of data mining techniques and algorithms to meet the rapidly increasing demand. These algorithms must be scalable and should be able to handle the rapidly changing variety and velocity, so they can accomplish the data transformation properly and efficiently before it can be processed by analytical tools. Many different techniques are used to acquire data from scalable household units where the energy consumption may vary continuously. Data is collected from the energy consumption and appliances’ status (ON/OFF), also it comes from different motion sensors, and heat sensors, also external data signals coming from weather forecasting, traffic and energy providers. Lately, together with the cloud-based

Platform-as-a-Service (PaaS) some methods and systems are developed to analyse real-time information sent by various sensors [38].

2.5.2. Data Storage

After acquiring and gathering the data, a platform for storing the data must be set up. This is an essential step before starting to process the data. There are too many storage platforms mentioned in the literature, however, this section will discuss and illustrate a comparative elaboration of them, pointing to the most suitable storage strategy for the proposed I3SEM framework. The industry offers many types of databases types; such as centralised databases, relational databases, cloud-based databases, NoSQL-databases, Object-Oriented (OO) databases, distributed-databases, graph-databases and operational databases, however, in this research, the focus will be on the most relevant types; relational and non-relational databases.

PostgreSQL, Oracle, MySQL, Microsoft SQL Server are among the list of relational database management systems available in the market. Per definition, relational databases are designed to deal with pre-defined data structures and schemed data. So based on this fact, storing structure-free or unstructured data, retrieved from different sources, in relational databases, is not appropriate. Relational databases perform efficiently and performantly due to pre-defined relationships and indexes. Lately, the shifting of such databases to the cloud has offered better scalability, availability and reliability, however, the lack of handling unstructured data put some restrictions on using this kind of database in the smart energy management systems field.

The alternative candidate is using non-relational databases [39]. This kind of database is known also as NoSQL databases or scheme-free databases. Basically, it saves the data as documents without a pre-defined structure which makes it perfect to deal with the various structure of data collected from different resources and hardware vendors, such as sensors, measurement meters, smart appliances, external data providers (weather forecasting, traffic, energy providers, ...). CouchDB, MongoDB and InfluxDB are among the list of the most known non-relational databases [40], [41]. Closer to the IoT hardware industry reveals that new types of equipment are developed and added at a very rapid rate, also the same type of hardware is developed and enhanced by adding new functions and evolving the existing ones. For instance, a sensor that collects and sends coordination were only sending latitude and longitude, however, recently the same sensor was enhanced to send further information along with Lat and Lon, such as the altitude, speed, or even the amount of covering satellites (affects the measurement's accuracy).

Besides the fact that non-relational databases can store unstructured data, research carried out by [42] has held a comparison between a relational database called PostgreSQL and non-relational database InfluxDB, has concluded that using both databases is equally performant when inserting the data, however, the disk usage overhead was significantly higher (58%) when using PostgreSQL database, same time InfluxDB has shown a reduction of the disk-usage size of 75%, which make it a more suitable candidate to deal with the enormous data volume expected in this field. The same research supports the conclusion that a non-relational database should be used for the proposed I3SEM framework.

2.6. Review of Selected Energy Management Systems Frameworks

Due to the fact that the primary energy resources, for instance, fossil fuels, are infinite supply and are harmful to the environment, and the fact that nuclear has raised a threat particularly after the Fukushima Daiichi disaster in Japan in 2011 [43], different and more sustainable solutions have been considered including the use of solar and wind energy. However, the required immense amount of base-load solar and wind energy harvesters stands as an obstacle to switching the efforts to obtain energy from primary resources to renewable ones. Therefore, the initial target of energy management is using the energy in both home and industrial sectors efficiently. To achieve this aim, many attempts have been made to design, implement and evaluate various energy management frameworks. These range from high-level frameworks to low levels detailed ones, some of which are considered in the following subsections [44].

2.6.1. High-Level Energy Management Frameworks

High-level frameworks give overall ideas about how energy should be managed without going into specific details of the implementation. These include an early integrated framework, which was introduced by Asare-Bediako et. al [45]. The framework's main characteristics are concentrated in three main aspects: Firstly – reviewing the energy management systems' concepts for housing customers. Secondly – inspecting the background of smart home energy management system technologies. Thirdly –

highlight the significant off-the-shelf components and provide a comparative analysis of different technological approaches. Moreover, it brings some light to some of the apprehensions and challenges represented in costs, confidentiality and deployment of smart technologies and future systems.

The proposed framework explains many issues that should be taken into consideration while building sustainable home energy management systems and how they interact. The first approach is built upon providing an overview of a network of interconnected items which can be modified according to needs. Figure 2.5 [45] illustrates a joined framework for developing upcoming home energy management systems. The building's main characteristics, such as building's type, construction year and orientation, are considered by the framework. Moreover, the nature of the building's occupants is also essential; they may be families, single persons or even students or older people. Secondly, the way how the home energy management system is deployed and installed defines its integration's grade and being sustainable. Thirdly, any previous installations, the existence of sensors [46], smart appliances, distribution sensors and the way how they are connected and integrated into the entire facility. Finally, the interoperability, in other words, the ICT software must support various devices, technologies produced by different manufacturers.

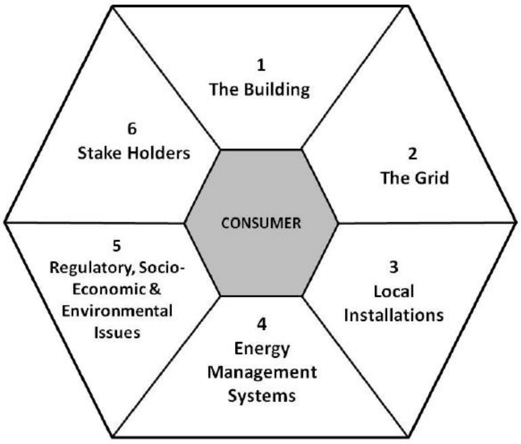


Figure 2.5: Integrated framework for future HEMS [45]

In addition, such frameworks provide a high-level view; they are highly descriptive, i.e. they show what to do, not how to do it.

Building and establishing an effective framework is the first step towards achieving the goals of an Energy Management System (EnMS). The main focus of such a framework is to manage ongoing energy consumption, identifying the chances to implement energy management technologies, especially the choices that do not necessitate massive investments. EnMS assists in establishing continuously running processes. To obtain the required results, all energy efficiency improvements efforts must not be made on a one-time basis, rather they must be figured and applied in the process of persistent enhancement and perfection. Moreover, even the recently optimised systems may become outdated because of the rapidly changing constants in this field, represented by the continuously improving technology, changing political situations and variety of the supply-and-demand forces in the energy field [47].

The implementation of an EnMS may follow either an established standard or may be derived from a custom approach. One of the well-known and accepted standards in this field is the ISO 50001 (This has been explained in a bit more detail in section Law and Regulations). Regardless of the applied approach, the key focus of any EnMS is the involvement and the continued commitment of all participants, including management, consumers and system users. The continued involvement of all participants in the EnMS increases the motivation and ultimately lead to the effective functioning of the EnMS. Regardless of the chosen standard, ISO 50001 or a customised one, the elementary EnMS progression always follows the (Plan → Do → Check → Act) approach (see Figure 2.6) [48].

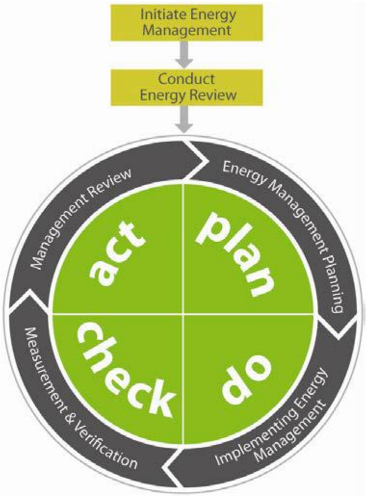

Initiate Energy Management
Conduct Energy Review
Energy Management Planning
Implementing Energy Management
Measurement & Verification
Management Review
plan
do
check
act

Figure 2.6: Plan-Do-Check-Act framework [48]

2.6.2. Low-Level Frameworks

Most of the current modern management systems are built on embedded systems with interfaces interacting with various smart home components, different heating and energy systems, demand-side management systems, and at the same time to cloud-based services such as weather forecast portals and electricity price engines. These kinds of systems do face some challenges, such as: To achieve a high grade of integration, these systems must be adjustable and can be easily installed in the industry or home sectors. It also should consume a low amount of energy and has low installation and integration costs. In reality, it is observed that interoperability among developed systems is very poor; this is due to the lack of common standards. Moreover, unfortunately, the future picture does not look promising; the incompatibility grade will become worse with all those newly developed proprietary devices. Some attempts have been made to address some of the above issues such as the open-source solution of a framework which is proposed by a group of researchers belonging to the Fraunhofer Institute in Germany [49].

This is intended to be available for a personal and organisational level. There is a similarity between this approach and other existing frameworks, such as Niagara Frameworks, Quivicon Framework, HomeOS, Apache Cocoon, OpenHAB [49]. However, the main difference resides in the fact that those frameworks are not fully open-sourced and are mostly concentrated on the smart home area.

The target of the proposed framework Open Gateway for Energy Management Alliance (OGEMA 2.0) is providing software and hardware-independent platform to form a basis for various units and interfaces in a high modularity structure. The framework's user's interaction is granted by offering several interactive interfaces and GUIs. It consists mainly of a JAVA-based middleware operating in a VM, and some fundamental components including an operating system, data storage and interfaces support. OGEMA 2.0 offers different security levels, user management modules and permission management as off-shelf components. Figure 2.7 [49] illustrates the proposed software architecture of the Energy Management Gateway (EMG) utilising OGEMA 2.0

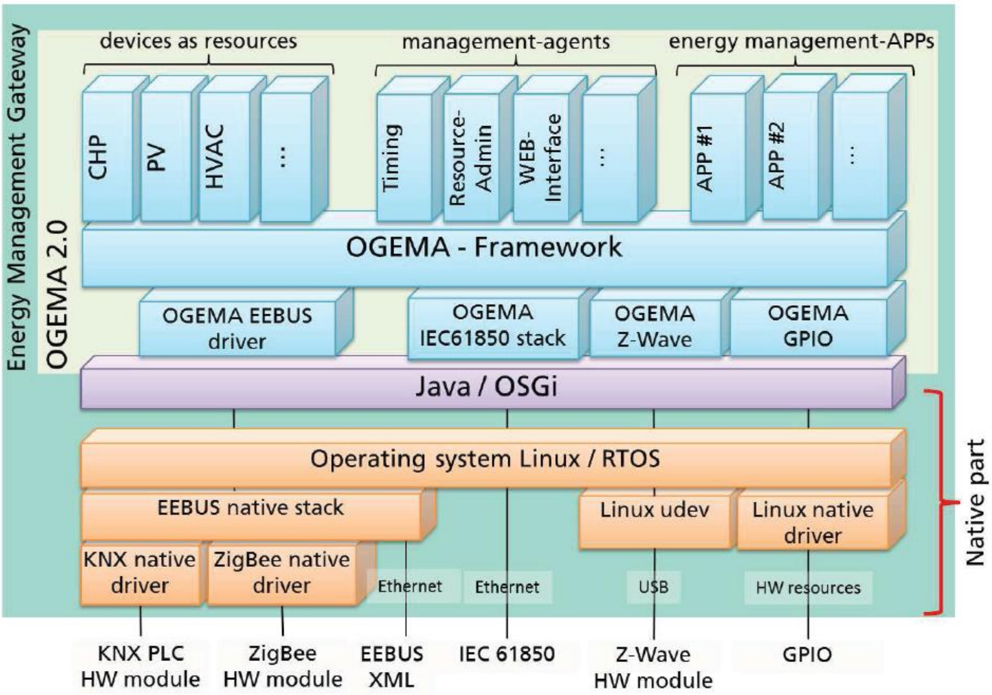


Figure 2.7: Energy Management Gateway (EMG) Architecture using OGEMA 2.0 [49]

Another low-level framework is the improved energy management framework, used to raise energy consciousness (Figure 2.8 [50]). It is designed to be used in advanced and innovative Industrial Energy Management Systems (IEMS). Energy-related data will be collected from various locations of the factory; then it gets combined with the enterprise dataset to sharpen the overall image, and ultimately enable further optimisations and emendations [50].

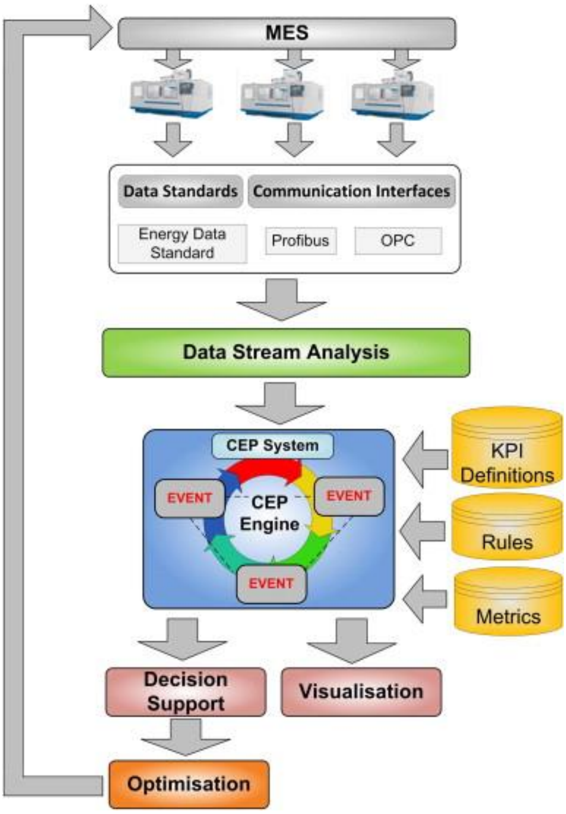


Figure 2.8: Industrial Energy Management Systems (IEMS) [50]

This framework can be used by the manufacturers who use a manufacturing execution system (MES), which is responsible for delivering real-time monitoring and production plans, however it does not deliver any output related to the energy consumption during the production's various processes. The framework enables integrating the consumed energy's data in the overall scene by defining energy data standards and integrating some existing communication interfaces such as process field bus (Profibus) which is a master-slave structured protocol invented in the '90s to respond to the industrial communication requirements. Moreover, the Open Platform Communications (OPC), which is a set of specifications and standards applied in industrial communications to establish a communication bridge among devices of different vendors [51]. The data stream analysis makes a real-time data streaming, which is usually combined with a massive amount of data, possible by standardising the transferred data format and applying data reduction to extract the targeted figures from the vast data set, fast and effectively.

The complex event processing (CEP) engine, which analyses events all the time trying to find patterns in real-time and suggests a response to them as soon as possible. An incident is classified as a data instance with a changed status, such as a machine that started or stopped running. Moreover, it offers some other features, such as standardised energy data, which increase the reliability and consistency of most of the current energy data sets. The CEP system is provided with three primary feedback resources: First, the industry's energy key performance indicators (KPIs) such as energy cuts, and the duration average, maximum allowable costs. Second, the rules cover all related production rules such as quality, speed, involved workers, and third the metrics which are similar to the KPIs, however, they differ in the way that it covers all business areas, whereas KPI is specific for one business field. Metrics example is sales revenue, customer loyalty, productivity ratios etc.

The proposed framework intends to deliver performance metrics and energy consumption figures visualised on a daily and hourly basis, to allow decision-makers to adjust their production processes in a way to avoid high-energy demanding production activities during the peak load, or taking advantage of the energy generated solar plants during the day-light. In the same way, some production processes that generate heat can be aligned before other processes which require heat building an optimal energy cascading approach.

2.6.3. Modelling and Data Prediction Frameworks

According to Kucuksari [52], many studies have been presented using modelling renewable energy systems and their simulations with and without grid connections. Some of the studies focused on various control algorithms and some others concentrated on capacity planning and optimisation of system sizing by using different simulation approaches. There was some work done on operational cost minimisation of Photovoltaic (PV)-wind-grid connected system by forecasting generation amount and heuristic optimisation. Mathematical models of each system and are employed linear programming along with heuristics. The amount of daily power generation is estimated, and energy allocation is performed based on cost minimisation. A policy-based two-level distributed simulation for energy management has never been explicitly addressed.

The described framework is illustrated in Figure 2.9 [52]; it presents two levels of modelling layers which illustrate an average household with a PV panel, storage unit and a grid connection. The framework's primary goal is providing the household units with the required direct current (DC) without suffering interruptions and with keeping the operational costs at their minimum level. Both levels are called high-level and low-level.

System dynamics (SD) is used to develop a high-level simulation model. SD is a computer-based approach that assists in understanding the irregular behaviour of complicated systems. This high-level, which is implemented by AnyLogic software, is designed to provide the system with operational decisions while involving policies, rules or restrictions to control all the components of the low-level. The taken decisions could be for instance when to supply the load from PV, or when to start charging or discharging the storage unit (battery), or even when to start buying energy from the grid to face any possible increasing demand. The low level is a representation of electrical circuits including PV, storage unit (battery), some household energy-consuming appliances, and grid. It is implemented using Matlab Simulink. The communication and interaction between both levels is performed via a set of SOAP-based Web-Services technologies.

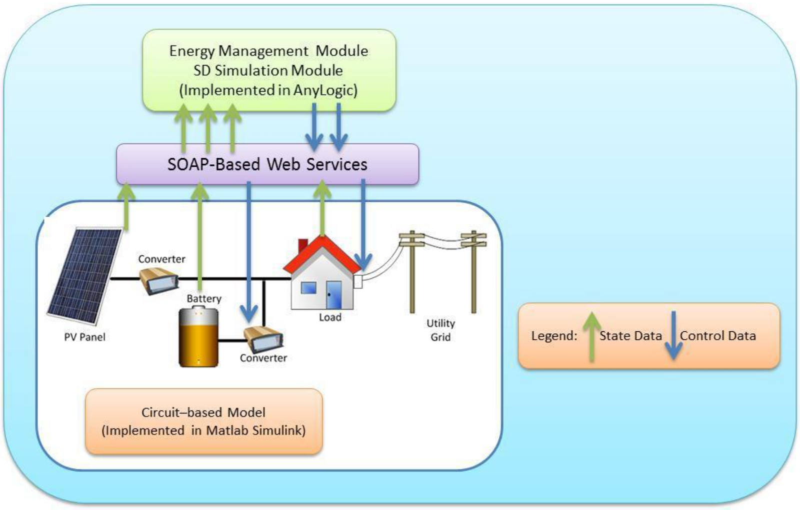


Figure 2.9: The proposed two-level modelling framework [52]

Another framework that is based on data prediction and modelling is called model predictive control (MPC). It is an enhanced technique of process control (PC) that is used to steer a process while fulfilling a set of rules and limitations. This approach has been applied widely in chemical plants and oil refineries starting from the '80s of the last century. A proposed (MPC) framework [53] which is shown in Figure 2.10 [53], consists of several parts such as a virtual building that simulates the real one, a predicted set of data using the MPC framework, savings and control variables. All these parts are connected to the outside world via an external interface.

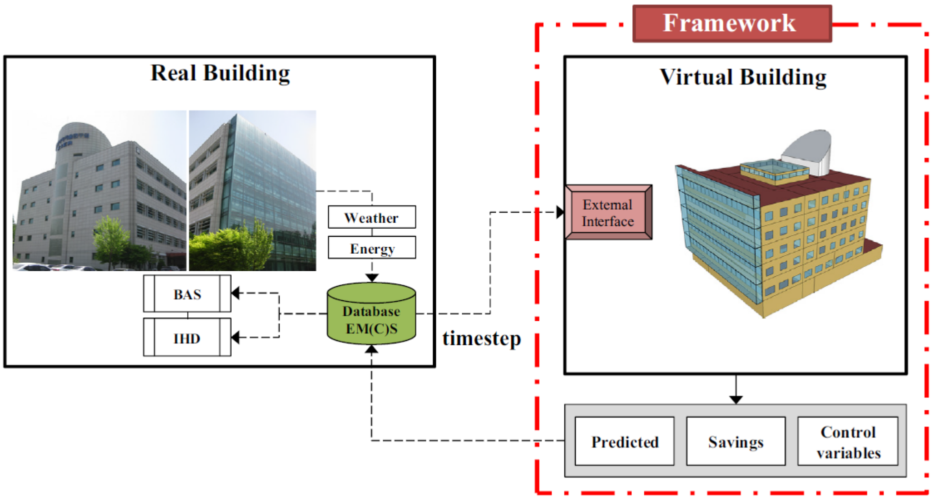


Figure 2.10: Model Predictive Control Framework [53]

2.6.4. Summary

It is clear that all previous energy management frameworks are aiming at one main target: achieving the highest level of energy efficiency. Frameworks attempt to achieve this goal by following different approaches. However, every one of these follows a set of rules which are specific to the framework itself. Moreover, it acts either as a stand-alone island by not communicating with the outside world, and thus not benefiting from the vast capabilities offered by available technologies, or lacks having proper sensors and actuators to gather, process and make proper decisions. To overcome this shortage, these frameworks and systems have been evolved by making them smart. The following section will discuss some of these smart frameworks and systems.

2.7. Intelligent Energy Management Systems

The continuous increase in energy cost and the rapidly growing energy demand, as well as the negative environmental impact, have created an urgent need to go to the next level of managing energy resources and consumption by applying smart energy management approaches and technologies. Such approaches enable the measuring, monitoring, controlling and ultimately saving energy by making real-time decisions based on the collected data while still being able to maintain the same level of quality of life and energy demand. The newly developed technologies such as IoT, Big Data concepts, cloud computing and ML techniques have played a massive role to switch from traditional energy management systems to smart ones. Moreover, the rapid development of communication protocols such as ZigBee and Z-Wave, and their related advanced metering infrastructures such as sensors and hardware have also pushed the use of smart EMS forward and brought it to the next level in residential, industrial and commercial areas. This section will explore some applied areas by reviewing recent research work in order to analyse the nature of these smart systems and their direct and indirect impact on the surrounding zones.

2.7.1. Smart Home Frameworks

As discussed in section 1.2.4, worldwide household electricity consumption has reached unprecedented levels. The illustration in Figure 1.6 shows that the energy consumption rate within the residential area is very high, for instance in the U.S. this rate reaches a total of 20% of the total energy consumption recorded in 2017, in numbers, this means approximately 19,54 quadrillion British thermal units. Considering this fact, any effort invested in reducing the consumed energy will significantly affect the overall energy consumption and bring society one step closer toward efficient energy usage. Furthermore, the evolving communication protocols, infrastructure, storage units and home area networks, led toward building smart home energy management systems which mainly consist of advanced technological infrastructures coupled with renewable energy sources and storage units. Figure 2.11 [54] illustrates an example of such systems.

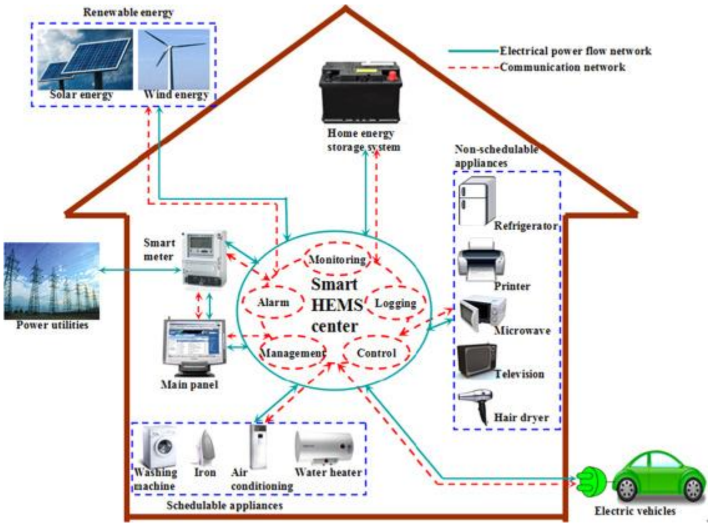


Figure 2.11: Home Energy Management System Architecture [54]

As proposed by [54] a smart home energy management system (SHEMS), as illustrated in Figure 2.11, consists of many units; the controller is centralised and has the task to provide the control possibilities and monitoring facilities. In the system, the energy consumption figures consumed by both appliance types; schedulable and none-schedulable appliances, can be tracked and recorded to offer an overview of the consumption rates and to deliver needed electricity optimally. The smart-meter is playing a gateway role between power utilities and smart home appliances; it is capable of receiving the demand response signal sent by the local power utility to inform the customer about possible modifications of the price or electricity consumption rates. Also, it assists in controlling home appliances to implement the residential demand response which is a new method that aims to improve the energy consumption efficiency by time-shifting the appliances running phases based on the local utility's condition. The electrical car is considered a schedulable energy consumer in the system; thus, it brings a high degree of stability to the energy system by avoiding consuming energy on peak loads. Having a solar source of energy, mainly obtained from photovoltaic PV, and proper storage facilities decreases the dependence of the system on the local utility and enhance the management capabilities of the HEMS due to having various energy sources. The proposed framework offers various functionalities such as controlling, monitoring, logging, alarming and management.

The previous framework was built on the assumption that local utilities do support smart meters devices and are capable of dealing with demand response signals. It also requires financial investment in the deployment of smart HEMS by purchasing and installing additional appliances. The security aspects are not mentioned in the design, so it does consider the strategies applied to protect and regulate the access of the local mesh network from outside. Moreover, there is no detailed explanation of the required solar PV technologies, or the storage units' limitations and the amount of lost energy during operating phases.

As a result, several new energy management frameworks have appeared recently in literature such as the Sense-Think-Act Framework introduced by [55]. The framework includes a number of core concepts of smart building management systems which observes, measure and collect data from the mesh network's various component, and then it delivers the collected data selectively to a central data management unit, where it gets processed and analysed to produce the required rule-based decisions. Finally, perform a suitable action on the controllable building appliances [55]. In fact, the paper introduced a streamlined framework to implement the Sense-Think-Act design concept, in which the middleware represents the Sense (S) part, the model-based optimisation methodology represents the Think (T) part, and finally, the Action (A) is made by the actuation and evaluation components. Thus, three main components are considered: (***S***ense) The data flow collected from different end-units (sensors, weather forecasts, traffic stations, ...). (***Think***) Represents the rules-set-based methodology to produce effective decisions — finally, the (***Act***) which represents the assessment and operating units. The hosting platform middleware is considered one of the most critical components of the framework (Figure 2.12 [55]). This middleware plays an essential role to reduce deployment costs, streamlining the data flow across building appliances, sensors and actuators.

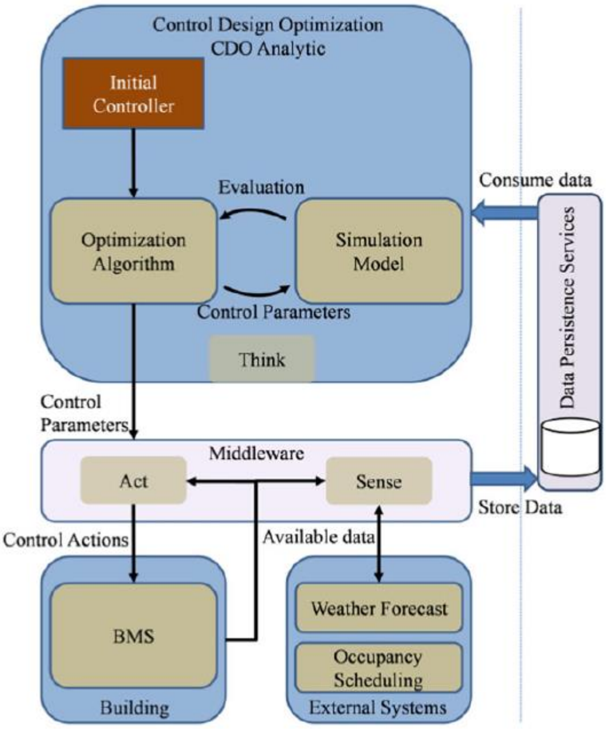


Figure 2.12: The proposed STA framework [55]

According to this framework, the (**T**)hink task is addressed using the following mathematical equation:

$$x_{k+1} = m(x_k, u_k, d_k)$$

where m is a vector function, x_k contains observable system states at the time k, u_k represent the control actions at time k, and d_k is contains the measured disturbances influencing the system (e.g., weather circumstances) at k time. Thus, an accuracy future states x_{k+1}can be estimated.

The experimental verification has been performed on the AC appliance installed in the offices of the Technical University of Crete. The building suffers from poor isolation and low airtightness. Prior to the project, every AC was managed manually in solo mode. The experiment started by adding sensors to measure the current energy consumption rates and the actual parameters, including ambient temperature, humidity, presence detection, windows and doors contact. Moreover, actuation capabilities were installed to control the AC units by switching them on/off, setting the temperature between [18 – 29] ° C, running mode (hot or cold) and fan speed [zero - hundred] %.

The conventional Building Energy Management Systems (BEMS) depend on a pre-configured rules-set that consider pre-defined and assumed constraints and act according to it. For instance, the expected total and the average number of occupants in the building, the average value of various weather elements such as temperature, humidity, or the price of energy in the region, will be fed to the system once or updated on regular basis to allow the system to make decisions that mostly achieve the challenging equation: how to reach the most efficient energy consumption while providing the targeted user's comfort. However, according to the concept of STA, reaching this golden rule requires applying the following ways. Firstly, the separation of energy-saving from user comfort is untimely. In other words, performance indexes, smooth building operation should be considered while taking the user comfort into account. Increasing user comfort may result in consuming more energy after establishing a BEMS system. Second, fully exploiting the intelligent BEMS design requires having a set of fault detections and assessments units to react instantly to users' disturbances. Finally, it is essential to have a hosting platform within the scene to allow mutual communication among sensors, appliances and various control units to ensure the proper flow of data in both directions. In addition, this approach does not achieve the maximum benefit during the edge-cases, for instance, when the weather suddenly changes, or several occupants leave the building, or even when new energy providers reach the region offering better or clean-energy-based energy plans. Substitutable, there is a need for a system that continuously measures, detects and make assumptions to decide for the best operational conditions to achieve the golden rule efficient energy consumption while matching the desired comfort.

A number of industrial and commercial online services such as If This Then That (IFTTT), Capterra, Automate.io, Zapier were invented in the latest decades to offer platforms for mutual communication among services that exist on the Internet. It is an automation platform for busy people because it allows automatically moves information between web applications letting individuals or businesses focus on more important work. IFTTT utilises five different concepts: Services or Channels, Triggers, Actions, Applets and Ingredients. Services are the basic blocks; they mainly describe a series of data from a certain web service such as YouTube or eBay. Services can also describe actions controlled with certain APIs, like SMS. Sometimes, they can represent information in terms of weather or stocks. Each service has a particular set of triggers and actions. Triggers are the (this) part of an applet. In other words, they are the items that trigger the action. For example, receiving a notification based on a keyword or phrase from an RSS feed. Actions are the (that) part of an applet that represents the output resulting from the input of the trigger. Applets or recipes are the predicates made from Triggers and Actions. For example, if the user likes a picture on Instagram (trigger), an IFTTT app can send the photo to his/her Dropbox account (action). Finally, Ingredients are basic data available from a trigger—from the email trigger, for example, subject, body, attachment, received date, and sender's address.

This approach sounds promising and offers endless opportunities to connect unlimited numbers of communication among providers on the Internet, however, it suffers from a number of drawbacks; firstly, it requires having pre-defined, and pre-negotiated agreements and protocols with the services providers to implement the required API functions and pre-defined data nodes, which brings some restrictions and efforts to add new bridges from new service providers, or modify the current APIs if necessary. Secondly, it does not define standard and unified data structures that should be followed by service providers, on contrary, these online services, are consuming what other service providers offer. Thirdly, it misses an overall security concept to deal with potential threats. Fourthly, it lacks the scalability and expandability to be rolled out covering micro solutions not related to known service providers, such as integrating self-developed components residing in a private network. Services must be public and reachable over the World Wide Web.

As will be seen in chapter 4, the proposed framework attempts to address these drawbacks by offering standard and unified data structures units to transmit data among various components, without a need for additional programming efforts, and without a need for previous negotiations or shake-hands agreements. It also offers on-the-spot solutions to deal with the most relevant security issues such as mobility management, bundle-node-of-attack to deal with the fact that nodes inside the mesh network can be exposed and be part of other networks, such as a mobile phone which can be part of the mesh network and same time part of the cellular network. Moreover, it can be physically stolen or hijacked. Finally, the fact that service providers must be public and reachable via WWW does not support the nature of energy management systems implemented within the household sector, because this kind of system depends on having micro and specific solutions located within a private network.

2.7.2. IoT Integration in Smart Buildings Management Systems

Integrating IoT based applications in all kinds of smart *things* such as buildings, organisations, cities, e-health, asset management, considered a major characteristic of the modern industrial witnessed today. The proper implementation of IoT based smart buildings' systems requires fulfilling some crucial non-functional requirements such as maintaining the same level of comfort, achieving a high level of usability and user-friendliness, matching the standard security standards. However, many technical challenges come up with applying IoT applications in the smart home area. A paper introduced by [56] provides a review of these benefits and challenges. The idea is to apply IoT concepts taken from different smart cities areas, namely, energy, water, mobility, constructions and authority, as a down-sized version in commercial buildings. In fact, not all the parts/units of a commercial building are suitable to apply IoT concepts, only those areas with high energy consumptions rates should be considered, such as server rooms, office spaces with the considerable need of lighting, HVAC rooms, cooling and heating appliances. Figure 2.13 [56] illustrates a pictorial view of an IoT environment with sensors farm, Building Management Systems (BMS), networks and various cloud services. As indicated in the picture, the main focus currently is on electrical energy consuming appliances; however, in the future, all other kinds of energy such as renewables and natural gas will be considered.

A transformation process to convert a conventional BMS into an IoT-enabled BMS should cover all existing BMS aspects or features. Starting from the *Scope* – which should be extended to support aggregated systems (e.g., energy, surveillance, alarm systems, etc.). *Sensors* – change the current specific sensors to more detailed ones measuring and tracking humidity, CO_2, temperature and motions. *Protocols* – play an essential role in establishing the core communication. Thus, these should be converted from Plethora to IP/IoT based protocols. *Architecture* – transforming the architecture from standalone and closed to open and networked shape. Also changing the *access* type from closed/local to open and remotely. Finally, the *security* must be enhanced to match the new changes by moving it from the basic level to an advanced level covering all known security aspects.

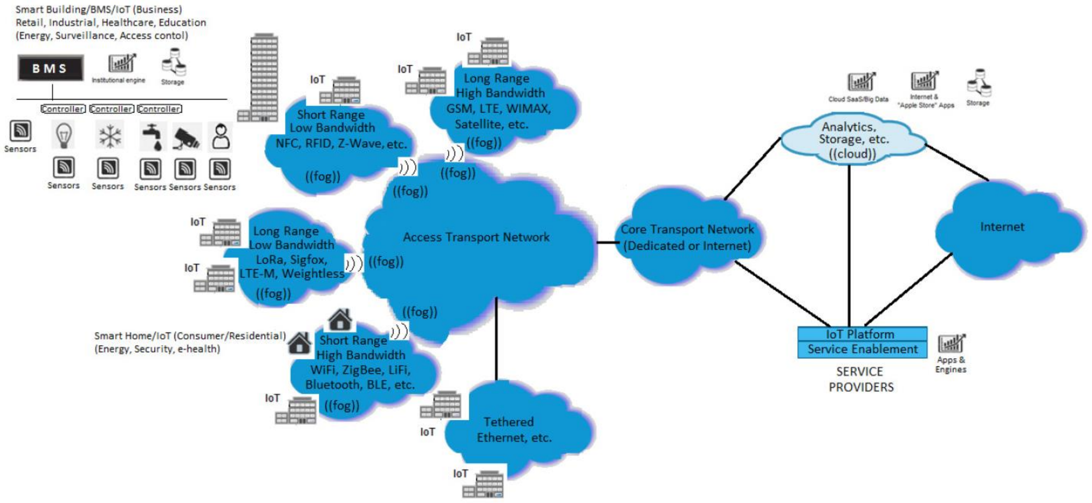


Figure 2.13: Sensors, BMSs, Networks, and Cloud Services in an IoT Environment [56]

The paper introduced the usage of Power over Ethernet (PoE) technology to fulfil most of the previously mentioned changes and enhancements. The basic idea of PoE is transmitting DC power among data conductors by applying a specific voltage on unit pairs, without having any interference between data packets and the applied power signals. This technology consists of two main components: First, Power Sourcing Equipment (PSE) – which is a device that injects power into the PoE environment. Second, Power Device (PD) – which represents any device operated by a PSE device.

2.7.3. Analytics-Based Approaches

Smart home energy management approaches which are applying Big Data analytics and IoT, were lately considered in a vast number of researches. The aim is to build a smart HEMS to achieve the previously mentioned sensitive and challenging equation by cutting the costs while still meeting energy demand [3]. The proposed system is utilised by interfacing each home appliance with a data acquisition module addressed by a unique IP-Address within the established mesh network. It collects the data and transfers it to central storage equipped with off-the-rack Business Intelligence and Big Data technologies. IoT, Big Data and Business Intelligence platform (BI) technologies offer a solution to address the challenges represented by the sheer quantity of collected data and the efforts to process and analyse it while applying the smart HEMS in various scales from one household to an entire community.

As seen in Figure 2.14, the system is prototyped in the lab for Heating Ventilation and Air Conditioning (HVAC) appliances as a case study. The proposed architecture consists of several subsystems: first, hardware architecture such as sensors and actuators, high-end microcontrollers, servers. Second, software architecture, which includes data acquisition modules, client application modules and the middleware module. The middleware module consists of a messaging protocol server based on the publish-subscribe approach called Message Queuing Telemetry Transport (MQTT) server, a database server, an analytics engine, and a webserver.

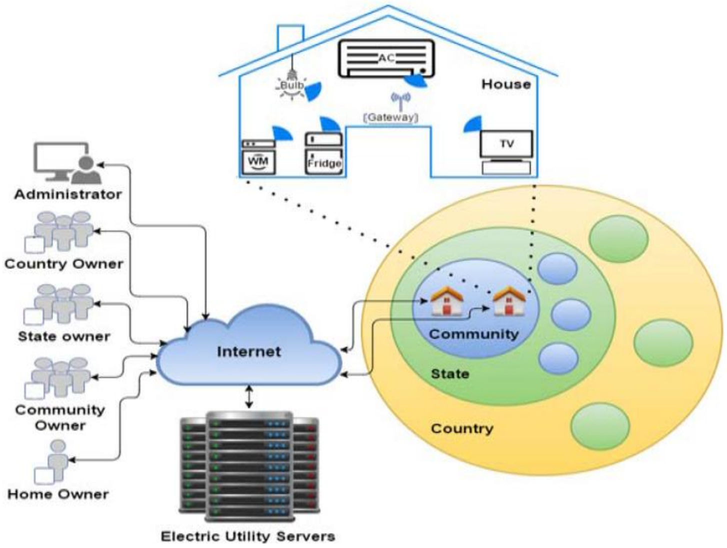


Figure 2.14: Analytics-based HEMS Architecture

Constructing the proposed Wireless Sensor Network (WSN) using the client-server paradigm via one of the commonly used smart home protocols such as ZigBee brings some limitations. These can be summarized in two points: First, the shortage of reliability because of the use of the client-server paradigm, which raises the possibility of losing the data in case of any system failure. Second, the need to build a data integration bridge between the smart home protocol, in this case, ZigBee, and the TCP-IP when integrating the stand-alone household smart HEMS with other units to build a community of homes. Moreover, utilising the GSM/GPRS networks in the WSN networks may offer more possibilities to control, monitor and schedule IoT appliances within the household. Collecting the massive amount of data from WSN centrally is a known approach that has been done many times, however analysing and managing this data efficiently by applying Big Data concepts, and obtaining comprehensive understandings remains a challenging mission.

2.7.4. Modelling, Simulation and Forecasting Concepts

Over the years, modelling, simulation and forecasting techniques have been used in many different applications. Recently, these have been applied for efficient energy management systems. [57] has proposed a simulation-driven and IoT-based smart home application. The purpose is to demonstrate the energy efficient IoT based smart home using kitchen appliances, heating and cooling devices, motion sensors with surveillance cameras, and coupled lighting and HVAC control systems. The whole proposed system will be managed via a mobile App from anywhere; in other words, it should offer a high degree of mobility. The Multiphysics simulations were carried out using ANSYS kitchen products. Figure 2.15 [57] illustrates a scenario that begins when the motion detector recognises a human being entering the kitchen, this information is transmitted to the home energy management system to perform some actions, such as turning the light on and switching the HVAC device ON/OFF.

Using ANSYS gives a vast opportunity to simulate the system as a whole or some parts of it under various circumstances and operational conditions, such as thermal changes, fluid forces, and even electromagnetic radiation. This ability has been accomplished in this case study by adding some stress testing for the antenna used in the smart LED and different other IoT devices.

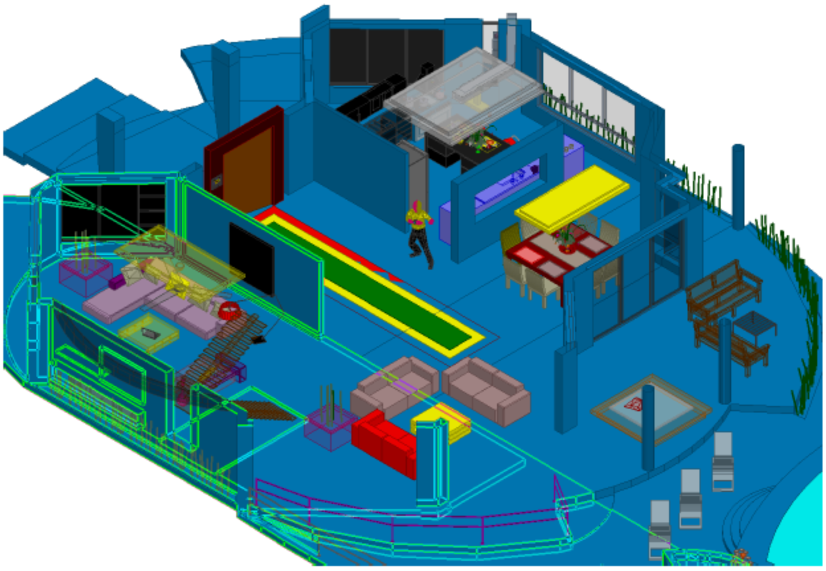


Figure 2.15: Simulated Smart Home Model using ANSYS products [57]

Other systems such as SHEMS introduced by [58] are based on the Bluetooth Low Energy (BLE) which is improved by offering an Artificial Neural Network (ANN) to overcome the limitations faced by customers who are not able to use some incentives provided by smart metering systems such as time-of-use, real-time pricing and demand response programs, to decrease the consumption of energy within peak-hours. The proposed mechanism focuses on forecasting and predicting the energy consumption at various times during the day and on different days during the week and provide a necessary ANN configuration to accomplish the required target. Basically, a combination of ANN and BLE will be used to predict energy consumption; this approach allows taking decisions not only based on the actual situation. moreover takes the likely short-term energy consumption progress. Figure 2.16 [58] depicts the core concept of the system.

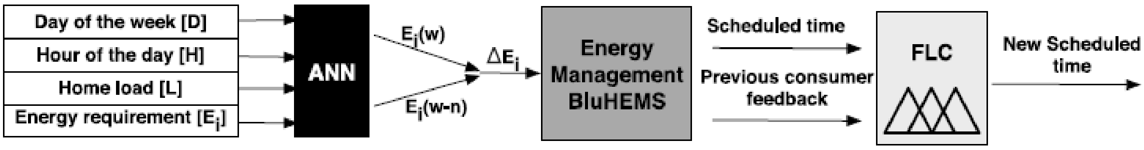


Figure 2.16: Proposed EMS Architecture [58]

The fact that the evaluation has been performed using a simulation campaign done via the Network Simulation Version-2 (NS-2), without real-time data taken from a real-world experiment, opens avenues for possible future work to apply the proposed architecture in a real-world example for a period of time to ensure getting realistic results.

2.7.5. Summary

All the reviewed frameworks cover different aspects of smart home management systems. They are pivoted around the main repeated idea of establishing a mesh network by adding hardware devices that have networking capabilities, together with having sensors and actuators, and finally designing a core management system to deal with the various network's nodes and data. Some of the proposed architectures and frameworks added more off-the-shelf units, such as microservices, simulation techniques, cloud solutions, emerging new techniques, and technologies. However, all of them share a set of drawbacks and challenges; these can be summarized as follows: Firstly, lack of integrated architectures – Most of the reviewed architectures lack the integrated nature that enables combing several functions, areas and sub-systems usually considered separately under one umbrella to achieve one main goal and acting as one entity. Secondly, the lack of standards and unified data structures; most of the provided solutions are following their own vendor's standards and data architecture. This has resulted in complicating the overall collaboration and interworking in the industry and prevent small vendors from contributing to the industry by developing specific aggregable units by following a certain standard and unified data architecture. Thirdly, the inconsistencies between the applicability on legacy and modern and smart environments; Implementing IoT-based BMS may not always be successful in all environments. It is not guaranteed that it works in every legacy or new BMSs during some shortages related to the lack of core components, the nature of the BMS and the level of the stakeholder contribution to make it happen. Fourthly, the lack of overall security concepts including general IoT security, and protection against potential bundled points attacks. An aggregated network, such as the energy management systems, brings enormous challenges to protect the network nodes in hardware and software terms. The protection must be extended to cover the physical stealing of devices and use it as a bridge to access the rest of the mesh network. Fifthly, lack of mobility management – some sensing, controlling or monitoring nodes (for example, mobile phones) may physically be located inside another foreign network; this may bring serious threat to the whole system and open severe security holes. And finally, the lack of appeal to stakeholders – The stakeholders in this field can be divided into four groups: governmental agencies, energy providers, home appliances' manufacturers and household occupants. None of the reviewed frameworks addresses proper communication channels to ensure a high degree of involvement of these stakeholders' groups. The organisational stakeholders represented by governmental agencies, energy providers or appliances' manufacturers may not feel the need to adopt this approach because of having other priorities or having some political conflicts and different points of view, or they are missing the motivation to invest in this area. Moreover, the individual stakeholders - the household occupants - may not be interested in applying and operating the system, especially those who are renting the property with an all-bills-included model, or the kids in the family who are not required to pay the bills, that kind of stakeholders either just do not care, or they don't want to sacrifice any comfort.

3. Techniques and Technologies

3.1. Introduction

Any appliance, regardless of whether it is smart or conventional, can be plugged in into smart nodes, which are compatible with any home automation protocol, such as Z-Wave, ZigBee, etc. This results in having a network called Wireless Mesh Network (WMN), which is a local network topology where controllers, nodes, switches and any other devices have a direct connection in a non-hierarchically way, where each point in the network can be either master and/or slave at the same time [59]. Appliances can be operated by functions offered by home-automation-protocols based nodes such as switching appliances ON/OFF, measuring energy consumption, recording behaviour based on the consumed energy. This chapter will briefly discuss the technologies and techniques used to implement a case study based on the proposed framework [59].

In the previous chapter, a detailed review of various smart and ordinary energy management systems was introduced. The literature review continues in this chapter in a more detailed way to review the technologies and techniques used to develop and implement these frameworks and will be also used to implement the proposed framework described in chapter 4. the chapter began by describing the microservices approach [60], then continues with describing various big data and data mining techniques including different machine learning techniques such as supervised, unsupervised and semi-supervised, tracking patterns, classification, association, anomaly detection, clustering, regression and time series algorithms. As an illustration of the usage of these technologies and techniques several energy management architectures introduced by several researchers such as Building Energy Efficiency Management Services (BEEMS) [61], Learning-based Demand Response and HVAC EMS [62], Residential Energy Management System (REMS) [63], were introduced. Cloud computing and all different types of services play an essential role in the proposed framework, so it was described in detail in this chapter. The Internet of Things concept [64], and different factors should be taken into consideration while implementing any IoT system, such as complexity, security, privacy, safety and standardisation. A glance at the efforts introduced to prevent buggy behaviour of IoT systems which were introduced by Nguyen et al. [65], is part of this section. This literature review chapter is followed by the proposed framework chapter where most of the reviewed techniques and technologies are implemented.

3.2. Microservices

The Microservices approach for the IoT means splitting up a monolithic application, in a top-down manner, into a set of distributed small size, self-contained services, to overcome the traditional monolithic software limitations and disadvantages, mainly the maintainability and scalability issues [60]. To make the best use of the IoT heterogeneity, it is required to bundle different small services, from different vendors, running on different platforms and developed using different technologies as one application to serve a specific goal, communicating together via lightweight mechanisms, Dragon et al. [66]. This is precisely the primary benefit of the microservice topology. For the planned project, several goals can be achieved when running a microservice on the cloud; this includes discarding the need to have local hardware to deliver the microservices bundle. Also, increasing the maintainability, scalability and flexibility. Moreover, cloud platforms running with green energy can be chosen, so the project's main idea of managing energy will be more assisted and highlighted. However, moving to the cloud will implicitly lead to investing more efforts in security and privacy. Besides that, the initial costs will be very high, and the ability to develop own services and customise the running ones may get reduced to keep the initial costs within the acceptable and viable level [67].

Moreover, microservices approach stands on the contrary to the monolithic architecture approach which offers a single unified unit with less scalability and changeability, where changes require rebuilding the whole application code and redeployment. It is important to mention that monolithic approach may be considered more suitable for developing and deploying lightweight applications due to many reasons related to the speed-to-market, compact and easiness of deployment via almost single package, moreover the problems of network latency and security are relatively less in comparison to microservices architecture. However, for a complex, evolving application with clear domains, the microservices architecture will be the better choice.

According to Dragoni et al. [66], although there are huge similarities between microservices and the paradigm SOA, still there are major differences in size, bounded context, independence. Following is an overview of these differences:

Size – According to the microservices approach service size must be kept small. In some organisations covering some processes with small services could be a challenging task. The main focus should be always on providing a single business value while still preserving the granularity. The benefit is seen in the increased degree of maintainability and extendibility.

Bounded context – A microservice should have a clear and single responsibility within a predefined frame. No mutual responsibilities are allowed.

Independency – Every microservice should be able to perform independently. Of course, mutual communication with other services in the environment is still needed, however, this should be performed via its interfaces which are communicating via a lightweight communication channel.

The microservice environment has some unique characteristics such as; Flexibility – being competitive and delivering responses to all emerging changes within the business environment is a key factor for any business to survive, this type of business flexibility must be accompanied and served by the microservices platform when it is needed. Modularity – The whole system is divided into small, isolated services that are concentrated in one business process. There is no single service or component which cover all different aspects inside the organization. Evolution – being maintainable while still allowing adding new features is the golden feature of a microservice that keeps it living and appropriately responding to the evolving needs of the business environment.

Microservices were covered in the literature from various aspects, one of these is using microservices for smart buildings. According to Khanda et al. [68], the buildings smart systems were designed to serve particular goals, offering a limited degree of flexibility. However, the rapidly evolving and maturation of various related technologies in almost every field has brought pervasive changes. Besides having the sensing technology, microprocessor-based appliances, and networking possibilities affordable and easy to use to be distributed everywhere, recently a microservices terminology began to be widely used. It has started with developing new programming languages and software topologies to handle the direct development of distributed applications, Dragoni et al. [66]. Recently the open-source Jolie-Programming language was built to support the microservices paradigm, jolie-lang.org [69]. The proposed architecture by Khanda et al. [68], was constructed over three steps: Step-one: Sensors are connected and configured. CC2650 SensorTags sensors were used to collect the heat, humidity and brightness. Because of its simplicity and effectiveness, data exchange was done using the Bluetooth Low Energy (BLE) network. A Z-Wave [70] protocol-based Aeon Labs device combined with the Operating System HomeOS were implemented. Figure 3.1 [68] illustrates the overall architecture with further details. Step-two: Using the programming language Jolie to build microservices that connect to sensors. A high degree of reusability and scalability among the targeted advantages, brings several implicit improvements, such as the reduction of bugs, decreasing the budget and achieving higher quality levels. Microservices farms can deal with various types of sensors because most of them share similar properties and configurations. Step three: Collect the data from different sensors and external API, then format it as CSV and pass it to MATLAB for further processing and graphics generation. Various data were collected: humidity, inside and outside temperature, light, pressure, number of opening and closing the doors, tracking persons in numbers and identity using MiBand2 fitness-trackers and device mac address. The need to configure the system to make it applicable with ZigBee protocol based devices, and distributing the system in other classrooms inside the organisation (Innopolis University), and improving the people-detection techniques, are some of the drawbacks which need to be considered in the next versions of the architecture.

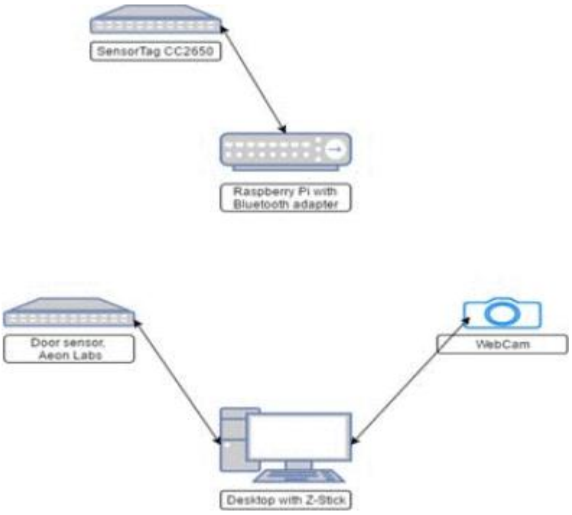


Figure 3.1: The framework of used hardware [68]

A slightly different approach has been introduced by Jarwar [61]. The focus of the research was concentrated on dealing with the collected data and enhancing its availability in real-time using different visualization’s methods. Figure 3.2 [61] illustrates the proposed architecture. It consists of several layers; Building Energy Efficiency Management Services (BEEMS), Web of Objects (WoO), Composite Virtual Objects (CVOs) and Virtual Objects (VOs). The journey of the data begins in the sensors farm located in a real-world environment. Sensors are connected with a gateway which is passing the data to the VO layer. The VO layer consists of an API, some data tagging, cleaning, caching and failure detection components. The second layer is called CVO and is responsible for aggregating and analysing the data. It has different components such as CVO life cycle monitoring functions, energy-related data processing. The third layer is dedicated to service and microservices. As part of the future work, there is a need to evaluate the results deeply, attempt to investigate the Semantic Web Agent.

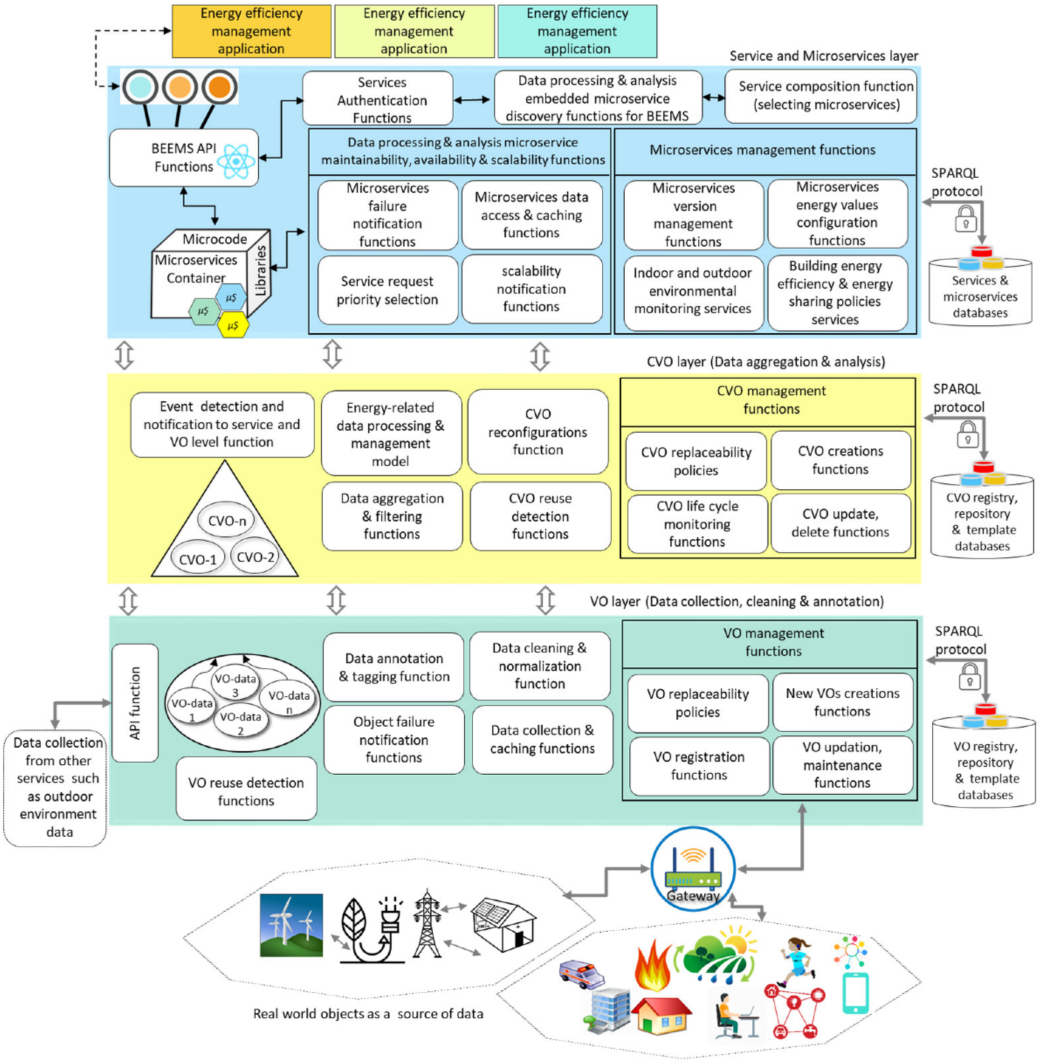

Energy efficiency management application
Energy efficiency management application
Energy efficiency management application
Service and Microservices layer
Services Authentication Functions
Data processing & analysis embedded microservice discovery functions for BEEMS
Service composition function (selecting microservices)
BEEMS API Functions
Data processing & analysis microservice maintainability, availability & scalability functions
Microservices management functions
SPARQL protocol
Microservices failure notification functions
Microservices data access & caching functions
Microservices version management functions
Microservices energy values configuration functions
Microcode Microservices Container
Libraries
µS
µS
µS
Service request priority selection
scalability notification functions
Indoor and outdoor environmental monitoring services
Building energy efficiency & energy sharing policies services
Services & microservices databases
CVO layer (Data aggregation & analysis)
Event detection and notification to service and VO level function
Energy-related data processing & management model
CVO reconfigurations function
CVO management functions
SPARQL protocol
CVO replaceability policies
CVO creations functions
CVO-n
CVO-1
CVO-2
Data aggregation & filtering functions
CVO reuse detection functions
CVO life cycle monitoring functions
CVO update, delete functions
CVO registry, repository & template databases
VO layer (Data collection, cleaning & annotation)
API function
VO-data 3
VO-data 1
VO-data n
VO-data 2
Data annotation & tagging function
Data cleaning & normalization function
VO management functions
SPARQL protocol
Object failure notification functions
Data collection & caching functions
VO replaceability policies
New VOs creations functions
Data collection from other services such as outdoor environment data
VO reuse detection functions
VO registration functions
VO updation, maintenance functions
VO registry, repository & template databases
Gateway
Real world objects as a source of data

Figure 3.2: BEEMS Architecture [61]

Dragoni et al. [66] have mentioned some issues and challenges facing the microservices paradigm. The idea of being able to develop microservices using different programming languages is attractive, however, it brings some challenges because some languages do not support any specification language, such as Node.js. Others use different specifications such as WSDL in Web Services and interfaces in JAVA. This makes the contracting approach among microservices difficult to implement and maintain. Trust and security are the other aspects. Microservices are spread over a huge surface where services are communicating via API definitions, this makes the possible attackable area huge and hard to control. Moreover, the whole idea of microservices is built upon the fact that services trust each other, so any stranger can place himself within the services farm once he manages to be part of it.

3.3. Data Analytics and Mining Techniques

Several data mining techniques extracted from the literature are reviewed in this section, by highlighting the related techniques applied for this Thesis. According to Eldén [71], during the data collection phase, it is often unknown which part of the data will be requested and used. The science to obtain beneficial information from the massive -usually- unstructured data, is called Data Mining or Knowledge Discovery. Eldén [71] also mentioned the terminology Pattern Recognition which is considered as a different technique than data mining, however, both have a related definition; "the act of taking in raw data and making an action based on the 'category' of the pattern" Duda [72]. The data mining techniques can be defined as a collection of algorithms that aim to identify, classify and group the patterns from the data being analysed, so it can be transformed into 'knowledge' and used for further decision-making and analysis.

Applying big data analytics techniques becomes a must when the volume of the generated and recorded data per second keeps overgrowing, reaching tremendous dimensions, moving from Megabytes (MB) the whole way till reaching Petabytes (1 trillion Terabyte) levels. Also, it is needed when the data's variety consists of structured and unstructured different types, such as text, photos, videos, emails, etc. It does offer the possibility to draw a full picture by gathering pieces of information of various kinds, at the same time, predicting the missing parts through data fusion. Keeping data accessible in real-time and making it useful by turning it into a value is one of the robust features of big data. Thus, velocity is measured when the output of the proceeded data is available shortly after processing to meet the growing demand. As a result of gathering data from different sources, with different formats, running on different platforms under different circumstances, it is required to apply techniques to reduce the invalid data sets (called Data Noise) this is measured and handled under the term big data veracity [73]. Figure 3.3 illustrates the 5V's of Big Data, variety, veracity, value, volume and velocity [74].

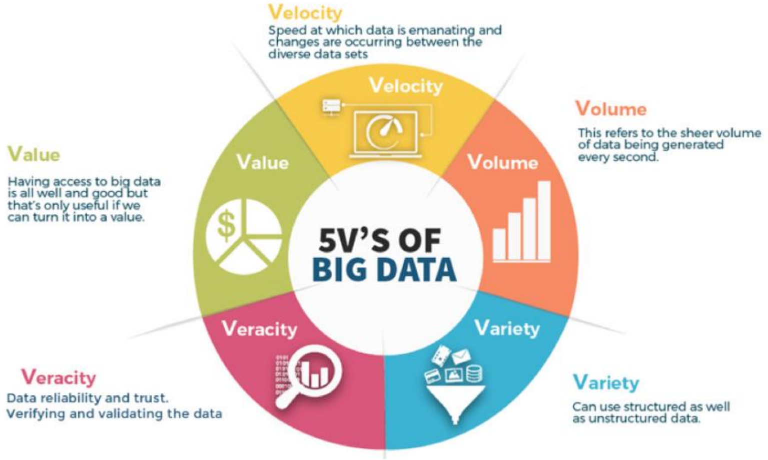

Velocity
Speed at which data is emanating and changes are occurring between the diverse data sets
Velocity
Volume
This refers to the sheer volume of data being generated every second.
Volume
Value
Having access to big data is all well and good but that's only useful if we can turn it into a value.
Value
5V'S OF BIG DATA
Veracity
Variety
Veracity
Data reliability and trust. Verifying and validating the data
Variety
Can use structured as well as unstructured data.

Figure 3.3: 5V's of Big Data [74]

3.3.1. Supervised-, Unsupervised- and Semi-Supervised ML Techniques

Data mining techniques are split up primarily into three categories: supervised, unsupervised, and semi-supervised.

Supervised Machine Learning – According to Schrider and Kern [75] the supervised ML is a machine learning task depends on previous knowledge driven from training datasets to make predictions about new datasets. It is responsible for predicting the response variable based on the input variables. Important to mention that most machine learning projects tend to utilise the supervised learning approach. Supervised learning is based on the fact that there is an input variable (x), and an output variable (Y), and an algorithm is applied to predict the output based on the input using a matching function ($Y = f(x)$). The target adjusts this matching function in the way that the output (Y) can be predicted based on the given input variable (x). This type of learning is called Supervised Learning because of the similarity between the situation where an algorithm is learning from training-data, and a session where a teacher is administering a learning process. Correct answers are known, the algorithm repeatedly delivers predictions using the trainings-dataset, till it reaches a satisfactory degree of performance and accuracy with a minimum degree of error. Linear regression, random forest and support vector machines are some examples of supervised machine learning. Supervised ML is applied in different areas, one of these is presented by Berral et al. [76], showing how this approach can be used to offer the desired Green IT in data centres.

Unsupervised Machine Learning – In contrary to the previously described supervised machine learning, neither correct answers nor a teacher exists. Its main interest is revealing the data structure without previous experience of the way how data are arranged, Schrider and Kern [75]. The algorithms are self-organised, attempt to analyse and predict the data's overall structure and the relationships among items. In this approach only input data (X) are available and known, output variables (Y) do not exist, and there is no defined matching function. The main target of this kind of machine learning is modelling the basic data structure and its relationships. K-Means clustering and the Apriori algorithm are examples of this kind.

Semi-Supervised Machine Learning – The world is not always perfect, so are the datasets. It is not always the case that data are 100% classified and labelled, or 100% unlabelled. In many cases, datasets are a combination of both. Therefore, neither supervised, nor unsupervised ML algorithms alone can deliver required responses, rather a combination of these methods. This combination is called semi-supervised, or hybrid supervised ML. An example of such a case is the photo album. In the album, the minority of photos are labelled according to location, dates, objects or persons, and the majority are not. In reality, many scenarios follow this approach, because in most cases it is an expensive, difficult and very time-consuming mission to get all data annotated as it may involve specific domain specialists. On the other side, collecting unlabelled data is considered relatively cheap, easy and painless. In the semi-supervised approach, supervised algorithms will be used to handle the annotated data, whereas unsupervised approaches will be applied to the unlabelled data to discover the structure residing in the input variables. Current improvements in machine learning have revealed that semi-supervised ML, compared to the classic supervised ML, has the possibility to resolve classification issues with fewer labelled data, Huo et al. [77].

In summary, the key difference between supervised and unsupervised machine learning approaches is data-labelling. When data are labelled, supervised ML will be applied to predict the output, however when labels are missing unsupervised algorithms will be utilised to inherit the structure from data. Having a mix of both types require applying semi-supervised or hybrid-supervised techniques.

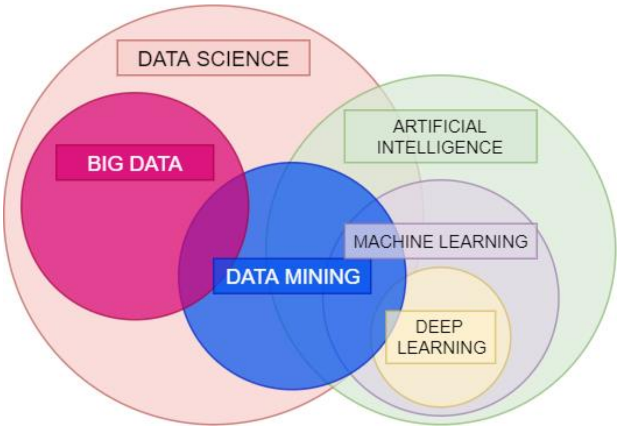


Figure 3.4: Data Science (DS), Big Data, Artificial Intelligence (AI), Deep Learning (DL), Data Mining (DM) and Machine Learning(ML) [78]

According to [79] and [78], machine learning involves studying algorithms that can retrieve information automatically, and continuously keep improving itself by gaining more and more experience when examining further data. It can predict the future by building models that represent what is currently happening, using data mining techniques. In fact, data mining is considered the technical basis for machine learning, where the whole thing begins with data mining by first collecting the data in a database, then applying data mining techniques to extract knowledge from the data, after that applying the machine learning techniques that propose algorithms based on the data and previous experience. Besides the differences of the historical roots between data mining and machine learning, they differ in their responsibility; where data mining is responsible for getting rules (knowledge or information) from the data, machine learning concentrates on teaching the system how to learn and understand these rules. Moreover, they differ in the implementation areas, data mining can be implemented on databases includes big data, however, machine learning can be implemented in artificial intelligence, decision trees, and neural networks. Inspecting the nature of each one reveals an important difference where data mining requires human interference, however, machine learning follows an automatic path, once installed it runs in stand-alone mode without further human effort. Cluster analysis is one of the major applications where data mining is used, however, machine learning is used in many applications such as spam filters, web search engines, fraud and fake detection.

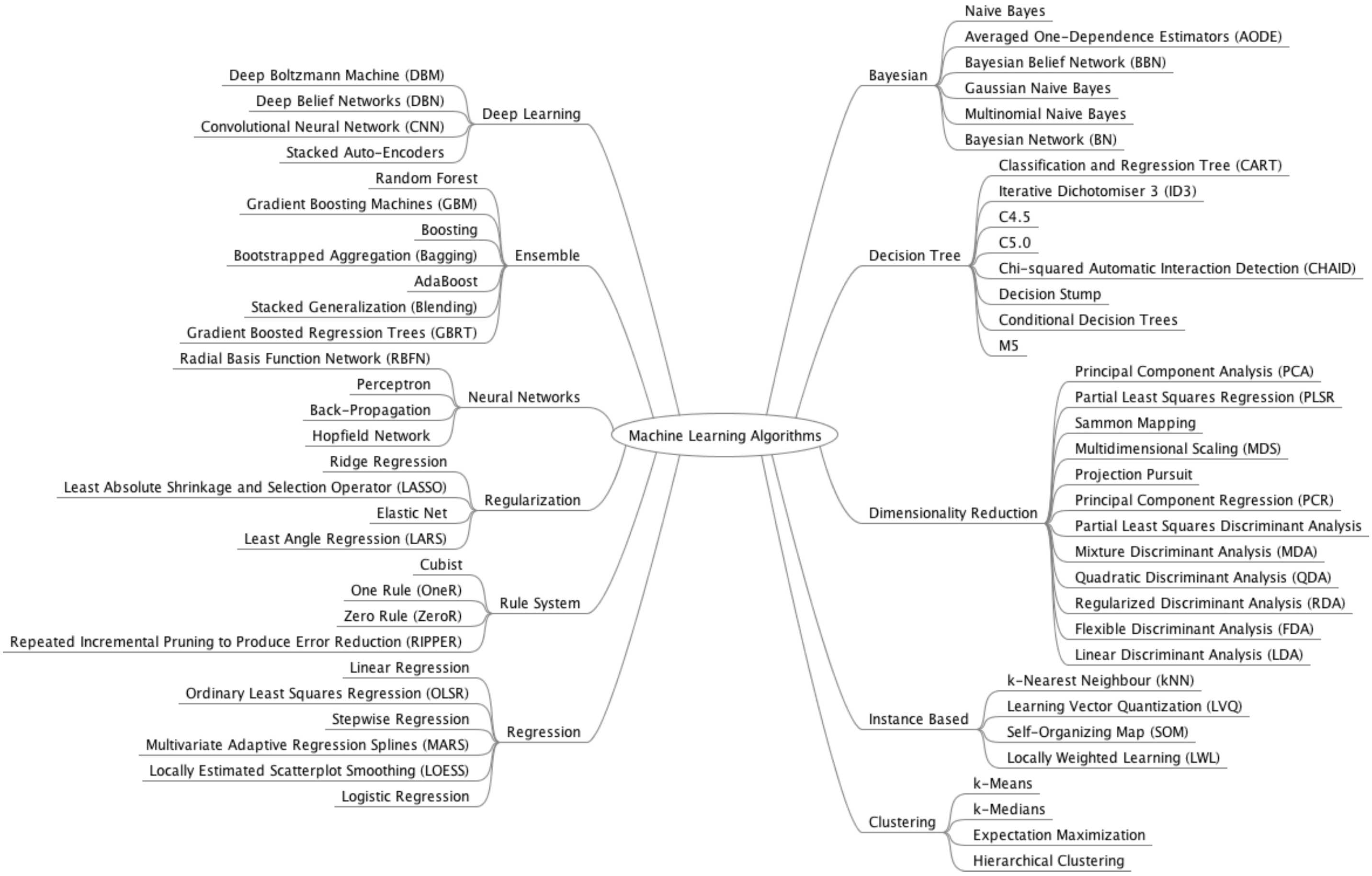

Machine Learning Algorithms
Deep Learning
Deep Boltzmann Machine (DBM)
Deep Belief Networks (DBN)
Convolutional Neural Network (CNN)
Stacked Auto-Encoders
Ensemble
Random Forest
Gradient Boosting Machines (GBM)
Boosting
Bootstrapped Aggregation (Bagging)
AdaBoost
Stacked Generalization (Blending)
Gradient Boosted Regression Trees (GBRT)
Neural Networks
Radial Basis Function Network (RBFN)
Perceptron
Back-Propagation
Hopfield Network
Regularization
Ridge Regression
Least Absolute Shrinkage and Selection Operator (LASSO)
Elastic Net
Least Angle Regression (LARS)
Rule System
Cubist
One Rule (OneR)
Zero Rule (ZeroR)
Repeated Incremental Pruning to Produce Error Reduction (RIPPER)
Regression
Linear Regression
Ordinary Least Squares Regression (OLSR)
Stepwise Regression
Multivariate Adaptive Regression Splines (MARS)
Locally Estimated Scatterplot Smoothing (LOESS)
Logistic Regression
Bayesian
Naive Bayes
Averaged One-Dependence Estimators (AODE)
Bayesian Belief Network (BBN)
Gaussian Naive Bayes
Multinomial Naive Bayes
Bayesian Network (BN)
Decision Tree
Classification and Regression Tree (CART)
Iterative Dichotomiser 3 (ID3)
C4.5
C5.0
Chi-squared Automatic Interaction Detection (CHAID)
Decision Stump
Conditional Decision Trees
M5
Dimensionality Reduction
Principal Component Analysis (PCA)
Partial Least Squares Regression (PLSR
Sammon Mapping
Multidimensional Scaling (MDS)
Projection Pursuit
Principal Component Regression (PCR)
Partial Least Squares Discriminant Analysis
Mixture Discriminant Analysis (MDA)
Quadratic Discriminant Analysis (QDA)
Regularized Discriminant Analysis (RDA)
Flexible Discriminant Analysis (FDA)
Linear Discriminant Analysis (LDA)
Instance Based
k-Nearest Neighbour (kNN)
Learning Vector Quantization (LVQ)
Self-Organizing Map (SOM)
Locally Weighted Learning (LWL)
Clustering
k-Means
k-Medians
Expectation Maximization
Hierarchical Clustering

Figure 3.5: Machine Learning Algorithms Mind Map [80]

In most cases, it is difficult to define clear borders between machine learning and deep learning. Figure 3.4 illustrates a general picture of key terminologies and techniques and their mutual relationships in this field, however, figure 3.5 shows an overview of a mind-map of all available Machine Learning Algorithms (ML) [80]. Both machine learning and deep learning are subsections of artificial intelligence, and both rely on algorithms that can modify themselves and also can feed themselves with structured data. Deep learning differs in the way that these algorithms have several layers of algorithms- everyone offering a unique analysis to the fed on data. These kinds of algorithms are called an Artificial Neural Network, this naming emphasizes the similarity with the function of the human neural networks in the humans' brain. For example, if there is a set of dogs and cats' pictures, and we need to distinguish between cats and dogs' pictures. The machine learning approach requires having these pictures labelled.

a variety of energy management systems were developed in this area. 1) An energy consumption management controller and thermal comfort management were introduced by Gao [81] and Althaher [82]. They aim to reduce the electricity bills while keeping the consumer's comfort within acceptable levels by controlling the thermal and deferrable and curtailable devices according to the retrieved dynamic price input. 2) Another EMS developed by Dittawit and Aagesen [83] aimed to reduce electricity costs in a controlled increase of offered comfort using JAVA based environment. 3) A Home Area Network (HAN) which includes a smart-meter, smart socket, a wireless communication interface [84] introduced by Saira and Ikram in 2014. It aims to reduce the overall costs and attempt to make it attractive and encourage consumers to implement it. 4) Another EMS was introduced by [85], which gives household consumers control over their devices based on their location, however, they need to update the database in case the devices' locations change. 5) Another ANN-based EMS was implemented to forecast consumption habits based on the training data sets. As a result of applying the ANN, a pattern is provided for the consumer for every device, allowing him/her to perform energy management [86].

Another research done in the household energy management area has been carried on by Zhang et al. [62]. The purpose of the research is to determine the most efficient Demand Response (DR) algorithm used to run an HVAC within the household sector. It also presents a way to design a system that can learn the energy consumption model of regulatable load appliances such as HVAC. Moreover, it suggests designs for the data structure to save and collect various appliances behaviours within the household area. The paper explains two different approaches to handle the DR within a smart grid framework: 1) Direct Load Control (DLC), and 2) Price-based Control (PbC). DLC approach enables power companies to switch ON/OFF the appliances according to available electricity levels. Its main disadvantage is the negative impact on the comfort level on the consumers' side. Meanwhile, the PBC approach offers the possibility to regulate the energy consumption based on the energy prices published every 15 or 5 minutes, or one day ahead. To achieve a better and efficient long-term energy management, and to overcome the disadvantages related to the power uncertainties which was presented during the 2015 IEEE PES General Meeting, the day-ahead approach is considered as the correct choice. Figure 3.6 [62] illustrates the proposed high-level architecture that consists of an optimization routine unit, Intelligent

Learning Routine (ILR), and a set of receivers that get some parameters such as outdoor temperature, HVAC energy, indoor temperature and the thermostat setting.

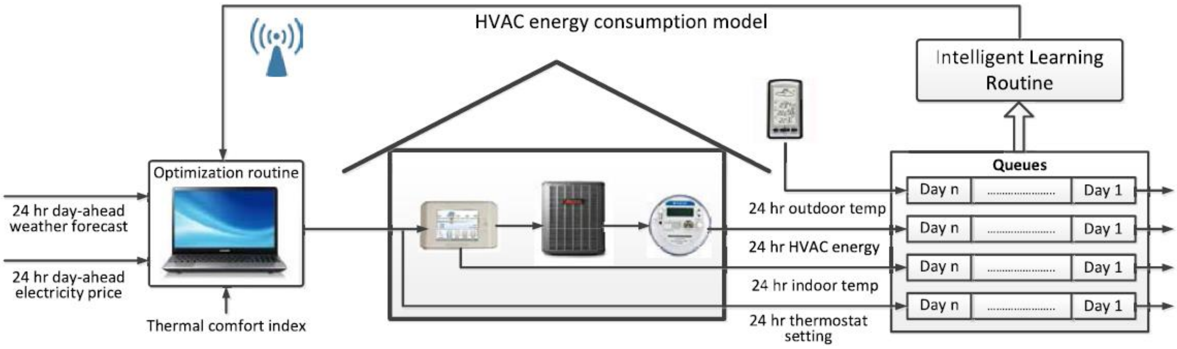


Figure 3.6: Learning-based Demand Response and HVAC EMS [62]

Basically, the system consists of two functional blocks: 1) A control block, 2) a Next-Day Demand Response (DR) block. The control block takes the DR policy generated from the previous day, then uses it to control the various wireless command receivers that are in charge of dealing with appliances distributed among the household. However, the Next-Day DR Block determines the DR policy for the HVAC for the next day based on the energy usage information obtained from the local receivers, and the predicted weather and electricity prices. The main purpose is to determine the maximum benefit for HVAC users. The evolved energy consumption is a comparison done among: (1) Results obtained from the simulator software eQuest, (2) Results from the equivalent temperature parameters ETP model, (3) Learned data from the neural network and (4) learned data obtained from regression models. The results of the comparison illustrate the fact that the learning-based demand response is the most efficient among all tested algorithms.

One of the proposed approaches is presented by Prakash and Vandana [63]. The paper illustrates a Residential Energy Management System (REMS) that switches between local storage charged by renewable energy sources, and the energy grids. The switchover automation (decisions) are based on two different machine learning approaches Artificial Neural Network (ANN) and Support Vector Machine (SVM). The paper suggests that SVM has a higher classification accuracy. The framework is illustrated in Figure 3.7 [63]

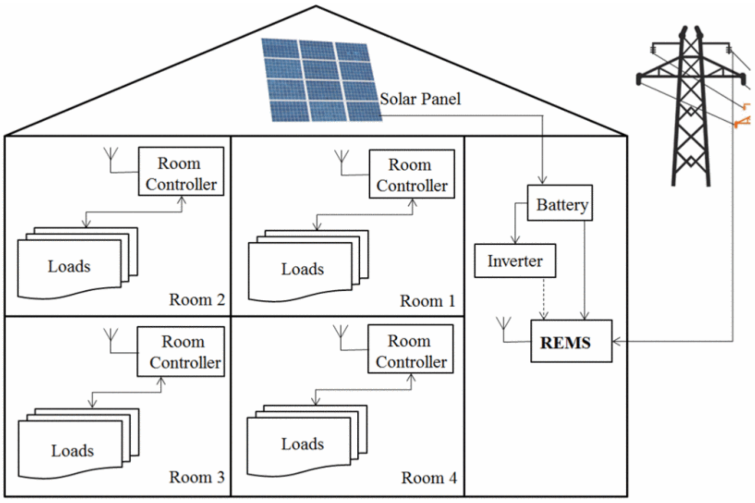


Figure 3.7: The layout of a domestic consumer [63]

[87] have introduced a system that uses the Non-Intrusive Load Monitoring (NILM) which is a method to find out the consumed energy amount for different devices centrally. Every appliance within the tested environment is equipped with a sub-metering unit to predict its upcoming consumption by applying the Bin Packing (BP) algorithm. The main goal is to provide the most suitable approach to figure the load disaggregation within a household environment. The paper uses MATLAB to predict the energy consumption of each appliance and show this information to consumers. In general, the system offers users energy usage predictions, send notifications if daily usage exceeds the set budget when maintenance or replacement is needed. Figure 3.8 [87] shows the overall block diagram.

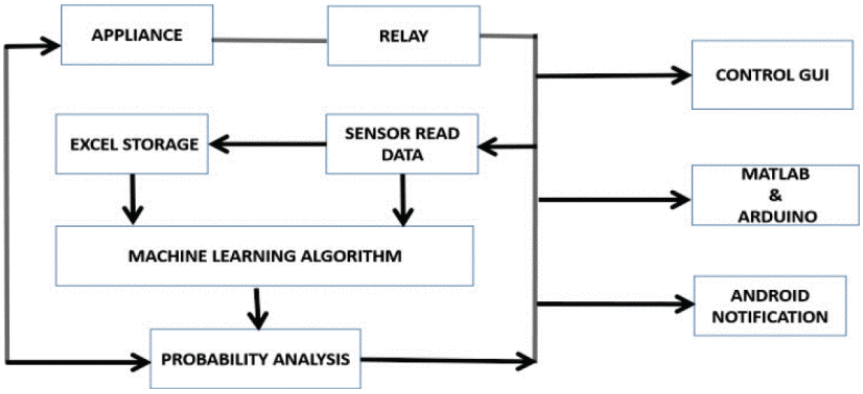


Figure 3.8: Overall Block Diagram [87]

To predict the futuristic energy consumption several machine learning algorithms, such as Logistic Regression, Decision Trees, Association Rules and Time Series, were inspected. Both the Nearest Neighbour Algorithm and Markov Chain were considered in this research because they offer the exact solution for the studied case.

According to [88], eight different regression models were inspected and tested to figure out the most suitable one for predicting short-term energy consumption in the household sector. Many other researchers have used different machine learning models such as Extreme Learning Machine Neural Network (ELMNN) [89], Generalized Regression Neural Network (GRNN) [90] and Support Vector Machine (SVM) done by [91]. Applying ANN showed that further other techniques such as weather categorization [92], parameter selection [93] [94] [95] and decorrelation [96] should be carried on. The result of the simulations done revealed that the Radial Basis Function (RBF) machine learning kernel is delivering the most reliable predictions in this field. The prediction of household energy consumption and the amount of generated energy from green energy resources is a challenging task due to the high grades of uncertainty and unpredictability. The proposed framework where the prediction takes place is illustrated in Figure 3.9.

In this approach the Artificial Neural Network (ANN) was utilised to enhance the data set and deliver behaviour predictions in many fields such as weather forecasting, energy consumption forecasting locally and on Grid level. This allows having an insight in the future to properly execute a set of pre-actions instructions to optimize the systems and enhance its ability to react properly. The same approach will be seen in the proposed framework explained in chapter 4, where it suggests utilising a number of machine learning techniques, including ANN, to predict a number of relevant data nodes related to the temperatures, operating hours, occupants behaviours, etc. The main advantage of this approach is enhancing the overall dataset and maintain a pre-reaction possibility to deal with appliances in the way to achieve the highest energy saving while keep offering similar comfort levels.

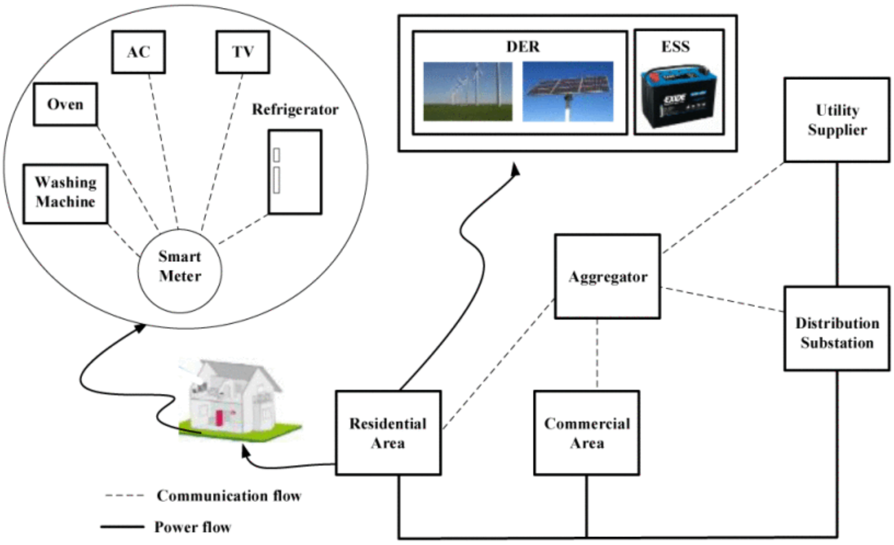


Figure 3.9: The Framework for the Smart Grid [91]

Koolen et al. [97] have introduced a different approach based on setting a natural experiment illustrated in Figure 3.10 with real-world customers and utility companies. The main aim is to ensure a high degree of demand flexibility and load reduction during peaks because it is seen that introducing a proper and right dynamic price tariff, may enhance the stabilization of the grid load [98]. A comprehensive and detailed analysis of the end-customers behaviours, preferences and household settings were carried out proving the effect of household on the energy management systems. In other words, this study proves that smart energy management systems do not perform equally on all segments and types of consumers, and therefore the extent of their success is not highly related to the consumers as well [99]. Machine learning techniques were applied to illustrate how smart management systems can recommend various tariffs schemes based on household occupants' attributes.

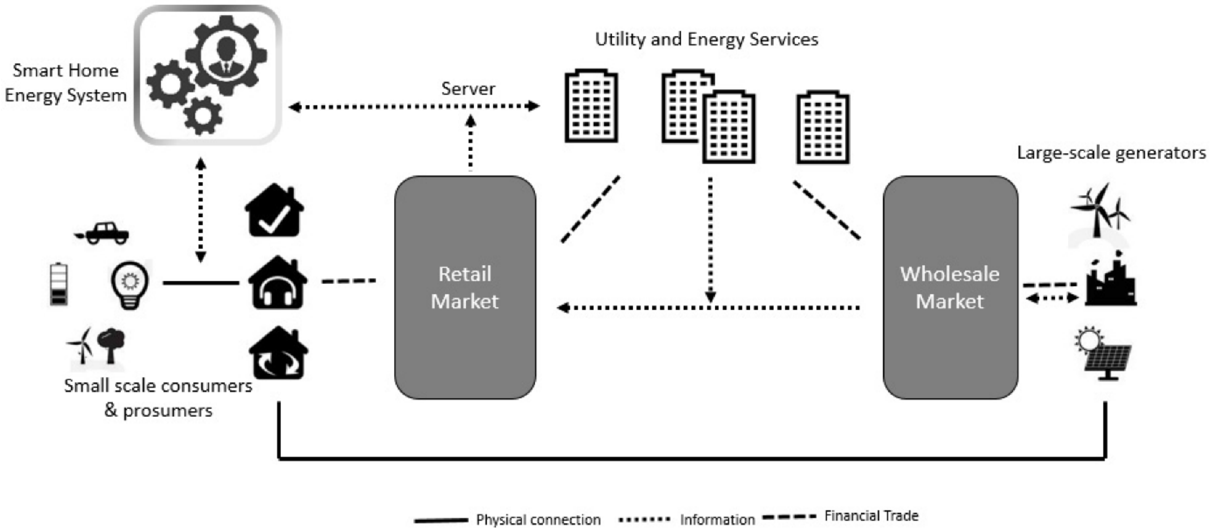


Figure 3.10: Natural Experiment Introduced by Koolen et al. with Real-World Customers and Utility Company [97]

Another approach is done using real-world data was accomplished in [100], an attempt to analyse and evaluate data-driven household energy models which are based on machine learning. Prediction models are explained and evaluated to make them easier to be understood and used by building professionals. Both advanced data analytics, which is divided into two main categories; supervised and unsupervised [101] [102], and BMS have a knowledge gap which was the focus on several efforts carried out by researchers and building professionals to overcome it [103]. Several previous studies focused on the potential of supervised machine learning to handle and analyse buildings data [104] and [105]. Mainly heating and building loads [106] and [107], and total energy consumption [108] and [109], indoor environment [110] and [111] were predicted by different machine learning techniques. On the other hand, supervised machine learning techniques are considered as high complex algorithms, such as Artificial Neural Network (ANN) [112] and [113], Support Vector Machines (SVM), Decision-Tree based methods [114] and [115]. These methods are the most used ones in the building energy management and prediction field [116] and [117] and [118]. Figure 3.11 shows the main flowchart. Most researches in this field focus on reaching high prediction accuracy levels, without paying much attention to the proper and easy interpretability, which negatively affect the proper applicability chances in practice. According to [119], there are two main ways to enhance the interpretability of predicted figures; First, using algorithms with high transparency. 2) Inject and adapt model-agnostic interpretability methods. The research work was accomplished by using the programming language R [120]

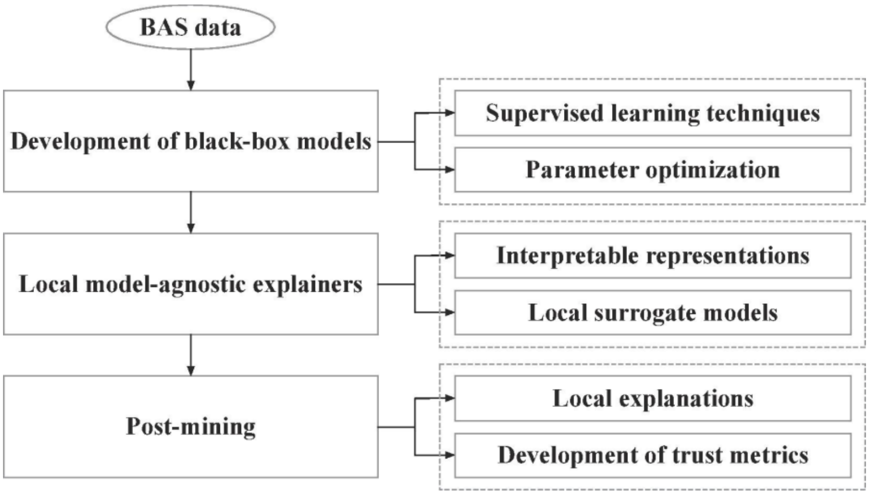

BAS data
Development of black-box models
Supervised learning techniques
Parameter optimization
Local model-agnostic explainers
Interpretable representations
Local surrogate models
Post-mining
Local explanations
Development of trust metrics

Figure 3.11: Research main Flow-Chart using Building Automation Systems [119]

Due to the enormous data volume expected in this work, and the need to have real-time results, also the need to formulate patterns to predict the future behaviour of appliances, applying data mining techniques are not an option rather a must. In this section, several data mining algorithms, including related techniques and methods, have been reviewed as part of the literature, while keeping the focuse on the particular techniques applicable to this work. Data mining or Knowledge Discovery is the terminology that describes the science used to obtain relevant and beneficial knowledge from a massive amount of data collected and stored in databases in an unstructured way [71]. Important to mention that Data mining does not imply pulling out new knowledge or information; instead, it is about estimating patterns and knowledge from collected data [121]. Data Mining techniques are a group of algorithms that offer the needed assistance to group and classify the collected and analysed data into information patterns; this ultimately allows a better understanding and treatment. The separation of data instances has the advantage of deriving pieces of unknown information from the collected patterns, which offers excellent help to take more accurate decisions. Data mining techniques are the only avenue towards getting the benefit of the data mining concept; these are:

3.3.2. Tracking patterns

This technique is considered among the very fundamental methods in the data-mining field. It is natural and intuitive for many users. It begins with recognising irregularity in the collected data sets over a period. In our context, for instance, energy consumption increases during low-temperature days. Identifying anomalies occur when data sets contain data elements that do not fit anywhere. For instance, when energy consumption usually increases during weekends, then suddenly it drops dramatically during the weekend within holiday seasons, this is considered a new situation that may need to be re-evaluated.

Aggarwal and Bhatia [122] have introduced research with a detailed analysis on different techniques to discover patterns in online data mining. Four different algorithms were analysed; Apriori, FP-Tree, Category based, and fuzzy logic-based. The comparison revealed that the Apriori algorithm can be used for searching large item sets, however it suffers from drawbacks because it requires a full scan even for a single itemset. Moreover, the FP-Tree technique is complex, however, it supports the frequent scan. The category-based approach can benefit from having similarities in users' interests. Finally, the fuzzy-based algorithm can better deal with uncertain and ambiguous situations.

3.3.3. Classification

Compared to the tracking patterns, classification is considered more complex. Classifying data requires defining distinct characteristics and attributes of the collected data putting them in categories in order to draw conclusions or futuristic predictions. An example of such technique is describing individuals' financial backgrounds according to their daily, weekly and monthly purchases by classifying them into three main credit risks: low-risk, intermediate-risk or elevated-risk. Having such a classification opens avenues towards learning further knowledge about these individuals in other fields, such as their ability to buy certain brands. Several classification algorithms are reviewed in the next sub-sections.

3.3.3.1. K Nearest Neighbour (KNN)

According to Fan et al. [123], K-NN is considered as a stable theoretical and easily implemented algorithm [124] used to provide solutions for nonlinear issues, for instance; credit-related and customer-related rankings. This algorithm is offering much assistance in cases when the gathered data do not adhere to the linear theory. In other words when there is little previous knowledge about the data. Moreover, while dealing with experimental processes it reduces the variables impact on the data [125]. K-NN has proven its ability to achieve high forecasting accuracy. In the real world, this algorithm is widely applied; for instance in the analysing process of the stock market

[126], in photovoltaic systems to detect and diagnose faults [127], and in social networks for facial recognition [128]. Figure 3.12 shows an example of a K-NN proximity map, where the new element (X) belongs to four neighbour elements in the group (Circles) and two neighbour elements in the group (Squares).

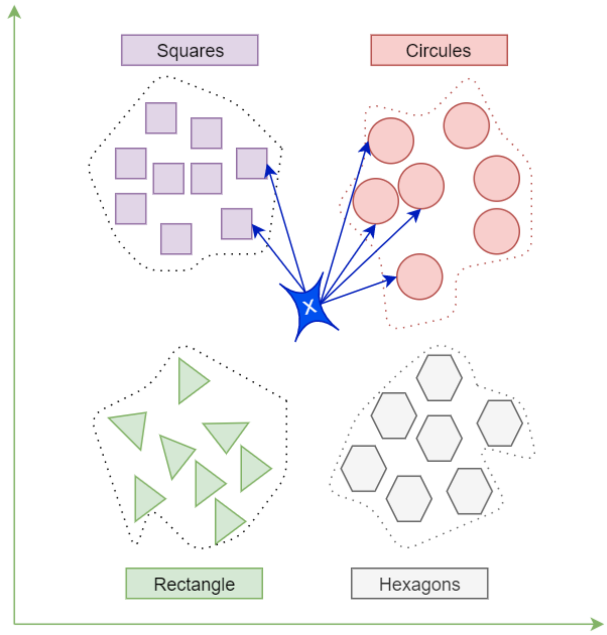


Figure 3.12: K-NN proximity algorithm map

Additionally, many improvements were made on the K-NN algorithms, for instance, the one introduced by Zhang et al. [129], which is applied to classification, regression. Moreover, the Weighted K-Nearest Neighbour (W-K-NN) has been introduced by Troncoso et al. [130], which has been followed by many scientists and researchers who have added the weight for nearest neighbour. For instance, the research carried on by Chen and Hao [131] illustrated a weighted K-NN which is established on the SVM. The purpose is to predict the stock market.

3.3.3.2. Neural Networks (NN) and Deep Learning (DL)

NN was designed to create artificial intelligence by simulating the human being nervous system, where artificial units are acting similar to human neurons, Aggarwal [132]. Rosenblatt's perceptron algorithm is considered to be the cornerstone of all neural networks that followed. According to Aggarwal [132] NN -theoretically- has the capability to learn any mathematical function when enough training data is available. Even more, specific variants of NN such as recurrent NN are considered Turing complete, this means that NN can simulate any algorithm if it is provided with enough and sufficient training data. Important to mention that a tremendous amount of training data, and long processing time, are needed in order to learn a simple task such as image recognition, even with using high-performance computers. As illustrated in Figure 3.13, the simplest form of NN consists of three tiers; the input tier, an output tier and in between a hidden tier.

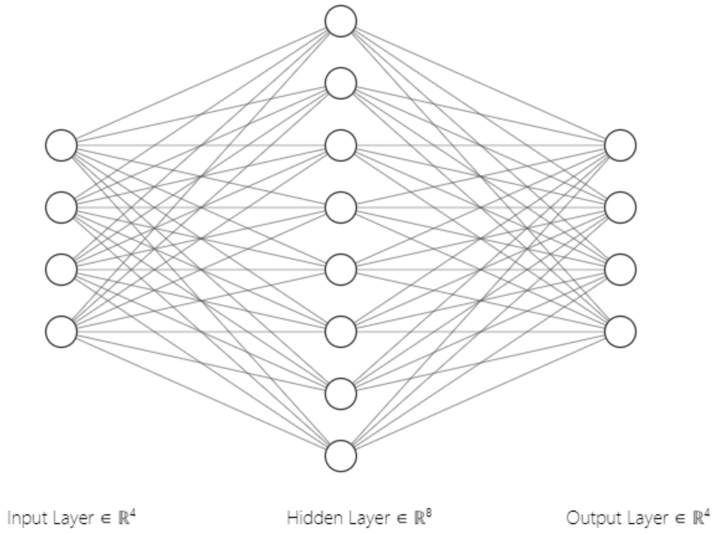


Figure 3.13: Neural Network Example

Having huge data and the expansion of the computational capacity at the beginning of the century has given a suitable environment to reborn a new approach called Deep Learning (DL) [132]. Deep Learning is an NN made up of more than three tiers: input-tier, an output-tier and many hidden tiers. It is a self-teaching and learning system that filters sufficient test data in the same way humans do. Comparing typical machine learning algorithms with deep learning technology reveals that deep learning can reach higher accuracy levels when enough sufficient data and computational power are available. In fact, in some fields such as; playing computer games, recognizing image or self-driven cars, deep learning has reached levels close to human performance or even exceeded it in some cases, and with a positive future prognosis related to the rapidly evolving computational power, sufficient data and intensive experimentation, the deep learning will even reach much better levels in new fields, by the end of this century it is expected that deep neural networks will be able to train neural networks with the same amount of neurons reaching similar levels to the human's brain. Figure 3.14 illustrates an example of 4-hidden layers deep learning neural network.

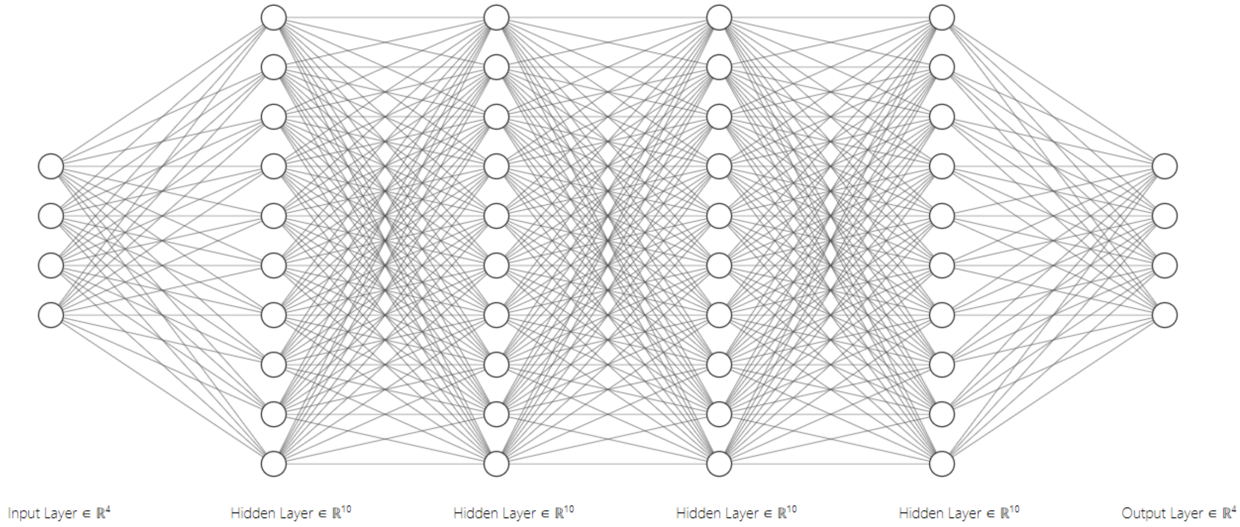


Figure 3.14: Example of Deep Learning Neural Network (DLNN)

Regardless of the number of hidden-tiers, both Deep Learning (with several hidden-tiers) and NN (with one hidden-tier) follow the same approach, the difference lays in the granularity level. Both approaches begin with a set of configuration options, such as (1) the quantity of hidden-tiers, (2) the nodes' quantity in every hidden-tier, (3) the used activation function, these can be:

$$\text{Sigmoid: } f(x) = \frac{1}{1+e^{-x}},$$

$$\text{Tanh: } f(x) = tanh(x) = \frac{2}{1+e^{-2x}} - 1, \text{ or}$$

$$\text{Rectified Linear units (ReLu): } f(x) = Max(0, x).$$

According to Hayou [133], the chosen function affects the performance dramatically during the training phase. An improper choice of the function may lead to loss of information during the forward propagation and the rampant disappearing/exploding of gradients within the back-propagation phase. (4) the learning rate, which means how much should this step outcome affect the weight and biases, (5) the momentum, which defines how much should past outcomes affect the weights and biases, (6) the number of iterations and (7) the desired error level. When training data are fed to the NN, both 'weights' and 'biases' get changed/adjusted till reaching the previously defined iterations level, or the allowed error rate. The more test data are used, the higher the prediction quality is gained.

Training a NN can be reached by utilising several techniques, such as Multi-Layer Perceptron (MLPs) (also called Back-Propagated Delta Rule Network), Newton's method, Quasi-Newton, Gradient Descent, Levenberg Marquardt and Conjugate Gradient. These techniques are explained in the following section.

Multi-Layer Perceptron (MLPs) – According to Heidari [134], the MLPs is a supervised learning technique developed to mitigate the drawback of the Single Layer Perceptron (SLP) related to not being able to efficiently detect and tackle the nonlinearly distinguishable patterns [135], by utilising several hidden layers. It consists of more than one perceptron with one input layer that gets the signal, and one output layer which is responsible for making decisions and an arbitrary amount of hidden tiers which are considered the brain of the whole technique where the whole magic occurs. The idea is to train the algorithm on a collection of input-output duos and extract the relationship from them. This process includes continuous adjustments of the used parameters (weights and biases) till reaching the minimum error rate or reaching the maximum iteration loops. The whole technique resembles the ping-pong game where the ball goes constantly forth and drawbacks, guessing what we think we know and as a response, we get the feedback on how wrong we are. MLPs advantages can be summarized as having a high potential to learn fast and effective, being robust to noise, the nonlinearity and parallelism approach representing a good tolerance towards faults, and finally introducing high-level competencies in generalizing assignments [136].

Gradient Descent (GD) – Also called Steepest Descent, considered as one of the most straightforward algorithms within the area of the neural networks. Information from the gradient vector is used. With every successive iteration, the error function is optimized by setting the training rate to one of these modes: fixed value, or one-dimensional optimization. In every iteration the training direction of gradient descent is computed, also a proper training rate is observed. The main goal of this approach is defining the local minimum point by applying the Hessian Matrix (HM) which defines whether this surface point is a stationary minimum or stationary maximum point [137]. Being slow is one of the major disadvantages of this approach, however, when it comes to dealing with big neural networks that include thousands of variables, this algorithm is recommended because the Hessian matrix does not get saved, rather only the gradient vector.

Newton's Method (NM) – This method optimizes the training direction by using the Hessian matrix to assess the 2. derivatives of the loss function. Similar to the previously described gradient descent, through line optimisation the training rate either can be calculated or fixed, however, the measurement -against expectation- may reveal maximum values, not minimum ones. This occurs because the Hessian matrix is not being positive-definite. Compared to gradient descent, the main advantage of such a technique is the fact that it takes fewer steps to locate the lowest possible value of the loss function, however, it consumes a high computational power to evaluate the Hessian matrix and the related inverse.

Conjugate Gradient (CG) – Proposed by M. R. Hestenes and E. Stiefel in 1952 [138]. Mitigating the slowness backward of previously described Gradient descent, and the high computational consumption of NM's algorithm is done by applying the conjugate gradient algorithm. In other words, this algorithm is designed to bypass the extreme information required by NM's method to store, evaluate and inverse the HM matrix, and also to speed up the GD's slow convergence. With respecting the HM matrix to reduce the required information, a search is done along the conjugate directions to speed up the convergence and to conjugate the teaching routes. This approach does not need the Hessian matrix inversions, so it is considered a better choice to deal with bigger neural networks. Moreover, it is also considered better than Gradient descent because of previously mentioned optimisations.

Quasi-Newton – Also known as the Variable Matrix Method, it is considered as an improvement on Newton's approach because it illustrates better computational advantages. During iterations, in every step, instead of performing direct Hessian calculations which are followed by inverse measurements, an estimation of the HM matrix inverse is calculated. The exact calculation of the Hessian matrix and its inverse are not necessary, because using the first partial derivative of the loss function allow the direct building of the approximation of the HM matrix inverse, and ultimately an improved performance than GD and CG is provided.

Levenberg-Marquardt Algorithm (LMA) – Also known as Damped Least-Squares (DLS). A numerical optimization algorithm named after Kenneth Levenberg and Donald Marquardt offers a solution for nonlinear curve fitting. It is a NN training algorithm designed to optimize the loss functions which are calculated as a collective of squared errors by using gradient vector and Jacobian matrix, which contains the derivatives of errors. Compared to both the Gradient descent algorithm and Conjugate gradient, LMA is considered a fast alternative to train a NN, however, it is only applicable on only a specific type of loss function. Moreover, it is not considered suitable for huge NN, because its memory requirements proportionally grow with the NN size.

The available resources and the range of the NN are the deciding factors that define the advantages and disadvantages of each algorithm. Compared to the Levenberg-Marquardt algorithm, Gradient and Conjugate gradients are considered more suitable in case NN consists of thousands of parameters. On contrary, Levenberg-Marquardt functions best with NN that have a few hundred parameters and a few thousand instances.

Evaluating the constructed model is as important as creating the model itself, therefore for the classification problems, there are some evaluation metrics taken into consideration in this field:

Area Under ROC Curve (AUC) – This metric simply represents the probability to rank a randomly chosen positive example higher than a randomly chosen negative example. This metric depends on calculating two different values: True Positive Rate (TPR), and False Positive Rate (FPR), then plotting these results on an XY dimension. AUC is the resulting area beneath the line.

$$TPR = \frac{T_p}{\sum_{i=1}^{n}(T_p + T_n)}$$

$TPR = True\ Positive\ Rate$

$i = variable$

$n = number\ of\ observed\ cases$

$T_p = predicted\ "True\ Positive"\ values$

$F_n = predicted\ "False\ Negative"\ values$

And,

$$FPR = \frac{F_p}{\sum_{i=1}^{n}(F_p + T_n)}$$

$FPR = False\ Positive\ Rate$

$i = variable$

$n = number\ of\ observed\ cases$

$F_p = predicted\ "False\ Positive"\ values$

$T_n = predicted\ "True\ Negative"\ values$

Classification Accuracy (CA) – This is considered the typical metric which is calculated by taking the proportion of true outcomes divided by the entire quantity of all observed cases. The equation looks like following:

$$CA = \frac{1}{n}\sum_{i=1}^{n} T_p + T_n$$

$CA = Classification\ Accuracy$

$i = variable$

$n = number\ of\ observed\ cases$

$T_p = predicted\ "True\ Positive"\ values$

$T_n = predicted\ "True\ Negative"\ values$

3.3.4. Association

It is much like the tracking-patterns approach mentioned in a previous section; however, it is specialized in linked variables that share high dependency rates. In other words, it attempts to spot the likelihood of the coincidence of elements within a group. This approach is specialized in looking at certain events or attributes that have something in common with other events or attributes. For example, in some online stores, usually, customers get recommendations to buy different items when they decide to buy a certain item. Recommendations are made based on the association done based on previous purchasing experiences from other customers in the same category.

Rules driven from the interconnections among items in the collection are called association rules. These rules are extremely important to define the relationships among items and therefore to find the hidden information inside the big data collections [139]. One of the most famous stories about how strange the relationships between items within one collection can be is the Beer & Diapers story. A supermarket survey has revealed that young men who purchase diapers tend to purchase an alcoholic drink (Beer) as well, this has been represented as the association rule mining “Beer and diapers are sold together” [140]. Several algorithms are used to create those rules; for instance: Eclat algorithm, Apriori algorithm, FP-growth algorithm, OPUS search and ASSOC.

Apriori Algorithm – The first and the most popular association rule mining technique [141], founded by Agrawal and Srikant in 1994 [142]. It consists of two main processing layers: connection and pruning [140]. It uses the Bottom-Up approach, the first step is called Candidate Generation where the frequently repeated sub-sets are extended step by step using either Hash-Tree or Breadth-First-Search (BFS) approach. The second step is examining a group of possible candidates against the database. The algorithm keeps running until no further extensions are found. This algorithm suffers from two major issues; the frequent scanning of the database, and the generation of large results of candidate data sets [141]. The following example illustrates this algorithm: Considering the data in table 3.1,

β	α	γ
β	α	δ
β	α	γ
β	α	δ

Table 3.1: Example Data used by Apriori Algorithm

following association rules can be extracted:

- 100% of datasets contain β, also contain α
- 50% of datasets contain β, also contain γ
- 50% of datasets contain β, also contain δ

Eclat Algorithm – One of the famous association rule mining methods, stands for Equivalence Class Clustering and bottom-up Lattice Traversal. Its basic idea is iterating through the data and using the transaction-id sets (called tidsets) to spot the candidate within the dataset. In the first iteration, all items will be assigned to their tidset. A recursive call of the function leads to identifying and joining the previously identified tidset-pairs. The whole process stops when no further candidates can be combined. Compared to the previous Apriori Algorithm, it has better efficiency and better scalability. Since it is based on the Depth-First-Search, it is considered faster than the Apriori technique and requires less memory. Moreover, it performs a smaller number of iterations because it is scanning the whole database repeatedly. Recently, in 2018 an extended version of the Eclat-Algorithm was introduced by Szathmary [143], it focuses on offering a way to filter the Frequent Closed Item-sets (FCIs) from Frequent Item-sets (FIs).

Frequent Pattern-Growth Algorithm – Introduced by Han et al. in 2000 [144], aims to obtain frequent entries which will be later on used to extract association rules. The strategy Divide and conquer is used, this implies compressing the database of the provided frequent sets using the special data structure FP-Tree, then splitting it into a series of conditional databases. In the first iteration, the incidences of items are searched and stored in a separate table. In the second stage, attribute-value pairs which were saved in the table gets converted into a trie (digital tree). In the tree, each transaction is put in descending order. Compared to the Apriori algorithm, the FP-Growth method requires only 2 scans to go through the whole database, whereas Apriori requires *n+1* scans. Also, it does not require as expensive computational resources as Apriori does.

All previously mentioned algorithms are doing half of the job, they are specialized in mining the frequently repeated item-sets. This requires applying further steps after defining the mining association rules in the relational databases. As illustrated in Table 3.1, the extracted rules were made based on the output of the Apriori algorithm.

3.3.5. Anomaly Detection (Outlier detection)

Anomaly detection is finding patterns that do not correspond to anticipated behaviour. In the literature, this approach has been defined using different expressions "Regions of the network whose structure differs from that expected under the normal model" Savage et al. [145], "Patterns in data that do not conform to a well-defined notion of normal behaviour" Chandola et al. [146], "Outliers may be considered as noise points lying outside a set of defined clusters or alternatively outliers may be defined as the points that lie outside of the set of clusters but are also separated from the noise" [147]. Simply identifying the frequently repeated patterns among the dataset cannot always provide a straightforward interpretation of the nature of the data. Detecting anomalies or outliers might play an essential role to understand and properly interpreting the data. For instance, in a shop, when statistics show that the majority of customers are male, then suddenly numbers show that in particular time-span customers turned to be females, an investigation must be carried out to derive some useful outcomes from this sudden change, such as how this can be replicated? Or which products should be offered to serve this new transient group and convert them to permanent customers.

This kind of detection is usually applied to unsupervised data and divided into two different assumptions: 1) It occurs very rarely, 2) and differs significantly from the usual instances. It is not unusual that researchers are getting confused sometimes when it comes to the exact definition of anomalies and the differences between it and other terms such as Noisy Data. As proposed by the definition of anomalies which is provided by Aggarwal and Yu [147], anomalies are seen differently than noise, because noise is usually considered as a random error or a divergence described in a variable. As an example, consider the credit-cards holders' behaviours during their purchasing activities. In one day when the somebody buys a larger dinner than he usually does or buys a smaller cup of coffee than usually does. This may be seen as a deviation or variance, however, it is in fact noisy data. Therefore it should not be considered anomalous. To avoid unnecessary and expensive dealing with anomalous, it is recommended to remove the noisy data, then apply the anomalies detection techniques. Figure 3.15 [148] illustrates an example of an anomaly situation.

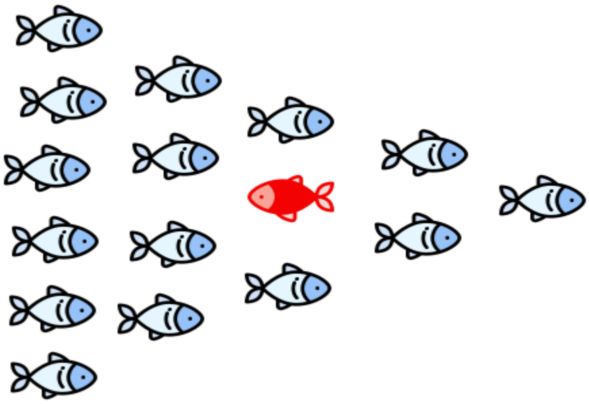


Figure 3.15: Swimming Fish - An example of anomaly situation

The detection of anomalies can be applied in almost every field, Kaur and Singh [149] have introduced a survey to apply it in the Online Social Networks (OSNs) to observe and fetch the anomalous activities represented by unusual and illegal activities. According to Kaur and Singh [149] anomalies can be divided into different types based on their nature, based on the availability of information in the graph/network, based on the behaviour, and finally based on the interaction patterns, such as; near stars, heavy locality and particular dominant links. Kaur and Singh [149] also introduced many data-mining methods that aim to identify anomalies which are divided into 3 major groups: supervised-, semi-supervised- and unsupervised-methods.

Supervised methods – It considers studying the case as a classification issue, where experts label only the normal data, so all other data are considered anomalous. Or the way around, which means labelling the abnormal data, and considering any other item not corresponding as a normal item. The classifier responsible for implementing the classification approach can be established on Neural Network, Support Vector

Machine or Bayesian Network (BN). The major challenge is teaching the classifier properly.

Unsupervised methods – This is applied when labelled data are not available, thus the clustering approach is applied. Normal items are expected to form a kind of clustered forming groups by following a certain pattern, where anomalies seem not to follow these patterns and form their own ones. However, sometimes, anomalies are the ones that form patterns, these are called collective anomalies as shown in Figure 3.16. In this case, clustering is not efficient, since it generates a huge false alarm overhead, where every normal object is considered anomalous.

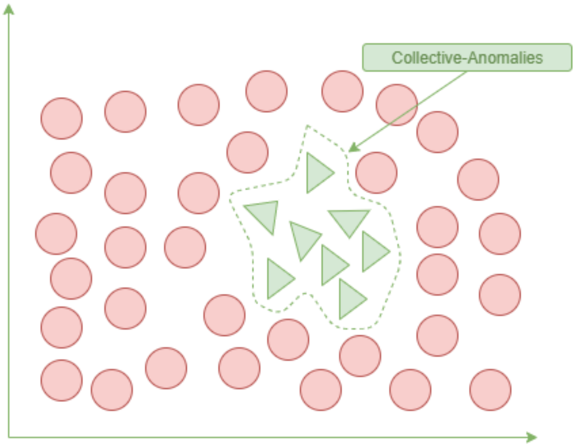


Figure 3.16: Collective Anomalies Example

Semi-Supervised methods – It works with 2 different types of data: labelled and un-labelled. In this approach, the classifier is trained using the available labelled data, so it can discover and classify the unlabelled data. Consequently, a significant model of normal data is constructed, which is used to discover the anomalies in the data based on the fact that these anomalies do not fit the model. This approach is called the self-training approach. Another approach is called co-training where more than one classifier performs mutual training against each other.

This approach is useful and will be applied to the data collected from the proposed I3SEM. Because it is expected to have some energy consumption peaks, due to unexpected extreme external effects, such as weather, accidents, catastrophic incidents.

3.3.6. Clustering

Clustering and classification are very similar. Clustering differs because it involves grouping items or objects into chunks based on common features, attributes and similarities. For instance, an audience classified based on its demographics can be also clustered into different chunks based on its income or shopping habits. According to Aggarwal and Reddy [150], clustering approaches are investigated in many fields begins with pattern recognition and data mining until reaching databases and machine-learning algorithms. To have a comprehensive understanding of clustering aspects three main areas must be covered: methods, domain and variations and insights [150].

Methods – the group of methods describe the key techniques used for clustering approaches, such as nonnegative matrix factorization, partitional clustering, probabilistic clustering, feature selection, grid-based clustering, density-based clustering, agglomerative clustering, and spectral clustering.

Domains – As mentioned before, data clustering almost exists in every data domain, such as multimedia, text, stream, biological, big-data, time series, graphs, and categorical data.

Variations and Insights – Clustering have different variations and types, among them: cluster ensembles, interactive clustering, cluster validation, Multiview clustering, and semi-supervised clustering.

As mentioned before, clustering and classification are very similar, both are considered machine learning algorithms that split a collection of items and objects into groups based on one or more common features and characteristics, however, there are some differences in the data mining's context. The key difference is the usage domain. Classification is used in supervised learning approaches where data has predefined labels designated by properties, however, clustering is used with unsupervised learning where objects and items are grouped according to their feature and properties. Moreover, in classification, training data that are used for teaching purposes must be provided, where, in clustering, there is no need for training data. Figure 3.17 illustrates the difference between classification (the left-side graph) and clustering (the right-side graph). The following sections describe a bunch of clustering types, such as K-Means and hierarchical clustering.

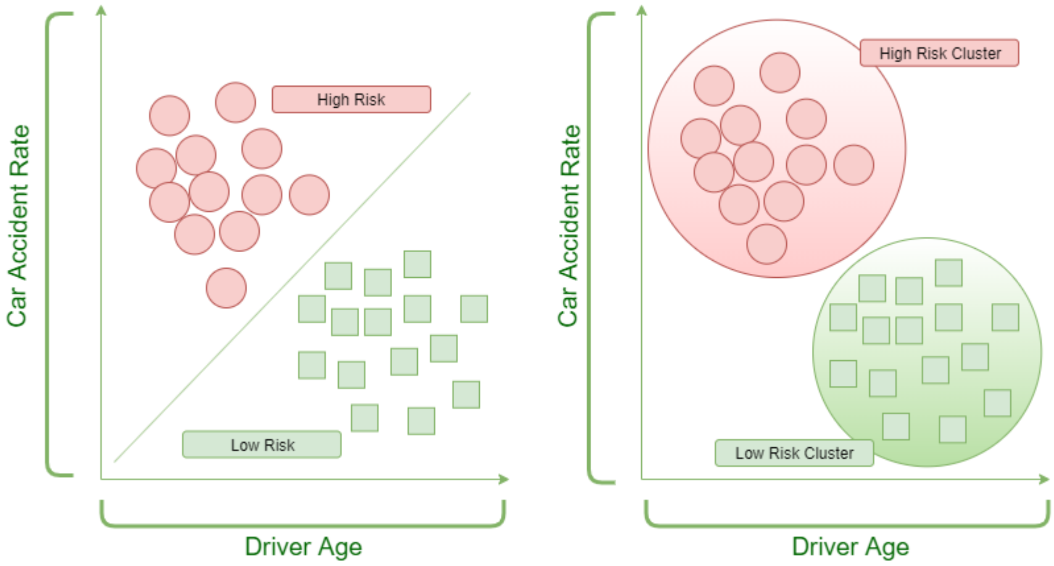


Figure 3.17: Differences between Classification (Left) and Clustering (Right)

3.3.6.1. K-Means Clustering

A straightforward, unsupervised machine learning technique aims to group data into groups based on similarities, however, it is known for its high sensitivity to the initially chosen cluster centres [151]. The determination of the similarity criteria among objects within one cluster/group is the key factor in the whole design, where each criterion is there to address a specific problem. In our context, the I3SEM, there are many similarities; for instance, the fact that appliances consume a certain amount of energy that can be divided into groups as *low energy consumption*, *medium* and *high*. Also, a similarity can be observed in the appliances' running periods, some appliances may need to run occasionally; a couple of times a week, such as washing machines, some of them run seasonally such as AC/Heating appliances, or on daily basis such as bulbs, dishwasher machines, or continuously such as sensors and cameras. *K* is the number of groups that will be created by the algorithm based on the provided configuration. Generally, it is not possible to determine the exact value of *K*, however, according to Oracle.com [152] an accurate estimation can be gained by a commonly used metric which is called *the mean distance between data points and their cluster's centroid*. This method is based on comparing the results acquired by applying different *K* values. There are several methods to verify the *K*'s value, for instance: the silhouette method, cross-validation, theoretic jump, and the G-means algorithm. Figure 3.18 illustrates an example K-Means graph.

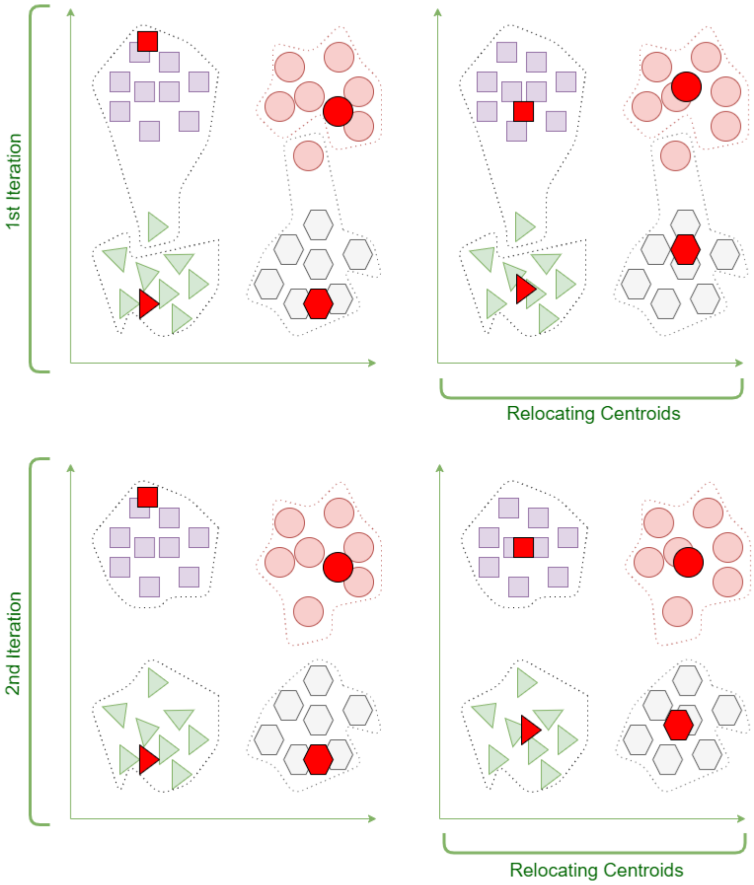


Figure 3.18: A K-Means Example

According to [153], K-Means has some advantages such as being easy to implement, suitable for huge data sets, offering a guarantee for approximation, can be easily getting adapted for new data sets. However, it suffers from some disadvantages, such as the manual choosing of the K value, the dependency on the initial values.

3.3.6.2. Hierarchical Clustering

Hierarchical cluster analysis is another name of it which is based on grouping similar objects in clusters, in the way that each group (cluster) is unique and objects within each cluster are highly similar to its neighbour objects. In other words, according to Cohen-Addad et al. [154], it is defined as a repeated separation of the dataset's chunks into clusters at a continuously finer granularity. This kind of clustering is split into two kinds: agglomerative and divisive. In the first type *Agglomerative* (may also be called *Bottom-Up* approach) every object begins as a cluster, then clusters are combined with climbing the hierarchy [155]. The second type; divisive, or *Top-Down* approach where objects belong to one cluster that split into further clusters descending within the hierarchy. A comparison mechanism is applied to find out the similarity or distance between two clusters, this occurred by weighing the similarity between items from one cluster and an item from another cluster, these get clustered when the degree of similarity among them is greater than other clusters. However, the *top-down* approach starts by dividing objects in clusters into further levels based on similarity by keeping similar data points (objects) in the same cluster. Figure 3.19 illustrates both approaches.

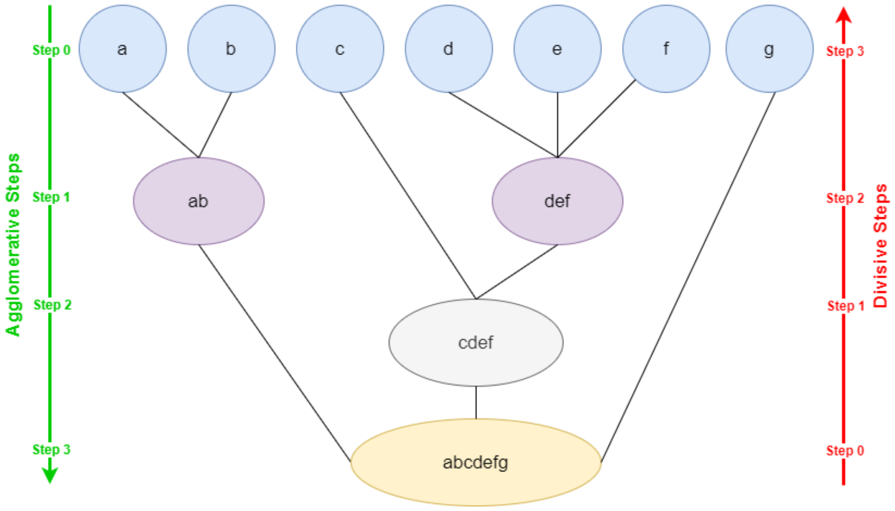


Figure 3.19: Example of Hierarchical-Clustering: Agglomerative and Divisive

Because of its easy-to-use attribute, hierarchical clustering can be useful for the I3SEM framework, if we can pre-process the data obtained from various sensors and appliances in a way the data dimensions are not huge. However, it has some weaknesses such as (1) not being able to always provide the best solution, (2) a lot of arbitrary decisions are involved. In other words, decisions, such as defining both *distance metric* and *linkage criteria*, are rarely taken based on a theoretical basis. (3) Poorly works with datasets containing mixed data types.

Hierarchical clustering is considered very popular, however, the applied types, such as agglomerative, are restricted to the offline setting and therefore needs the whole dataset to be available, this brings some restrictions when used on large datasets [156].

3.3.7. Regression

One of the supervised machine learning algorithms is used principally as a method of modelling and planning. It assists in identifying the likelihood of a variable (target or dependent) based on the relation to other variables (predictor or independent) in the same field. The main focus of regression is discovering the precise correlation among two or more variables within a collection of variables. For example, a regression could be used for projecting the price of a product based on other aspects such as competitors, demand or weather forecast. According to Oracle [157] "Regression is a data mining function that predicts numeric values along a continuum". It all begins from a data set that contains similar target values, for example, if the regression is going to be used to predict house values in a region, a data set with observed houses prices over a period of time, together with other attributes (called predictors) such as; the number of rooms, proximity to shopping centres and schools, house age, must be provided. The regression model is built during a training process, where the relationship between predictors and the target value is predicted. The regression can be tested by applying known historical statistics and then assessing the difference between the forecasted and exact values. Typically, historical statistics are split into two groups: one to build the model and one to test it.

Regression functions have different families, and errors are measured in different ways such as Linear Regression, Logistic Regression, Polynomial Regression, Stepwise Regression, Ridge Regression, Lasso Regression and ElasticNet Regression. Following is a brief description of each technique.

Linear Regression – One of the most well-known and most chosen predictive modelling techniques. According to this technique, the target (or dependent) is a continuous variable, the predictor (or independent) is either continues or discrete variable, and the regression's nature is linear. Important to mention that continuous variables are the variables that last forever to count, such as the age of something. It will take forever to count, because it may be: 15 years, 3 months, 9 days, 18 hours, 3 minutes, 31 seconds, 9 milliseconds, 34 nanoseconds, 50 picoseconds…etc. However, according to Joshi [158], discrete variables are the ones that are countable within a period of time, for example, the money in the bank account. The linear regression is

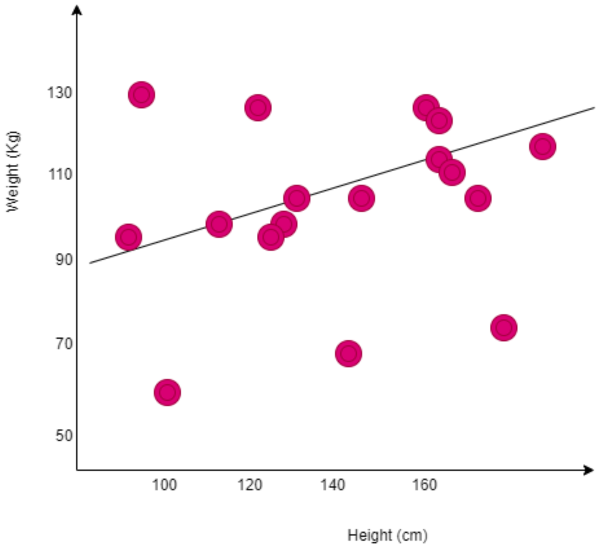


Figure 3.20: Example of Linear Regression

represented using the equation: $f(x) = Y = a + b * X + e$, where Y the target (dependent variable), X the independent variable(s), a and b are considered the intercept and the slope respectively and can be obtained by the Least Square Method. Using this equation target variable might be predicted depending on the provided predictor variable(s). Figure 3.20 illustrates this technique.

In literature, there are two types of regressions, simple and multiple. The regression is called multiple when there is more than one predictor (independent variables). However, having only one independent variable produces a single linear regression.

Logistic Regression (LR) – A statistical model, which is used in case the target variable is binary (True/False) to find its probability. The main equation can be represented as follows (Y values can be either zero or one):

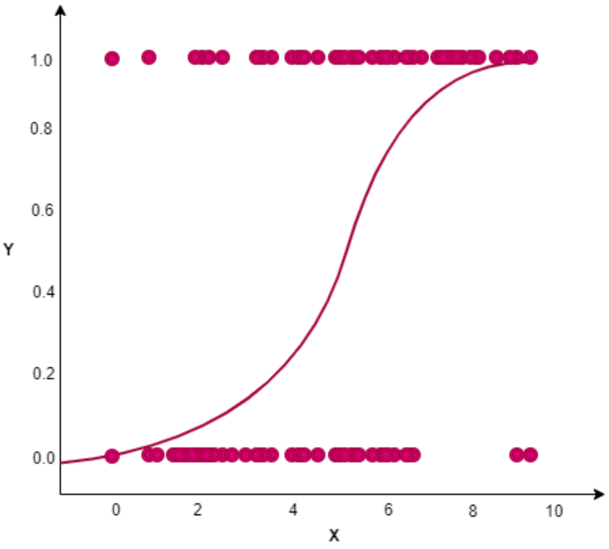


Figure 3.21: Logistic Regression Example

$$\mathrm{logit}(p) = \ln\left(\frac{p}{1-p}\right) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_k x_k$$

where x_k refers to the predictors, p for probability, and β_k stands for the model parameters. The linear relationship between target and predictors is not required, therefore a non-linear transformation is applied to handle various types of relationships. Estimating the logistic regression can be ensured by using stepwise methods, which also avoid over and underfitting. Important to mention that this regression technique requires a large size of training data. Figure 3.21 illustrates an example of logistic regression.

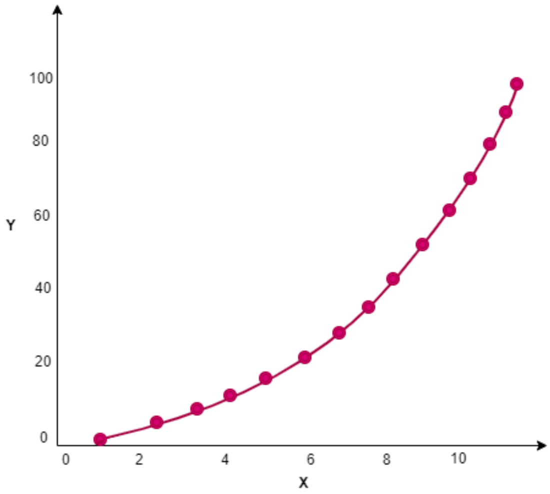


Figure 3.22: Polynomial Regression Example

Polynomial Regression – A linear regression's form, applied when the power of the predictors (x) higher than one. The relationship between the predictors (x) and the target (Y) is represented by a polynomial curve with the nth degree. An example equation looks like this:

$$Y = a + b * x^2$$

The resulted graph shows a curve instead of a straight line. Figure 3.22 shows an example of this type of regression. Attempting to get lower error rates by fitting a higher polynomial degree, may lead to weird results such as over-or under-fitting. Polynomial regression is becoming attractive in case of residuals inspection, or when curvilinear relationships will be hypothesized.

Stepwise Regression – is preferred when having multiple predictive variables. An automatic process, without any human's intervention, is applied to decide on the best appropriate predictors' variables (independent variables). In each iteration, an addition or abstraction is done to the variable based on a predetermined criterion. However, according to Flom [159], during the regression analysis, often there is a large number of independent variables, where researchers attempt to choose the best one to build the best regression model. For this purpose, they try to use some automated processes such as stepwise, forward, or backward selection. This approach is not recommended and should be substituted by other methods such as PROC GLMSELECT. This technique is controversial because the test is biased, and the created models may not reflect the real data [160].

Ridge Regression – This technique is used in two different cases: 1) in the case of multicollinearity (when predictor variables are highly correlated). 2) when the independent variables' number goes above the observations' numbers. Its main advantage is avoiding overfitting that appears when the trained model works properly on the training data and performs inadequately on the testing datasets. Ridge regression works by applying a penalizing term (reducing the weights and biases) to overcome overfitting. This may let the model perform a little poor in the training datasets, however it will perform consistently well on both the training and testing datasets. Figure 3.23 illustrates an example of this regression technique. In the figure, we see the least square regression which is drawn to best match the available training data 🔴, this line is tilled to best match both training data 🔴 and testing data 🟢. A penalty is taken into consideration when tilling the line, to avoid some training data, against achieving better results with the testing data.

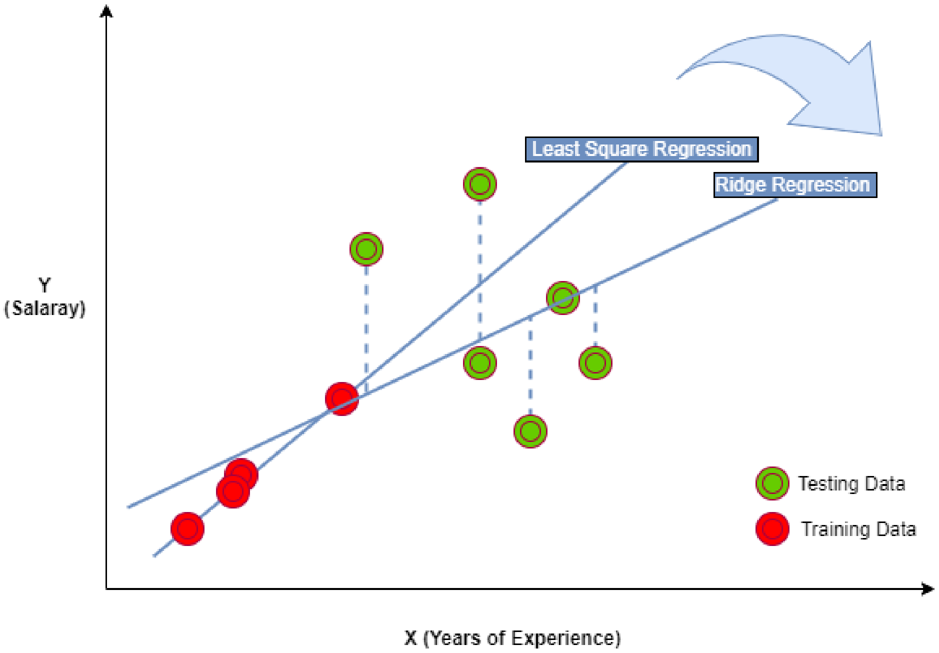


Figure 3.23: Ridge Regression Example (Salary/Years of Experience)

As a response to the multicollinearity, most of the researchers suggest the mean-centring of variables. According to Assaf et al. [161], this approach does not work. Even more, it is considered one of the greatest misconception approaches. Assaf et al. [161] recommend using the Bayesian ridge regression instead of the mean-centring approach.

Lasso Regression – is very comparable to the ridge regression, however, it differs in the way how it uses values of the penalty function. It uses the following equation: $The\ sum\ of\ the\ squared\ residuals + \lambda * (slope)^2$, however, Lasso's equation looks like $The\ sum\ of\ the\ squared\ residuals + \lambda * |slope|$. There is an essential difference between them. When the $\lambda = 0$ the Lasso Regression line will be the same as Least Squares Line, as λ grows in values the slope gets smaller until it equals zero. In other words, it can only drop the slope asymptotically near to Zero, whereas Lasso Regression can drop the slope all the way to Zero.

Elastic-Net Regression – This type of regression is the combination of both previously explained regression types. It is used when there are tons of variables and parameters. The equation looks like following $The\ sum\ of\ the\ squared\ residuals + (\lambda_1 * |variable_1| + \cdots + |variable_x|) + (\lambda_2 * variable_1^2 + \cdots + variable_x^2)$

After reviewing these different regression types, a legitimate question comes up; which one to choose? Many approaches suggest different answers, such as if the outcome is *continuous*, then the linear regression should be the right choice. Whereas, if the outcome is binary, logistic regression should be considered. In fact, the decision should be based on the number of dependent and independent variables, also the data dimensionality and characteristics. Following are the steps to choose the most proper regression methods:

1. Proper and detailed data assessment – Data must be investigated and checked to determine the nature of variables inside the data and the relationships among them.
2. Checking possible bias in the model – Applying some metrics such as AIC, BIC and Mallow's C_p which check the fit of the estimated model.
3. Cross-Validation – Occurs by dividing the data into two parties: training's data and testing's data, then applying it to different methods. A mean squared difference between observed and forecasted reveals the accuracy and the suitability of the method.
4. Judge your objective – Own objectives may decide for the most appropriate approach. Implementing a less complicated method that outputs less accurate however satisfying results may be considered than implementing a sophisticated method with higher accuracy rates, accomplished with a high implementation difficulty grade.
5. The last three methods including Ridge, Lasso and Elastic-Net are more suitable for datasets with multicollinearity and dimensionality.

Data variance is an important approach that must be considered while dealing with regression algorithms. The following deviation equation is used to calculate the variance rates among the data columns, where features with low variance rates are eliminated and removed from the dataset:

$$Variance = VAR(x) = \sigma^2 = \frac{1}{n}\sum\nolimits_{i=1}^{n}(x_i - \mu)^2$$

$i = variable\ i, n = number\ of\ non - missing\ data\ points, x_i = observed\ values, \mu = Mean$

According to Torabi et. al [162], the error of the consumption can be calculated using the following equation:

$$E = \frac{AC - FC}{AC}$$

$AC = actual\ consumption$

$FC = forcasted\ consumption$

This gives in percentage the prediction error.

Another common metric is the MSE. A statistical estimation parameter is used to measure how good an algorithm is. The MSE is calculated using the following equation:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}\left(Y_i - \hat{Y}_i\right)^2$$

$MSE = mean\ squared\ error$

$n = number\ of\ data\ points$

$Y_i = observed\ values$

$\hat{Y}_i = predicted\ values$

Squaring the difference between both values results in removing the sign, to obtain only a positive error value. It also helps to inflate large errors, which leads to punishing models with larger errors. Another important metric in this field is the RMSE.

$$RMSE = \sqrt{\sum_{i=1}^{n}\frac{\left(\hat{Y}_i - Y_i\right)^2}{n}}$$

$RMSE = root\ mean\ squared\ error$

$i = variable\ i$

$n = number\ of\ non - missing\ data\ points$

$Y_i = observed\ values$

$\hat{Y}_i = predicted\ values$

Because this metric is calculated, it will have the same unit of the predicted value, which is kWh. So, it differs from the previous MSE approach because it gives the exact unit, not its squared value. Another popular metric is called MAE stands for Mean Absolute Error. It shares the same attribute as RMSE because it carries the unit of the predicted variable, however in MAE changes are linear and therefore intuitive. Unlike MSE and RMSE, MAE does not punish models with large errors, because it does not square them. Errors increase linearly as errors increase. The following equation was used to calculate it:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|Y_i - \hat{Y}_i|$$

$MAE = Mean\ Absolute\ Error$

$i = variable\ i$

$n = number\ of\ non - missing\ data\ points$

$Y_i = observed\ values$

$\hat{Y}_i = predicted\ values$

3.3.8. Time Series Algorithms

History events repeatedly reoccur most of the time, therefore events that occurred in the past, are probably will occur once again. Time series data are the recorded observations done at regular periods. Time Series watches the data continuously to estimate and predict events that will take place in the future, based on patterns retrieved from prior times. A common basic example is seasonal sales profits. Yearly within holiday periods, sales profit increases, however, within periods outside seasons, profit decreases. Predicting this situation is not difficult because almost what is to come every holiday season can be expected. Also, other time-based trends in the data can be obtained, such as a steady growing trend or improvement in a company's operations, or a decreasing trend, where the company consistently falls each year, or month or day. According to Chen et al. [163] In the world of Big Data, huge data are continuously produced. Data retrieved from meteorological and financial sectors are among the data examples that have a periodicity character. finding the possible periodic patterns within this huge time-series data, and providing precise predictions, is a critical mission. Therefore, Chen et al. [163] have introduced and implemented the *Periodicity-based Parallel Time Series Prediction (PPTSP) algorithm* in a huge amount of time-series data, using Apache Spark technology running in a cloud-computing environment.

The fact that prediction is based on inspecting the sequence of observations in a given time, is considered the major difference between this approach and other predictive methods. A Time-Series forecast can give a better-estimated figure for how much it could continue. For example, a Time Series model predicts 100,000 people to log in online. It might be known there would be a lot of people online, and now it is possible to plan for how many further servers and infrastructure are needed for your online platform, based on the amount of predicted online users. Or it could be that the model predicts a million users in the upcoming years, significantly expanding from last year and even more so the year before then. When reaching a point of continuing significant growth, a decision might be taken that it is the right time to invest in better infrastructure for the year ahead and coming years ahead. Another example is having a sensor device recording the number of vehicles that cross an intersection every 20 minutes. These counts of vehicles can be utilised to predict that in the next 20 minutes, traffic at the intersection is likely to spike to a huge amount. So now maybe a trip planning App could re-route drivers to avoid this congested, problematic intersection, distributing the traffic load more evenly across roads.

It is also possible to model data with no recognizable pattern or trend in Time Series. When there is a trend or pattern, inspect the whole history of data and see that pattern occurred over time. However, if there is no recognizable pattern, then the best bet is to rely more on what is recently happened and less on what has happened far in history. What is happened recently is more useful in guiding to what will happen next than to look back at the whole of history, which only shows pretty much anything could happen.

3.4. Cloud Computing

Cloud computing simply explained, is having computer resources, storage, networking, applications, and analytics available on-demand without being anchored in a physical location, offering the ability to scale elastically. This attribute is exactly the suitable one for any system or business that support scalability, which is exactly the case for the proposed Integrated Scalable System for Smart Energy Management (I3SEM). Cloud computing is part of everyone’s life, starting from sending emails, editing online documents, streaming films or music to online gaming most of the modern offered services are likely based on cloud computing. This approach is not only attractive for small entrepreneurs’ companies, also for profit and non-profit organisations. The usage of cloud computing can be summarized in several categories such as creating cloud-specific applications, storing, backing up and recovering data, streaming music and movies, delivering software, testing and building applications, analysing data and finally embedding intelligence.

Cloud computing types are divided into three main categories: public cloud, private cloud, and hybrid cloud. Public cloud, such as Microsoft Azure and AWS are clouds offered by third-party partners who are responsible for the whole infrastructure including hardware, software, and networking. Clients usually access their part via client-side applications such as browsers. However, private clouds, are tailored and dedicated only for a particular client and exclusively used by this particular client. It may physically be on the client's side or somewhere in the third-party data centres. Lastly, hybrid cloud computing is a combination of both described clouds, with a special communication software that allows the full integration of them. The possibility for data and application to existing in both clouds generates a huge opportunity for businesses to be scalable, gets better deployment opportunities, gain more security and complicity. An example of such infrastructure is used during the development phase to develop and test using the private cloud, then deploy the release to the public cloud. The benefit here is having identical environments which increase the likelihood that newly developed software will also run in the public domain the same way it runs in the private zone.

According to Microsoft Azure [164], cloud computing has massively changed the classical way how business deals with IT, this change has been supported by the following factors: First – dropping the initial costs to buy hardware and set up the software, and getting rid of continuously running costs such as electricity and the costs of experts to manage the datacentres. Second – scalability is guaranteed by offering more or less componential, storage and networking power tailored according to instant needs. Third – Performance is taken seriously in the state-of-art cloud computing datacentres which are always equipped with the latest advanced hardware and software versions. Fourth – High levels of security are implemented by applying strict policies and tools. This approach ensures better protection of both data and application. Fifth – Delivering needed resources just with a few mouse clicks and within minutes ensure the rapid response to the client needs and minimize the tension of resources planning. Sixth – Enhancing productivity by shifting the whole infrastructure administration and management from the local IT team to the cloud. This approach frees up the local resources to deal with more specific and business-related tasks, and ultimately enhance their productivity. Seventh – cloud computing is reliable because it always offers additional backup routines, disaster mitigation procedures and business recovery and steadiness, moreover the cloud data centres are physically distributed on various locations, so when a particular location faces disasters (such as hurricanes) other datacentres where the data is replicated can continue to serve the business uninterruptedly.

Cloud computing offers different services or stacks, these can be categorized into seven main groups: Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS), Software-as-a-Service (SaaS), Functions-as-a-Service (FaaS), integration-Platform-as-a-Service (iPaaS), Identity-as-s-Service (IDaaS) and Serverless computing. In details:

Infrastructure as a Service (IaaS) – Considered the most basic and most used service. Simply it offers all IT related infrastructure such as componential resources, highly scalable databases, private or public networking capabilities, big data management, machine learning algorithms software, hardware, operating systems and monitoring on a pay-per-use-as-you-go basis. Famous examples of such services are offered by Amazon Web Service (AWS), Microsoft Azure, Google Cloud Platform, and IBM Cloud.

Platform as a Service (PaaS) – It offers a collection of services explicitly dedicated to the use of developers, who can utilise tools, applications, and APIs to improve the development, testing and deployment quality and speed. Salesforce's Heroku and Force.com are well known for open cloud PaaS contributions. PaaS can guarantee that developers have restricted access to a collection of assets, follow predefined exact processes, and utilise just a particular set of instructions.

Serverless Computing (SC) – On one hand, it is similar to PaaS in many aspects, for example, developers are only responsible for writing their code, and there is no need to manage the server. However, on the other hand, there are some differences on various levels. First – PaaS offers more control over the deployment environment, whereas Serverless computing offers less control. Secondly – in PaaS applications must be prepared and configured to support the automatic scalability, however, application in SC supports the automatic scalability without previous configuration. Third – the code in the SC environment will be executed only after invoking.

Functions as a Service (FaaS) – This is the serverless computing version dedicated for use in the cloud. It is an enhancement of the Platform-as-a-Service because it adds another abstract layer on top to isolate developers completely from the environment where they run their code. Developers are only required to push their code then trigger a process, without having to deal with any VMs, containers, or run-time applications. This kind of service is offered by most vendors such as Google Cloud Functions, AWS Lambda, Azur Functions, and IBM OpenWhisk.

Software as a Service (SaaS) – This kind of cloud computing conveys software applications over the internet through a browser on demand using their desktop or mobile. SaaS applications offer broad configuration possibilities of their development platform enabling them to apply their own alternations, while still being responsible for applying all necessary updates, patching and maintenance. The most famous SaaS applications for business can be found in Google's G Suite and Microsoft's Office 365.

Integration Platform as a Service (iPaaS) – Adapting SaaS in a business environment usually requires using this iPaaS to ensure an appropriate data integration through using compatible data connectors. This is widely used in Business-to-Business environments and e-trade because it allows clients to implement data matching and workflows within their integrative process. Dell Boomi, Informatica, MuleSoft, and SnapLogic are among the providers of such a service in the market.

Identity as s Service (IDaaS) – The appropriate recognition of the users' identity and assignment of permissions and rights, is considered one of the most challenging and difficult tasks. IDaaS is providing a reliable way to deal with such risk. Users' profiles are kept and maintained to provide authentication for users to access various applications based on predefined access policies and user groups. These services usually offer integration possibilities with directory services such as Active Directory, LDAP. Okta, Centrify, IBM, Microsoft, Oracle, and Ping are some examples of such service providers.

Both cloud computing and the Internet of Things (IoT) are strongly coupled technologies within the wireless communication area, where the growth and enhancement of each one lead to support the enhancement of the other. According to Stergio et al. [165], this side-by-side development brought some security challenges, and this should be solved by applying different encryption algorithms, such as AES and RSA. Stergio et al. [165] also suggest the contributions of cloud computing in the IoT world by mentioning some examples, such as sensors in building using the cloud computing to store data, and to call services, also the computational resources. Where the remote monitoring of patients relies on the services and applications retrieved from the cloud. Both IoT and Cloud Computing fill gaps in the other part, IoT fills gaps in

CC by expanding its limitations and boundaries to enhance the scope, on the other hand, CC fills some gaps of IoT related to the limited storage and applications over the internet. Moving IoT applications to the cloud requires closing a contract in the form of a Service Level Agreement (SLA) to ensure mainly the application's availability and other essential attributes.

However, besides the tremendous advantages of cloud computing platforms, there are a number of disadvantages that can be summarized as 1) Network connection is always needed – The cloud computing services are all offered only when the client is connected to the internet. There is almost no business continuity plan when the connection is cut due to an outage or storm. 2) Features depend on the provider and its speciality area. Not all providers specialized in all types of services and applications, some of them offer excellent storage applications with poor operating systems varieties. A detailed look and decision matrix must be put in place to make the right decision. 3) The absolute reliance on the provider to maintain the system and the data, the client has no control and must trust the provider. In some cases, this may bring hazardous consequences. 4) It is a fact that not all providers are secure as they claim. Security is an essential aspect and may suffer in the cloud computing environment, compared to the closed local data centres. 5) No control in case having technical issues. Clients cannot do anything other than call or open a ticket and wait for somebody else to solve the problem. Besides all of that, there are some challenging contract and political issues [166] [167].

3.5. Internet of Things (IoT)

According to AWS Amazon [64], "IoT is a system of ubiquitous devices connecting the physical world to the cloud". IBM [168] defines IoT as "At the heart of IoT are the billions of interconnected 'things or devices with attached sensors and actuators that sense and control the physical world". It can be also defined as "the network of physical objects, devices, vehicles, buildings and other items which are embedded with electronics, software, sensors, and network connectivity, permitting these objects to gather and interchange data" [169], [170]. This type of connection leads to constructing smart entities, whether homes, cars, cities, industry fields, and ultimately a smart world [171]. A scenario inside a smart home environment can deliver a good example to explain this approach; let us consider the ringing alarm in the morning that sends a signal to the coffee machine which begins automatically preparing the coffee without the need for any human intervention [171].

Cisco Inc. expects that by 2020 there will be 50 billion connected devices [172], therefore it is obligatory to consider a few factors while implementing any IoT system; such as complexity, security, privacy and data storage, safety and standardisation [171]. **Complexity** – Having a large number of devices, services, link-layer technologies, etc. mutually connected, dramatically increase the complexity grade of the system [171]. The complexity's sources are the management efforts, the frequent maintenance and the handling of generated huge data and ensure compatibility. **Security** – Moreover, opening communication gates and channels among devices bring with it some security challenges. One of the proposed solutions to address this issue in this project is implementing the industry-standard protocol for authorisation OAuth 2.0 [173]. **Privacy and data storage** – Because sensors are an essential part of any IoT system, thinking about handling data storage and privacy is a must. Wireless Sensor Networks are proposed as an appropriate solution for this purpose. In these types of networks, data gets shared within sensors farms and ultimately sent to distributed systems to analyse [174]. **Safety** – The nature of the topology of IoT requires having event-driven smart applications that interact with the connected devices, sensors and actuators, however these applications may not always properly function, and can be buggy, or perform inappropriate interactions, or suffer from telecommunication deficiencies, all of these could cause insecure and risky situations. For example, unlocking the main entrance door in an empty house, or even turning off the heating system during freezing periods while people sleeping at home. As an attempt to prevent such buggy behaviour a novel system called IotSan is introduced by Nguyen et al. [65]. The system attempts to identify all possible events, that could bring the system to perform unsafe actions, which occurs by revealing the weaknesses on the interaction tiers. A case study has been carried on using Samsung SmartThings, this revealed that the IotSan could detect 147 vulnerabilities on 76 systems. **Standardisation** – Similar to any technology, where a huge number of vendors, devices, protocols and applications are involved, having a set of standards encourages rapid development, simplifies the manufacturing process, also advances the manufactured appliances due to specialising in building particular modules according to predefined and agreed-upon standards.

3.6. Summary

Most related techniques, technologies and paradigms which are considered the main pillars of any smart management system were reviewed in this chapter including Microservices' definition, structure, application fields and the main advantages compared to other paradigms. This was followed by a number of data analytics and data mining techniques covering supervised, unsupervised and semi-supervised machine learning techniques, tracking patterns, classification, association, clustering, regression, time series and anomaly detection. Cloud computing was explained due to its capabilities of supporting and enhancing the scalability and performance issues while implementing any energy management system. Moreover, the chapter covered all related issues to the Internet of Things approach because any modern smart energy management system should deal with both conventional and smart appliances, where IoT entities are considered essential for supporting the integration of these appliances. Together with the techniques and technologies, a number of selected smart energy management systems were reviewed as illustrations where these technologies are applied.

Technologies are described in detail to make the implementation phases faster, easier, and more reliable due to the clearness and the deep understanding of each piece and its capabilities and drawbacks to achieve the most appropriate choice. It also assists in building a sufficient and reliable decision matrix when needed to compare different technologies to choose the most suitable and appropriate one, without having a clear picture of all available technologies and their attributes, it is not possible to hold any reliable comparison. As will be seen later in the case study chapter, not all of them will be considered in the implementation, however, these are reviewed to make sure that only suitable ones will be chosen and implemented and give clear arguments why some technologies were not used in the implementation by describing its capabilities and boundaries.

4. Integrated Scalable Smart Energy Management Framework

4.1. Introduction

Both previous literature review chapters reviewed a number of related techniques and technologies, conventional and smart energy management systems which are proposed and constructed by researchers and industry to achieve the challenging question: how to reach the most efficient energy consumption while keep offering the targeted user's comfort. A closer look at these systems reveals some drawbacks and deficits that have been addressed in detail in section 2.7.5. The framework presented in this chapter is an attempt to take a step towards having an integrated system to overcome some of these shortcomings and deficiencies by proposing a new architecture consisting of new components or remodelling existing ones covering many aspects, such as scalability, reusability, pluggability, security, enhanced user experience, mobility, more extensive stakeholders' engagement. Although the lack of standards plays an essential role in slowing down the development of any system in any field, not only in the energy management sector, this aspect is not addressed and considered in other sectors as well..

This chapter presents the proposed framework in detail. It begins with a detailed review of quality factors that must be met in the framework in order to sustain for a long time and establish a proper basis for any future enhancement and development. Since data plays an essential role, section 4.3 will be dedicated to examining and assessing all framework data's related challenges including data integration and processing, handling a huge amount of data, data privacy and protection, and data's real-time evaluation and visualization.

The in-depth explanation of the framework begins with a high-level view and mentioning of the functional requirements which should be fulfilled by the framework. Then it is followed by an explanation of the components and modules. The followed terminology suggests dividing the framework into three main zones: client zone, cloud gate and cloud processing zone. Each zone consists of a number of components that put all related modules, processes, services or devices under one umbrella to enhance modularity, scalability and security. The data flow in the framework is touching nearly every module therefore in this chapter three main workflows will be described and explained to illustrate how data is handled and dispatched. Moreover, this chapter covers all aspects that will be used in the next implementation and evaluation chapter by providing a detailed explanation and definition of boundaries and frames of each component, module or service that aims to achieve many goals: Firstly, reducing or even eliminating any unexpected issues or deviances during the implementation phase. Secondly, assist in deciding on the most appropriate applications, techniques, and technologies. Thirdly, reduce the implementation phase duration by fulfilling some modules or services by ready-to-use, off-the-shelf software pieces such as security modules, alerting management systems, or front—end templates and visualisation engines.

One of the expected benefits of the framework is measuring and reporting the amount of the consumed energy within certain times during the day, within days during the week and within seasons during the year, together with some anonymous data related to the occupants, living area size, region, weather and living area's type (apartment, house, villa, organisation, …). Processing this data and offering it anonymously to stakeholders such as local energy generators may open avenues toward better energy generation management and reduce the lost energy during off-peak slots. Moreover, governmental agencies may use this data to be more accurate while designing and issuing new laws and regulations. It also gives appliances' manufacturers the chance to retrieve various running parameters of their devices while operating under real-life conditions.

4.2. Quality Factors

Energy Management Systems are complex structures in nature, in order to sustain for a long time and establish a proper basis for any future enhancement or development they should match a variety of quality criteria such as scalability, reusability, security, data integrability and interoperability, for instance, different smart home protocols such as ZigBee and Z-Wave operate simultaneously. As seen in the previous literature review section, none of the reviewed frameworks managed to address all mentioned quality factors. The proposed Integrated Scalable Framework (I3SEM) is an attempt to consider and address these quality factors during its design and implementation phases.

Achieving a high level of scalability in any system is not a bonus feature, rather an essential quality factor, which determines and maintain its lifetime value and reduce needed resources in the long term. In many cases, scalability is ignored or shrank to low levels due to many constraints related to budget, time, resources and customer requirements. A scalable system is a system that keeps being performant without the need to redesign any part of it regardless of the growing workload which is represented by anything that goes beyond the system's limitations such as the increase of simultaneous user access, number of transactions, storage capacity. Due to the nature of the field where the proposed I3SEM framework will be implemented, it must consider a high degree of scalability to allow a smooth and cost-efficient deployment of the system when applying it in a broader range of facilities.

Performance, in general, is a loose term. It can be concreted by breaking it down into three main measurable factors: 1) Response time – is representing the take needed by the system to process a given stimulus or event, keeping this time shorter, will decrease the performance of the system and improve the overall user experience. 2) Throughput – by definition, the throughput represents the amount of data or work-units passed or processed by the system. Here it is essential to differentiate between throughput and bandwidth; where bandwidth represents the maximum allowed capacity of the system, the throughput reflects the currently used amount. 3) Utilisation – A resources utilisation unit plays a pivotal role to track and recording the resources' occupation level while running a performance test. It helps to figure out the bottle-neck spots in the system — the next section, *4.2.2. Component Overview* will describe in more detail the way how the proposed I3SEM framework is planning to deliver responses to enhance the performance factor.

Reusing previously written, fully functioning, tested and debugged a piece of codes, components, or modules, does not only enhance the productivity it also increases the quality of the whole software by avoiding programming new code, which ultimately brings too many advantages related to the development time, budget, resources. Basically, a piece of code is considered reusable when it can be used, without any changes, to achieve several targets. In fact, the reusability factor relates to the code and the used programming language, which cannot be directly influenced by the I3SEM framework, however, using a new paradigm such as microservices implicitly indicates the necessity to develop code with a high grade of reusability. More details on how microservices might increase the portion of the reusable code within the system will be seen in the next sections.

Offering a high grade of integrability and dynamic aggregation possibilities to enlarge the system by continuously adding new units, or involving new households, brings enormous security challenges to keep the whole system distanced from hijacking and undesired attacks. Different security approaches and arrangements must be offered within the proposed I3SEM framework to secure both local aggregated mesh networks inside the household, and the overall framework because of several reasons. These reasons can be summarized as 1) The nature of the proposed framework which is based on offering integration and scalability possibilities to enhance one household's network by adding an unlimited number of appliances, sensors etc. 2) The fact that the framework designed to support managing an unlimited number of households within a community. 3) The use of cloud-based solutions adds new tasks to verify upcoming requests and apply extra permissions mechanisms. Many security concepts will take place in the proposed framework to respond to the possible threats.

One of the significant challenges faced in this area is the lack of standards. This has a direct impact on nearly everything, starting from the communication protocols, ending up with the transferred data packets. Because many vendors, protocols and devices will be used in the local mesh network, the proposed framework must have a Data-integrability component to match the different data types resulting in having smooth communication among them regardless of the different data format. Such data-integrability components support the interoperability competency of the system. In other words, if a new device, sensor, or protocol which supports a different type of data, added to the system, all that needs to be done is add a matching data- integrability module to match the new format. More specific details will be explained in the coming sub-sections.

Designing and developing an attractive, user-friendly system encourages stakeholders to interact with the system and use it more frequently. Achieving this goal requires following several steps: *firstly*, analysing the audience – knowing the target audience allows offering the proper GUI that match their needs and expectations. *Secondly*, keeping it as simple as possible – simple, clean and minimalistic are the essential keywords to describe proper software. All well-known applications have this attribute.

Thirdly, minimal interruptions – users do not like being interrupted while using an application, therefore some techniques such as implementing self-disappearing notifications, or inline error messages must be considered. *Fourthly*, offering online help – no matter how the application is self-explaining and easy to use, there will always be users who ask questions, offering an online alternative to get information is much better than forcing them to write emails or call support hotlines. *Fifthly*, studying and analysing user behaviours – this is one of the modules running in the user-side to collect data resulting from the user usage, behaviour, clicks, to build complete user journeys to have an in-depth understanding of how users interact with the system. Since the first mentioned four points are application-specific aspects, the proposed framework will not have any influence on it; however, the last point can be implemented by adding a dedicated component to track users behaviours and log this data as a preparation to build complete user journeys. The component is called the *Logging User Journeys' Module*.

4.3. I3SEM Data Challenges

As discussed in the previous section (2.5.2. Data Storage), the non-relational database is considered the most suitable choice to manage the data in the proposed I3SEM. In this section, this decision will gain more clearness by illustrating the special data challenges associated with this type of research. Challenges can be summarized as:

1. **Data Integration Checking and Processing –** Due to the fact that 1) data will be retrieved from different types of sensors, and providers. 2) new sensors and providers may be added, even the same sensors may get enhanced, or external data providers may modify the outputs. 4) data may get corrupted while generated or transferred. Various cleaning-up, data integration and homogenization steps and routines must be applied to enhance the data quality and ensure its robustness and correctness.
2. **The planned huge amount of data –** Data will be collected from an enormous number of units. Accumulatively, this will generate enormous data records which must be transferred to the cloud. This puts enormous pressure on the infrastructure and the processing and storage units.
3. **Data privacy and protection** – Some data is considered private and therefore it should be handled properly and securely. An example is the data coming from detection sensors which tells whether if someone is in the house, this kind of data can be misused to commit some burglary crimes.

4. **Real-time evaluation and visualization –** results, predictions, visualisations need to be illustrated and updated accordingly.

Data mining is essential for the proposed I3SEM framework. In the literature, several techniques and algorithms can be used for processing and predicting purposes. In the next section 4.4, together with the other core components and modules, the related analytics components, and the way they are connected will be illustrated and explained.

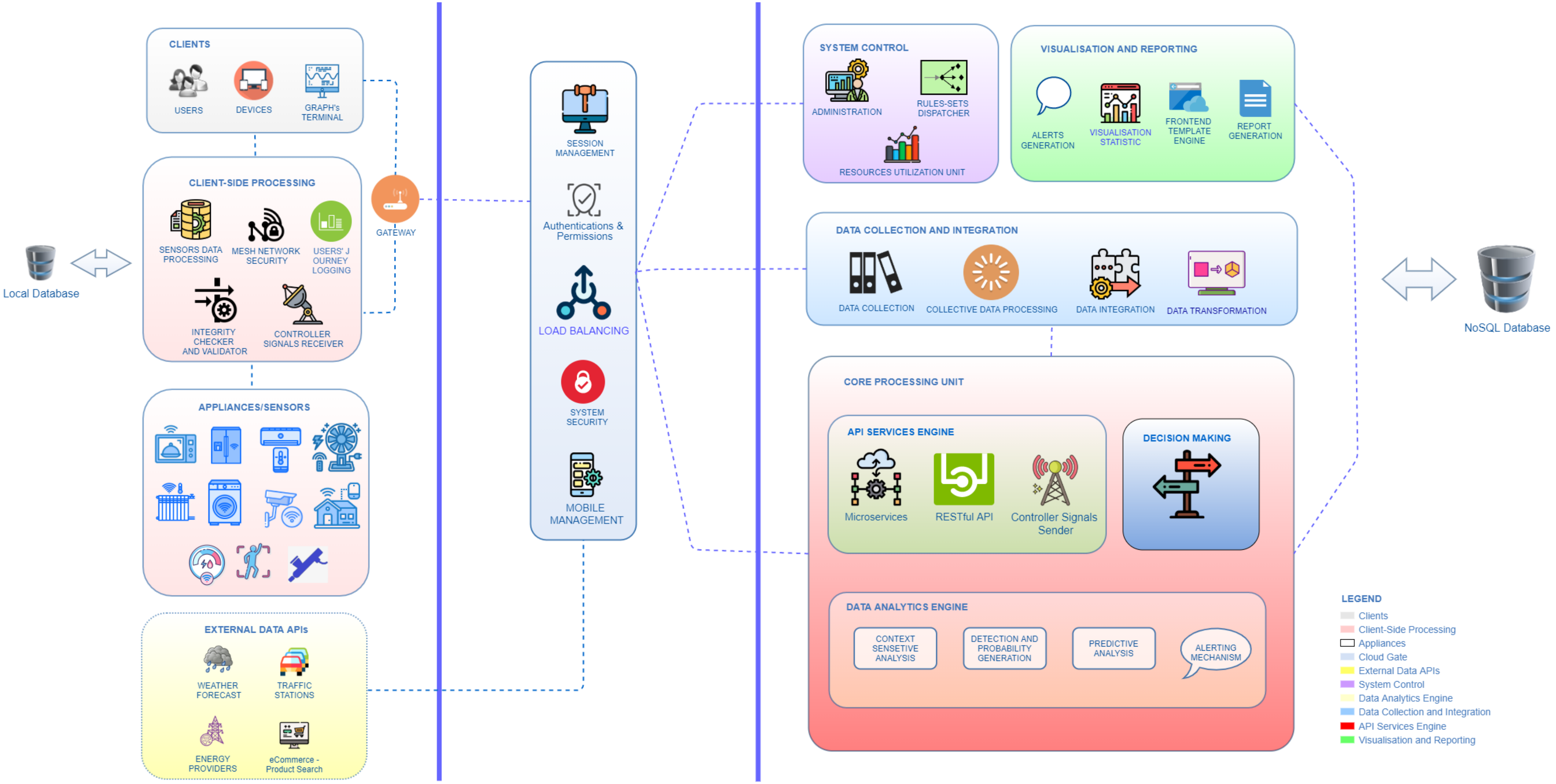


Figure 4.1: The proposed Integrated Scalable System for Smart Energy Management Framework (I3SEM)

4.4. Actual Architecture

4.4.1. High-level view

The proposed Integrated Scalable System for Smart Energy Management Framework (I3SEM) framework is illustrated in Figure 4.1. As seen, the framework consists mainly of three main zones: client-side, cloud gate and the central system. All these main zones and their components and modules will be described in detail in 4.4.2. Three main motivations stand behind designing the framework in this way; firstly, attempt to provide answers for the deficits of the reviewed frameworks from literature. Secondly, fulfilling a number of non-functional requirements is discussed in section 4.2. Thirdly, delivering responses to the following functional requirements: 1) data should be gathered periodically, upon need or action. 2) collected data must be processed; the outcome will be saved in central storage. 3) sensitive raw data will not be transmitted outside the local network (client zone), however, the processed anonymous data can be transferred. 4) readings should be used to generate visual charts, graphs. 5) all generated graphs should be accessible by end-users via mobile applications. 6) stakeholders' interfaces should interact with the server using web techniques that consume web services. 7) user management GUI should allow users to perform different tasks based on their privileges.

The next section will provide an in-depth explanation of all components and modules that construct the proposed architecture.

4.4.2. Components' View

This section will highlight the I3SEM Framework different zones, components and modules. It consists of three main zones: Client Zone (**CZ**), Cloud Gate (**CG**), and Cloud Processing Zone (**CPZ**). Following is a detailed description of these zones and their components and modules:

1. **Client Zone (CZ)** – This zone includes all components, modules and APIs that are interacting with the cloud. Some of the components are physically located in the smart home side, some of them in the external APIs. It consists mainly of five different components as follows:

1.1. **Clients** – Both end-users and devices such as tablets, desktops and mobile Apps, are considered clients of the system. Devices contain adjusted versions of the graphical user interface used by end-users to interact with the system. It is designed to operate within usual platforms such as tablets, desktops and mobile phones. A determining differentiation is offered for both users and administrators, depending on their contributions and functions.

1.2. **Client-Side Processing** – Every household consists of an aggregated mesh network of unlimited units such as sensors, appliances, gadgets. Additionally, the network will have further components and modules to provide necessary factors; these can be summarized as follows:

- *Sensor's Data Processing & Integrability Checker and Validator*: This checker and validator module plays an essential role in reducing the total amount of data sent from the household to the main processing and storage system in the cloud, in other words, only relevant data will be carried over. This approach helps in decreasing the unnecessary traffic and decentralise -partly- the processing efforts. The checker consists of a set of rules that can be applied straightforward using the local central processor; this rules-set will be managed by another central part called *Client-Side Rules Set Dispatcher* upon need. The biggest winner to this extent is scalability and performance. With this approach adding new households, implicitly adds new resources to the whole system, by reducing the 100% reliance on the central processing. Same time, the performance increased because of the reduction of the very expensive throughput, which reduces the processing efforts needed to handle it.

- *Mesh Network Security Layer*: This offers security implementations to the household mesh network. The household network can be hijacked by stealing any device (outside cameras, for instance), or by registering an unauthorized device or sensor. Every single unit in the mesh aggregated local network is essential for the whole system; therefore, securing these nodes and the flow of data among them is one of the framework's critical missions. The security is guaranteed by using the Z-Wave protocol which comes with several built-in security features such as AES symmetric block Cypher algorithm using 128-bit key length, the End-To-End

Security offered on the application level, and the use of Single Network Wide Key to offer protection against the physical attacks, creation of zombie computers by infecting the system by malware and viruses. According to Daser David [175], data cryptography represented by the order illustrated in Figure 4.2, Wi-Fi Protected Access (WPA) and WPA2, and Advanced Encryption Standard (AES) are examples of the strategies and approaches used to enhance the wireless mesh networks inside households.

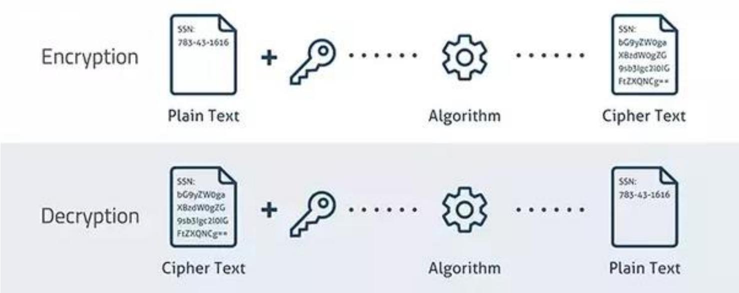


Figure 4.2: Sample of Encryption and decryption process [175]

- *User's Journey Logging Module*: This is responsible for collecting the user's behaviours while using the system as a preparation to build complete user journeys. This approach is crucial to judge the suitability of the application and see the possible improvement potentials to make the system even more friendly and usable.
- *Controller Signals Receiver* – All communication between the central processing unit in the cloud and all clients are established and performed via this receiver unit which is interfering with the signal's sender located in the core processing unit in the cloud.

1.3. Appliances and Sensors – The wireless mesh network consists of many hardware components such as sensors, appliances, gadgets and some other devices as routers, or any additional Network Attached Storage (NAS) devices. Devices must support one of the available major smart home protocols: ZigBee and Z-Wave. Technically it is not possible to have them both communicating with each other due to differences in the frequency.

1.4. External Data APIs – Some decisions will be taken based on additional external feedback such as weather forecasts, traffic or energy providers' prices, and the eCommerce external product API which is used by the decision-making component in the core processing unit.

1.5. **Gateway** – This hardware device (router) is a network node that is standing between both networks, the local home mesh network and the internet. Its main function is enabling the IN and OUT traffic in both directions.

1.6. **Local Database** – The local system requires a storage medium to keep the collected raw data from the local network together as a preparation for further processing. The local database will be responsible for saving collected parameters, processing results, APIs raw data, and various logs.

2. **Cloud Gate (CG)** – All traffic going IN and OUT of the cloud must first go through this gate from both directions. All requests and responses are managed, secured, authenticated and load-balanced through this gate. Such a paradigm allows more control and better management to handle and classify the traffic. It consists of the following five different modules:

2.1. **Session Management Module** – This module handles and manages all requests coming from a user or entity after the authentication process. It plays an essential role to enhance performance, privacy.

2.2. **Authentication and Permissions Module** – This component resides in the cloud entry point to serve two essential goals: firstly, protecting the application and the data against any unauthorised access. Secondly, managing the access permissions of users based on their privileges and roles. All subsequent requests will be validated, checked, and verified.

2.3. **System Security Module** – This part includes off-the-shelf security components such as firewalls, Secure Socket Layer SSL encryption and Transport Layer Security (TLS).

2.4. **Infrastructure and load balancing Module** – Per definition load balancing aims to distribute the tasks on the available resources in a way to prevent them from being under or over-utilised. This concept is considered one of the most critical techniques to improve the scalability and performance of any system. There are many approaches for this purpose, such as the Honey Bee Approach done by [176], Ant Colony Approach by [177], Genetic Algorithm, Biased Random Sampling. However, the approach Weighted Biased Random Walk (WBRW) recommended by Jain and Kumar [178] illustrated in Figure 4.3 will be used in the cloud environment in the proposed framework.

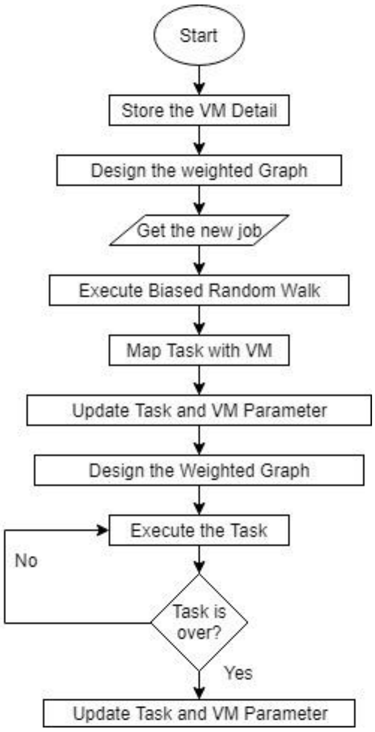

Start
Store the VM Detail
Design the weighted Graph
Get the new job
Execute Biased Random Walk
Map Task with VM
Update Task and VM Parameter
Design the Weighted Graph
Execute the Task
No
Task is over?
Yes
Update Task and VM Parameter

Figure 4.3: Flowchart of Weighted Biased Random Walk

2.5. **Mobile Management** – Together with the mesh network security module located in the client-side processing, this module assists to manage and securing all physically portable mobile devices to ensure high-security levels and prevent potential attacks. Section 5.6.3 proposes an illustration example.

3. **Cloud Processing Zone (CPZ)** – The main and central zone that consists of many components and modules where the data collection and integration, predictive analysis, context-sensitive analysis, decision making, visualisation, and reporting and alerting take place. A detailed overview of this zone can be found below:

3.1. **System Control Component** – Having a general system resources overview and establishing add-on components to react to the rapidly changing running environment parameters, leads to better performance and higher reliability grades. In the proposed framework, this approach is introduced by the following units:

- *Administration* – Constants, constraints, rulesets, users and all other pre-defined parameters can be handled and managed using this module.

- *Rules-Sets Dispatcher* – A central unit to dispatch all rules defined by administrators to all clients (households).
- *Resources Utilisation Unit* – This is a performance key indicator added to the framework to track how busy various resources of a computer system are when running a performance test or during the usual operational phases. Defining the most used resources allow system administrators to take suitable decisions to enhance the performance by adjusting the resources accordingly.

3.2. Visualisation and Reporting Component – This component is considered one of the framework's core components. Generally, it covers all aspects related to the visualisation and reporting of the processed outcomes, and selected data portions. It consists of the following modules:

- *Front-End Template Engine* – This module represents an additional step towards offering more modularity, flexibility and interoperability to the framework by separating the data from the layout templates. The idea is based on pre-designing templates with a pre-fixed data structure represented by XML or JSON formats, and front-end technology, such as HTML and CSS. The Engine's primary function is merging both templates and data sets to generate the final output (as HTML pages for the browser), Figure 4.4 illustrates this principle.

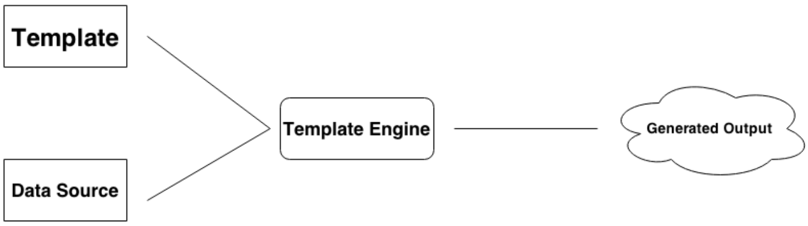


Figure 4.4: Frontend Template Engine Principle

- *Visualisation Statistics* – Statistics are generated and saved as raw data. The final view of data may be shown using any common, off-the-shelf platforms such as Grafana.
- *Alert generation* – The process starts by defining rulesets, it ends up by sending out alerts in many shapes, once the predefined rulesets found a match among the processed data.

- *Reports Generation* – Different types of reports for different stakeholder types will be produced. Additionally, reports can be provided to local energy providers, appliances' manufacturers, and any interested local governmental agencies.

3.3. Data Collection and Integration Component – All collected data retrieved from the various hardware units and external services will be collected and managed by this component. It has many modules to fulfil this purpose:

- *Data collection* – Data is collected, validated and stored.
- *Collective Data Processing* – Both sensor generated and external API data will be checked and validated, then processed to get the most relevant and useful part of it. The obtained information then gets saved to the storage unit.
- *Data integration* – Since data are coming from different resources, there is a need to harmonise and offer data integration to allow translating different data formats into a standard processable piece of data. This approach in the proposed framework enhances the interoperability, consistency, and accuracy of data stored in the database, it also allows handling different types of data collected from various hardware vendors.
- *Data Transformation* – Collected data may have different types, such as texts, timestamps, numbers or even images. Therefore, there is a need to transform the various data formats into a processable shape. This approach supports the strength point of the framework to deal with smart and conventional appliances. Where smart appliances can directly generate processible data, and images can be taken from conventional appliances settings panels then converted to processible ASCII formats using these data transformers. This approach can be seen in the case-study chapter in section 5.3.

3.4. Core Processing Unit – This is the central unit of the cloud. It is responsible for getting, processing, analysing, and storing the incoming data, besides making decisions based on the collected and predicted data which is retrieved from the Data Analytics Engine. It is divided into three main components: API Services Engine, Data Analytics Engine, and Decision-Making Module. Following is a detailed description of each.

3.4.1. API Services Engine: Consists of the microservices farm, the RESTful API and Controller Signals Sender.

- *Microservices* – This module consists of a highly maintainable and testable, loosely coupled, independently deployable collection of services which will be written in Java as explained in the case study chapter and reside behind a RESTful webservice. In the proposed I3SEM framework, this module supports several quality factors such as scalability by offering Platform-as-a-Service (PaaS), and Function-as-a-Service (FaaS). According to [179], microservices assists in achieving high grades of performance due to the efficient use of available resources, and the performant way of handling processes tasks. Moreover, the reusability is highly supported by this module because one microservice (for instance; the logging) can be re-used in many other microservices. load
- *RESTful API* – According to [180] REST is neither a protocol nor a specification; it is an architectural style of networked systems focused on exposing the resources on a network, mainly Web. In the proposed I3SEM framework it offers all functions needed to communicate with all households' networks and the other external services. Applying this approach in the framework supports scalability and data- integrability by offering a single point of access to a scalable microservices farm and offering a set of standardised communication verbs and operations.
- *Controller Signals Sender* – All communication between the central processing unit in the cloud and all clients are established and performed via this sender unit. For example, reporting error codes to the client to resend a portion of missing data.

3.4.2. Decision-Making Module (DMM): As will be explained in table 5.1 in section 5.2, among the six strategies and policies described in section 5.2 (Home Appliances Analysis and Energy Saving Strategies), the framework follows three main strategies to reduce energy consumption, these are: 1) Measure and predict the consumed energy of appliances from any category, to decide to substitute it with a more efficient one. 2) Measure and predict the percentage usage of appliances from any category to check the possibility to use smaller ones. 3) Measure and predict the usage habits and intervals of schedulable appliances, belong to the previously mentioned third category, to apply automatic running schedules. Based on several criteria this module decides which strategy should be applied and which action is recommended.

3.4.3. Data Analytics Engine: The output of this engine will be used in many modules such as the aforementioned *Visualisation and Reporting component*. This component is one of the most resources consuming parts of the whole system because it deals with a massive amount of data. It consists of three main modules, as follows:

- *Context-Sensitive Analysis* – As the name suggests it offers the possibility to evaluate and analyse the flowing data while considering its contexts and the surrounding environmental parameters and background to deliver the most relevant, suitable, and accurate analysis. Together with the data, several anonymous data are sent for instance to define the property type where the system is implemented, or to describe the types and quantity of appliances. Important to mention that context-sensitivity adds demands additional processing power and resources, therefore it is considered an expensive approach, however, it adds a valuable added value by delivering more accurate and relevant outcomes.
- *Predictive Analysis* – One of the most significant modules in the framework. It is responsible for making predictions of future behaviour. It uses different techniques such as data mining, machine learning and big data analytics. Figure 4.5 illustrates this approach. Both experiments in chapter 6 are illustrating the implementation of this module.

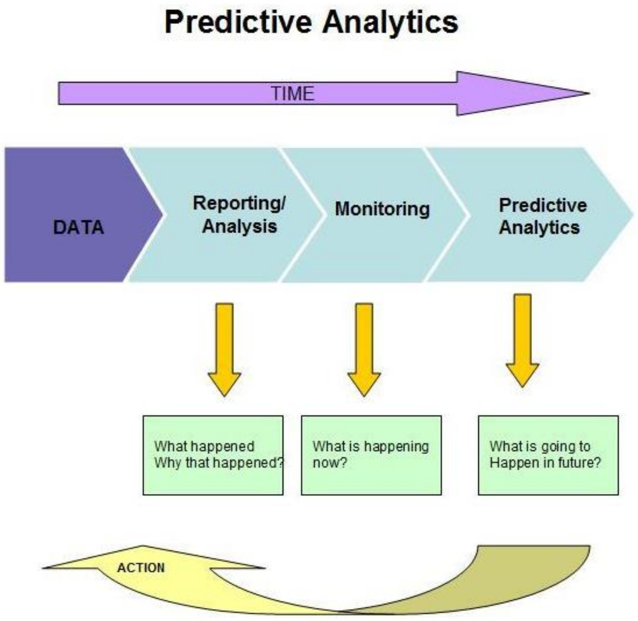


Figure 4.5: Predictive Analytics Lifecycle

- *Detection Probability Generation* – This is one of the most important modules of the framework, basically it does calculate the probability of an event based on the collected data. The processed data consists of the data collected from different sensors, external data providers and predicted data.
- *Alerting Mechanism* – Alerts will be issued when certain circumstances are detected. In the I3SEM framework, there will be predefined rules-sets, these will be applied to the proceeded data, when a situation that matches the rules-sets, is detected an alert will be issued, for instance: sending an alert when an appliance exceeds the expected energy consumption levels.

4. **Database**: In the proposed framework, a choice has been made to operate the system using a NoSQL-Database, not an SQL-Database to support the scalability and the cloud computing, because NoSQL database is designed to store data without structure (as objects), and can be spread across cloud platforms. For example, a NoSQL database such as MongoDB has built-in features such as replication and sharing.

4.4.4. Main Workflows

The framework is based on data being collected, processed and analysed, then used for visualisation and predictions. In this section, some of the leading dataflows will be described.

Sensors' Data Workflow – The data's flows start when sensors produce measurements data and send it to the local controller. Data gets processed first of all locally by validating and checking them; only valid and processed data goes to the central system in the cloud. Over the Gateway the RESTful API is the first contact point in the cloud, data and sender get checked and validated, only data from authenticated senders with enough permission levels are accepted and forwarded for further processing. Processing's output is tightly coupled with the visualisation module. Processing may require having some external data from external services such as weather forecasts, or traffic situations, which is retrieved over the Cloud Gate. This workflow is illustrated in Figure 4.6

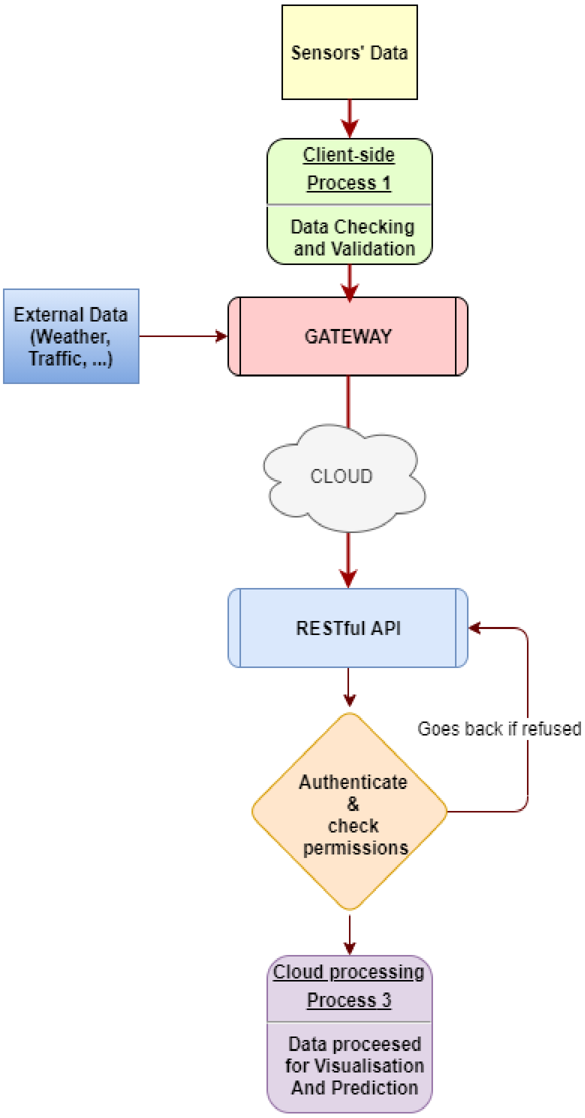

Sensors' Data
Client-side Process 1
Data Checking and Validation
External Data (Weather, Traffic, ...)
GATEWAY
CLOUD
RESTful API
Goes back if refused
Authenticate & check permissions
Cloud processing Process 3
Data proceesed for Visualisation And Prediction

Figure 4.6: Sensors' data collection, validation and processing workflow

Rules-Sets Dispatching Workflow – As described in section *4.4.2 I3SEM Framework's Components* the client-side checking and validation process requires rules. These rules are dynamic and may change according to needs. Due to the fact that

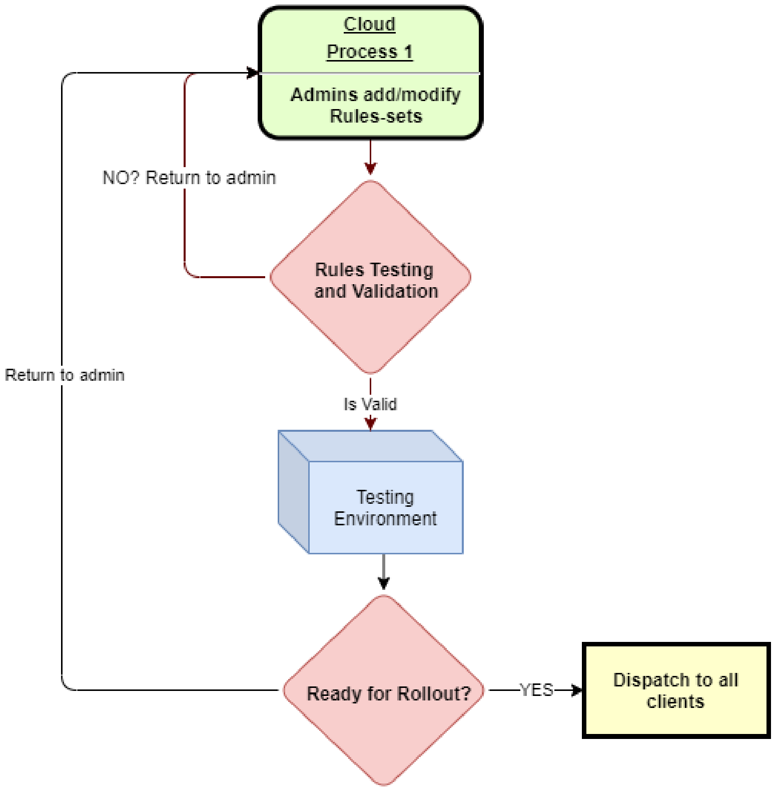


Figure 4.7: Rules-Sets Creation, Validation and Dispatching Workflow

this framework is scalable and may be used with any number of households, changing the rules manually will be a painful process. Substituting this manual process with an automatic one using the rules-sets dispatcher will solve the problem and add more flexibility, dynamicity and reduce the running costs during the system life cycle. The flow starts with an admin changing an existing rules-set or adding a new one, then publishing it, these get regularly sent to the clients after performing some tests and validation actions using simulations or testing environments. Figure 4.7 illustrates this approach.

External Data Services Data's Workflow – The internal data processing may require additional data than the one collected from different clients (households). This approach is supported by this framework. Data may be retrieved from external resources such as weather forecast stations, traffic stations or energy providers to get the price and the current load. As illustrated in Figure 4.8, this workflow is initiated by the processing unit, which invokes a call to one of the corresponding microservices over the RESTful API. The API performs the necessary communication with the external data services and return the results to the processing unit after applying some validation actions. The combined data (household's data and external data) will be used for visualisation or prediction purposes.

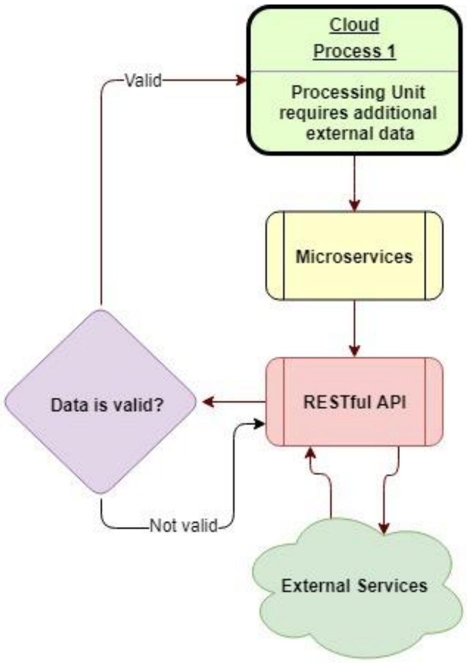


Figure 4.8: External Data Services Workflow

4.5. Conclusion

The proposed I3SEM framework is an attempt to offer responses to overcome a number of shortages, drawbacks that are considered challenges in this field and were not fully addressed in the reviewed frameworks, which can be summarized in firstly, the lack of integrated architectures. Secondly, the lack of standards and unified data structures. Thirdly, the restriction of the applicability on legacy and modern, smart environments. Fourthly, the lack of overall security concept – including IoT security, and protection against potential bundled points attacks. Fifthly, the lack of mobility management, and finally the lack of stakeholders' involvement. These issues were addressed in the proposed framework by providing a number of components and modules matching several functional and non-functional requirements.

Quality factors such as scalability, performance, data integrability, interoperability, and user-friendliness are the main non-functional requirements where the framework is built around. These factors were addressed during the design phase by introducing corresponding components. Moreover, the proposed framework is designed to cover many functional requirements, such as gathering, validating, processing the energy consumption figures periodically and predicting future usage as a result of analytical processing. The output of the processed data will be used for visualisation and stats purposes as graphs and charts, also will be the input for some prediction calculations processes. The framework also offers web-based communication channels which follow state-of-art approaches to enable efficient and performant communication within the system's units. Finally, visualisation interfaces are introduced to ensure proper informative medium, interaction possibilities and efficient contribution of stakeholders.

As previously mentioned in the literature review chapter number two, the reviewed frameworks suffered from some drawbacks related to the lack of integrated architectures, lack of standards and unified data structures, restriction of the applicability on legacy and modern and smart environments, lack of overall security concept, lack of mobility management, and lack of appeal to stakeholders. Whereas the proposed framework attempts to deliver clear answers to those drawbacks by incorporating standards for exchanging, analysing and displaying energy data, and measuring the performance. Also, by supporting decision-taking mechanisms and organisation services to consider the amount of energy consumed by various assets, or by different processes, to enable energy optimization on both local and global levels. And meeting the requirements of compatibility, expandability and interoperability to support further future developments and extensions. Finally, offering a platform to run all together.

I3SEM is divided into three main zones: client zone, cloud gate and cloud processing zone. Each zone consists of a number of components and modules and has its responsibilities inside the framework. The client zone is implemented inside the household or the organisation where occupants live, and its main responsibility is gathering, validating, and processing data. Outputs are sent to the cloud processing zone via the cloud gateway. Cloud gateway contains a number of modules to ensure a high level of security, privacy and enhanced performance; these are session management module, authentication & permissions module, load balancing, system security and mobile management. The cloud processing zone with its corresponding components and modules is responsible for processing, analysing, prediction, visualisation, and decision-making activities. The communication and networking inside the proposed framework are designed to support a high level of integrability, security and scalability. This is also emphasized by integrating several state-of-art technologies and paradigms such as microservices, predictive analytics, decision making and cloud computing.

5. Implementation & Evaluation

5.1. Introduction

This chapter provides a detailed description of the implementation of the proposed I3SEM integrated framework in a selected household environment which consists of a combination of conventional and smart equipment. Beginning with analysing the household's appliances by categorising them into three main groups according to their operational behaviour: uninterruptable appliances, instance or run-on-demand appliances, and schedulable appliances, which enable deciding to apply the most appropriate strategy. Algorithms and a data subset are evaluated in this chapter to provide a proof-of-concept for the fully implemented system, which is presented in the next chapter using the entire dataset. The data-subset, algorithms and equipment are used to evaluate the implementation of selected energy-saving strategies which are mainly based on firstly, utilising appropriate appliances' sizes that match the household's occupants' needs, secondly systematically running appliances Just-on-Demand, and finally, offering the possibilities to substitute inefficient appliances with more energy-sufficient alternatives.

In this scope, data will be gathered from various resources including the sensing devices installed within the household environment, external APIs, administrational interfaces, and data mining techniques. Part of the sensing devices will be bought off-the-shelf, however, the majority of them will be designed, built, or tailored especially for this case study as will be introduced in section 5.3 which includes: Energy Consumption Recorder, Unique Occupant Detector, Refrigerator Fullness Detector, Refrigerator Settings Panel Reader, and the Immersion Heater Inspector. The external APIs are providing the system with data related to the weather forecast, traffic and general data such as national holidays, etc. System administrators are responsible for feeding the system with additional data related to the appliances' attributes, household occupants' data including working hours, work addresses, etc. Predictive analysis is essential in the implementation phase to construct a robust dataset together with measured data, covering both historical and future predicted data to allow applying various strategies and techniques to reduce energy consumption. This will be accomplished by following the Cross Industry Standard Process for Data Mining (CRISP-DM) methodology that suggests following a number of steps explained in this chapter including understanding the business, then understanding and preparing the data, followed by applying machine learning algorithms to obtain the most suitable models by passing them through a number of cross-validation methods. Appropriate data mining algorithms will be applied for prediction purposes to deliver the most accurate picture for the near future, which adds more reliability and sustainability attributes to the system.

A number of applications, paradigms, components will be explained in this chapter in detail to accomplish the case study. This includes the Java-based Reduction of Energy Consumption in Household Sector (RECHS) dedicated to being the central interface among stakeholders, administrators, various system components, APIs. Also explains the developed microservices farm and its role to support the scalability and performance which is illustrated by simulating simultaneous requests sent by 5000 users in parallel using a load testing tool called *JMeter*. The data integrability attribute of the framework will be illustrated by introducing a number of data transformers that can handle all data collected from different sources having different formats varying from images to binaries, or even ASCII with different data structures. The mobility management approach and its rules, which is one of the security approaches introduced in the framework will be explained in section 5.6.3.

5.2. Home Appliances Analysis and Energy Saving Strategies

Almost every household in western countries has a number of essential household appliances to maintain daily life duties such as cleaning, cooking, food preservation, heating and entertainment. These devices can be grouped into different categories based on various criteria, such as function and purpose, size, energy consumption, operational energy source, operational mode (digital, analogue). For the purpose of this research, since it is not possible to dedicate a special experiment for each home appliance within the household, a new classification criterion based on the running or the operational behaviour, is used to divide home appliances into three main groups, where at least two experiments will be designed for two sample appliances from the following groups:

1) **Uninterruptable Appliances (UA)** – which includes all appliances that should not be interrupted by switching it on/off during running periods, they have their own built-in management module, because any exterior interfering with the running mechanism may lead to disturbing the main function of the

device. A clear example is the refrigerator, where running periods are controlled internally depending on the internal temperature.

2) **Instant or Run-on-Demand Appliances (IA/RODA) –** where a device is turned on/off explicitly based on the household wish and upon their needs. Appliances keep their status (on or off) as long as it gets changed by the user. Most appliances in the household belong to this category, such as TV, washing machine and clothes tumble dryer.
3) **Schedulable Appliances (SA)** – these appliances can be switched on/off upon need additionally they can be pre-programmed to run in certain periods, heating systems, air-conditioning, or under-sink immersion heaters are obvious examples of such appliances.

Optimizing energy consumption in a household can be accomplished by applying a number of strategies: firstly, replacing old and high-energy-consuming appliances with new ones, based on particular calculations. Secondly, detect the habits of energy consumptions and adjust appliances accordingly. Thirdly, observe the habits of energy consumption and send this data to the local Grid companies to adjust and optimize their energy generation and supply. Moreover, the energy consumption data from specific appliances together with other relevant data recorded from the operational environment such as the number of occupants, internal temperatures, times the door is opened and for how long, aggregated from different households, offer real-life energy consumption measurements, so it assists the manufacturers to identify the weak points in their appliances and allow them to produce more efficient appliances. This data could be also interesting for relevant government agencies to issue proper and real-life regulations. Fourthly, detect the phases when pre-selected appliances are running on sleep-mode or standby-mode, and shut it completely off, for example: Under sink water heaters (with tank), TV and audio equipment. Fifthly, detecting and calculating the probability of forgetting switching off pre-selected appliances. People may forget to switch off appliances before going to sleep or before going on holiday, so based on several signals (Movement detection, regular habits, weather, etc) the system may decide to switch off or switch on an appliance. Sixthly, alerting users upon misbehaviours, where user gets notifications, either as mobile notifications, or recorded voice alerts, when pre-defined rules take place. For example, when heating or AC runs in a room, and when the window is left open, and when the temperature drops under predefined level, and the weather forecast is cold.

However, in order to achieve the main goal of reducing energy consumption, the following three main strategies and policies were chosen for this research, each of them is suitable for one or more groups of household appliances as shown in table 5.1,

Appliance/Categories **Strategies/Policies**	Uninterruptible Appliances (Example: Refrigerator)	Schedulable Appliances (Example: Immersion Heater)	Instant or Run-On-Demand Appliances (Example: Tumble Dryer)
Energy Consumption based Appliance Substitution Policy (ECASP)	X	X	X
Usage Percentage based Appliance Substitution Policy (UPASP)	X	X	X
Automatic Scheduling of Running Periods Policy (ASRPP)	-	X	-

Table 5.1: Various household appliance categories and the correspondent strategies/policies that might be applied.

these are:

1) **Energy Consumption-based Appliance Substitution Policy (ECASP)** – according to this strategy the total dataset will be constructed by combining historical and predicted wattage data. The historical data is collected from the observation of energy consumption of a certain device, where the future consumption data is predicted by applying various prediction techniques, as will be explained in the next chapter. Combining both sources will offer a more accurate basis to take the proper decision to substitute the device with a more efficient one. The decision of which device to choose depends on the amount of the expected energy saving percentage, which is calculated using the following two equations:

$$\boldsymbol{E_1} = \boldsymbol{MPE_{(kWh)}} * \boldsymbol{T_{(hr)}}$$

Where:

$\boldsymbol{E_1}$: *Current Appliance's total daily energy consumption in kWh,*

$\boldsymbol{MPE}$: *Measured and Predicted Energy Consumption in kWh,*

$\boldsymbol{T}$: *Amount of Hours Refrigerator Runs Daily*

$$\boldsymbol{E_2} = \boldsymbol{NAC_{(kWh)}} * \boldsymbol{T_{(hr)}}$$

Where:

$\boldsymbol{E_2}$: $New\ ppliance's\ total\ daily\ energy\ consumption\ in\ kWh,$

$\boldsymbol{NAE}$: $New\ appliance's\ total\ daily\ energy\ consumption\ in\ kWh,$

$\boldsymbol{T}$: $Amount\ of\ Hours\ Refrigerator\ Runs\ Daily$

$$\boldsymbol{Annual\ Energy\ Saving} = \frac{\left(\boldsymbol{E_1} * 365_{(day)}\right) - \left(\boldsymbol{E_2} * 365_{(day)}\right)}{\boldsymbol{E_1} * 365_{(day)}} * 100\%$$

2) **Usage Percentage based Appliance Substitution Policy (UPASP)**– Similar to the previous strategy, with one difference related to the motivation behind substituting the device. In the previous strategy, the decision is taken based on the consumed energy, however, in this approach, the decision is taken based on the percentage usage of the device. For example, when a single person uses a device designed for bigger families, the system should make a recommendation to substitute this big device with a smaller one, which automatically leads to reduce the energy consumption.

3) **Automatic Scheduling of Running Periods Policy (ASRPP)** – The appliance in question will be observed for a certain period, and the dataset resulting from this observation together with the data collected from the relevant surrounding parameters (such as room temperature) will be analysed and used to predict the periods when a device must run, the system will then automatically decide whether to switch it on/off accordingly, for example, the immersion heater will be switched off during the night. Table 5.1 shows the relation between the household appliances group and the correspondent applicable strategy.

5.3. Experimental Settings

The planned experiments aim to evaluate the proposed I3SEM Framework by utilising a data subset collected from the installed equipment and different appliances in the household where the implementation takes place. Basically, the household will be equipped with five different self-developed systems and appliances, as follows:

1) **Z-Wave-based Energy Consumption Recorder (ZW-ECR)** – As shown in Figure 5.1, this recorder consists of three main components: Z-Wave-based, AES-

128 bit-encrypted, 868.42MHz Z-STICK GEN5 controller [181], Z-wave-based, 868.42MHz, Smart Switch 6 [182], that delivers a real-time appliance electricity consumption with 99% accuracy, and uses 3 different security layers including Z-Wave S2. Both are produced by AEON LABS. And the local server hub was built using Java Spring Boot, with RESTful API support. All operate within a local Wi-Fi environment. The main function of this structure is measuring and recording the voltage (V), Current (Amps), Usage per hour (kWh) of the connected household appliance, over a certain period of time. A timestamped data are saved to the database in 3-second intervals.

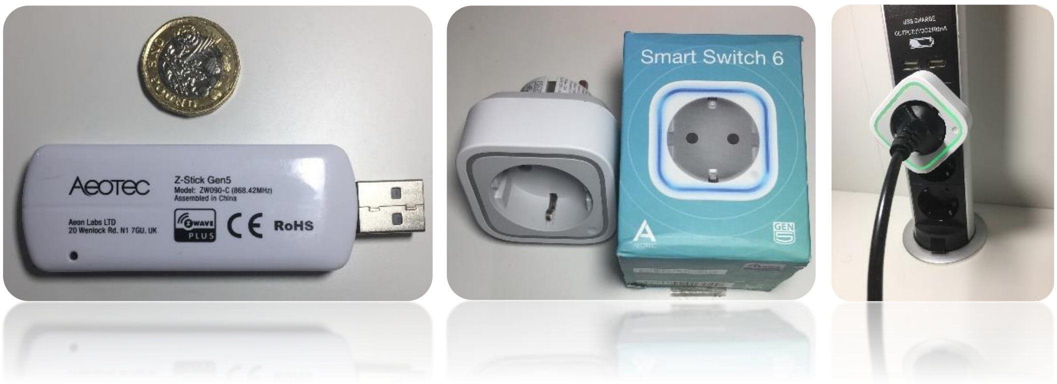


Figure 5.1: ZW-ECR Module mainly consists of a Z-Stick GEN5 controller, smart switch 6 from AEON-LABS

2) **Arduino Uno-based Unique Occupant Detector (AU-UOD)** – The settings are a combination of several off-the-shelf and self-designed-and-assembled hardware and software components put together to detect the number of unique household occupants living in the household. Visitors who stay longer than a predefined and adjustable period of time; such as grandparents, or babysitters etc., are also counted. The hardware consists of an open-source microcontroller board called Arduino Uno, which is based on the Microchip ATmega328P microcontroller, developed by Arduino.cc [183], a combination of a camera and Wi-Fi/Bluetooth module called ESP32-CAM is attached to the board to provide the images and send them to the server hub regularly. The component has 802.11b/g/n Wi-Fi BT SoC module with support for STA/AP/STA+AP operation mode, equipped with Up to 160MHz clock speed, and Built-in 520 KB SRAM, and external 4MPSRAM [184]. It consists of three main parts: Wi-Fi/Bluetooth module, an embedded IP camera with streaming capabilities over Wi-Fi and Bluetooth in two

different resolutions; high and low, and a built-in PCB antenna. The ESP32 offers support for both cameras OV2640 and OV7670, however, the used camera is an OV2640, because it has better resolution. Additionally, a Zigbee 3.0-based magnetic door open/close detector with an actuation distance of 15-25 mm, and Voltage of 100V and 0.5A current, is implemented to trigger the cam upon opening/closing the main house door to capture the occupants when entering or leaving the household. The software part consists of a program, called Sketch [185], written in Arduino Programming Language, which is based on C/C++ and compiled using AVR GCC. A sketch is uploaded to the board via an open-source Arduino IDE 1.8.13 [186]. The server hub application is developed in Python 3.9.0 supported by many libraries such as OpenCV 4.4.0.46, pip 20.2.4, Keras 2.3.1, tensorFlow 2.4.0, sklearn.

3) **Arduino Uno-based Refrigerator Fullness Detector (AU-RFD)** – With almost identical settings to the previous AU-UOD, this system consists of an Arduino Uno as a board, an ESP32-CAM with Wi-Fi and Bluetooth capabilities, all put inside a self-made styrofoam thermal box with appropriate holes for the camera and the instant flash (Figure 5.2). The hardware is programmed using a sketch which is uploaded to the board via the open-source Arduino IDE 1.8.13. Images will be taken from the upper refrigerator shelf from where the camera is located, every 10 seconds, then it is transmitted via the built-in web server running under the URL http://192.168.2.115/ to the hub application developed in Python. This application receives the images, process them and determine the approximate percentage of the occupied area to ascertain the percentage of the fullness of the refrigerator. Data are saved accordingly to MySQL. The main application runs with the support of many libraries such as OpenCV 4.4.0.46 and NumPy 1.18.5.

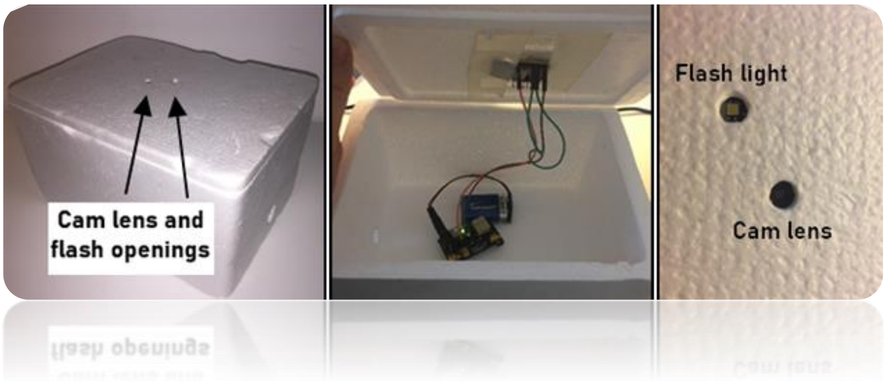
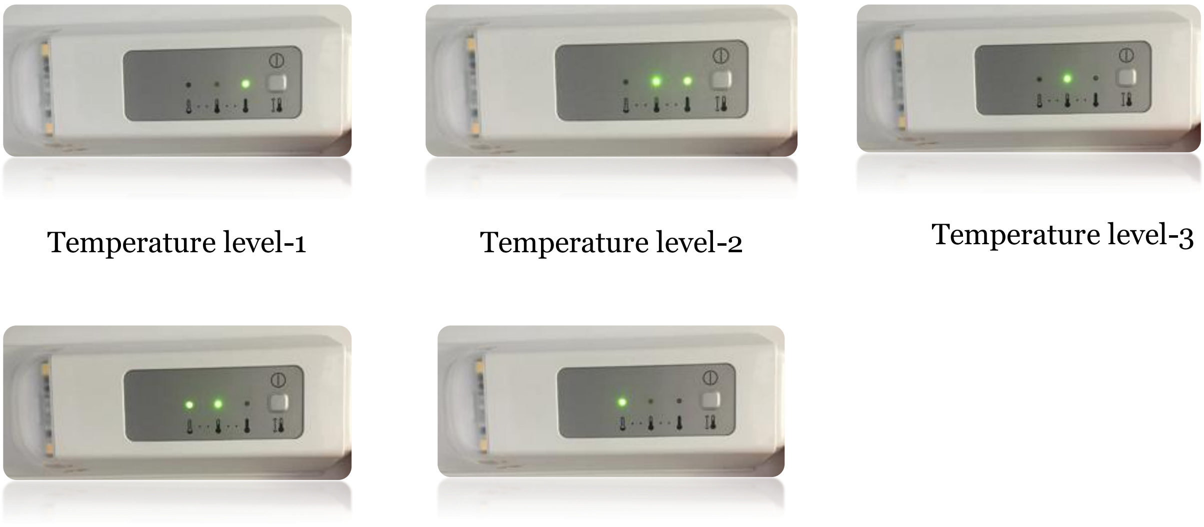


Figure 5.2: The self-made Arduino-Uno based refrigerator fullness detector consists of a cam, flash and Wi-Fi-enabled ESP32-Cam

4) **Arduino Uno-based Refrigerator Settings Panel Reader (AU-RSPR)** – The same hardware which is used in the fullness detector is used in this module, however, both positioning of the box and the image processing is different. Figure 5.3 shows the five different temperature levels of the refrigerator temperature settings panel, which is converted into digital numbers using this module.



Temperature level-1

Temperature level-2

Temperature level-3

Temperature level-4

Temperature level-5

Figure 5.3: Refrigerator Manual Temperature Settings Panel (5 levels)

5) **Arduino Uno-based Immersion Heater Inspector (AU-IHI)** – It is designed to measure the amount of water, the consumed energy and the periods when hot water flows from the hot water cylinder with an immersion heater (located under the sink). The previously mentioned ZW-ECR module is used to measure the consumed energy. Measuring the consumed hot water is accomplished using the open-source microcontroller board called Arduino Uno together with a water flow sensor, model YF-S201, with a working range between 1-30L/minute and a capability to function under a water pressure of ≤ 1.75MPa, as the lowest-rated working voltage runs between DC4.5 5V-24V, which requires 15mA (DC 5V) to operate with a load capacity of ≤10mA (DC 5V). The flow range must vary between 1-30 l/min. Similar to the previous software setting the hardware programming is accomplished using Arduino IDE, and the server hub application is achieved in Python. The assembled module can be seen in Figure 5.4

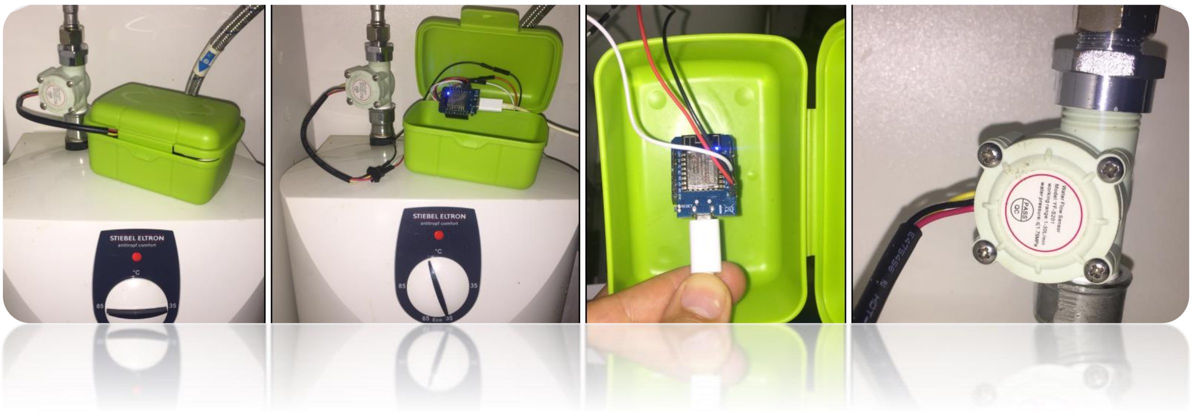


Figure 5.4: The self-made Arduino-Uno based immersion heater inspector consists of a YF-S201 water flow sensor and Wi-Fi- and Bluetooth-Modul NodeMCU ESP32

As indicated in section 2.5.2, the decision was made to use the InfluxDB NoSQL database, however, due to the tiny feature differences between InfluxDB and MongoDB, and the available know-how of MongoDB, the decision is made to use MongoDB in this implementation phase.

All previously self-developed systems are served by a Microsoft Windows 10 Home, version 10.0.19042 Build 19042 running on an x64-based Intel(R) Core (TM) i5-7200U CPU @ 2.50GHz, 2712 MHz, 2 cores, 4 logical processors. The hardware manufactured by HP Pavilion Laptop 15-cc007ng, with a BIOS-version Insyde F.13 from 03.11.2017, and the hardware abstract level version: 10.0.19041.488. The programming was mainly carried out using JAVA Spring Framework 5.0 on JDK 8 with the Java EE platform running under the Apache License 2.0. Python 3.7 with a number of compiled libraries such as TensorFlow 2.5.0, OpenCV 4.1.2.30 was used mainly to accomplish all artificial intelligence and predictions tasks.

As explained in chapter 4, the proposed I3SEM introduces three zones divided physically into two main areas, one takes place in the household or community where sensors, appliances and occupants exists, and one in the cloud where systems controls, visualisation components and core processing unit exists. The implementation and the whole setup reflect this structure by preparing all components, units and modules in the way to be deployed and run separately. Important to mention that applying certain software paradigms and technologies such as microservices and Java have enabled this approach, because both offer a huge number of relevant built-ins and features. Although everything was prepared to implement the system locally and, on the cloud, the cloud part was not deployed to the cloud, rather to a simulated cloud that runs locally, due to costs and hardware limitations offered in the free hosting version offered by a number of cloud hosting companies such as Heroku, AWS. The applied alternative, by running the whole system locally, offers more stability and control on the whole application, so no need to plan any backup or crash scenarios in case of downtime, which makes the implementation efforts shorter and easier. Moreover, cloud computing was invented in the framework to support dealing with endless number of households, however in the implementation only one household was considered, so using the cloud for this particular case may considered sort of overfitting.

5.4. Predictive Analysis

Obtaining the energy consumption wattage and the operational usage data from the sample devices; the refrigerator and immersion heater in both experiments by detecting the previous historical along with the future predicted data, establishes a robust background to apply various strategies and techniques to reduce the energy consumption for this appliance. Important to mention that this approach can be applied to other different appliances within the same household with high energy consumption which belong to the same category, such as heating systems, air-conditioners, tumble clothes drying machines, and so on. Or even aggregating this approach by applying it to a large number of households for this sample device, and or other devices. Data can also be used by different establishments such as local governments and energy suppliers to adjust their energy production accordingly. Moreover, data could be interesting for the appliances' manufacturers who require different operational parameters collected from their devices while running in a real environment, so they can spot the weak points and deliver improved versions.

The implementation phase went through a number of sequences to define and prepare the data to produce the desired prediction. The whole data mining process is managed by the Cross Industry Standard Process for Data Mining (CRISP-DM) methodology [187] which divides the data mining process into six main steps: beginning with business understanding, then data understanding, followed by data preparation, modelling, and evaluating the resulted models, and finally deploying the chosen models, all phases are illustrated in Figure 5.5.

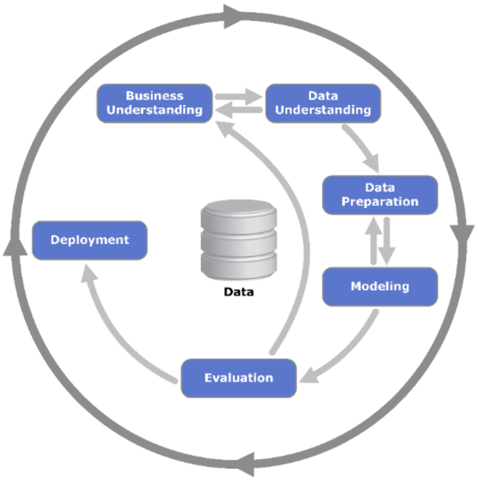


Figure 5.5: Cross Industry Standard Process for Data Mining (CRISP-DM) Methodology‘s Six Phases and the flow [187]

As the CRISP-DM methodology suggests the process is not a one-way direction process, rather there is a need to move forward and backwards among various phases to make necessary adjustments till reaching the desired result. Using the subset data and this methodology prediction models are created to predict both energy consumption of the refrigerator and the energy consumption and flowing water rates (usage or running periods) of the water immersion heater. The combination of historical data and the predicted data are put together to establish a solid database which will be used to suggest various approaches to save energy which will be explained including results and saving rates in chapter 6. During the implementation phase, the relevant CRISP-DM methodology phases have been considered and applied in the next sections.

5.4.1. Business Understanding

The household where the implementation of the proposed I3SEM framework takes place contains a number of conventional and smart equipment. As mentioned in section 5.2 household appliances are divided into three main categories: instant or run-on-demand appliances such as TV or light, uninterruptible appliances which should not be switched on/off externally and have an internal module that decides when to switch it on/off based on internal parameters. The refrigerator is one of the best examples, and the last category, the schedulable appliances that can be switched on/off upon need, such as under-sink water immersion heaters, heating-system, air-conditioning. In this research, the main focus will be on applying the proposed I3SEM framework on two sample appliances from both categories: the uninterruptible appliances category and schedulable appliances. Data were gathered from different external APIs, sensors and cams. Part of the sensing devices was bought off-the-shelf, moreover, the majority of the sensing and control equipment were completely designed and assembled especially for the purpose of this research. Datasets are prepared as training and test data for the use of several prediction analytics to obtain models, which are compared and evaluated based on several related metrics.

The Cross-Industry Standard Process for Data Mining (CRISP-DM) methodology is followed to predict energy consumption and appliance's usage percentage. According to the followed CRISP-DM methodology, all begins with understanding the business and determining its objectives, also analysing the business area and its related circumstances. Generally speaking, the framework is built around the idea of reducing energy consumption in the household area by applying different strategies and policies, such as delivering recommendations to substitute appliances with more efficient ones based on the measured and predicted energy consumption or the usage percentage or automatically adjust the running periods of certain appliances based on predicted parameters. However, the dynamicity, integrability and scalability nature of the framework opens avenues for an indirect impact on various application fields such as offering energy consumption feedback for energy suppliers when energy consumption tracking and prediction techniques are aggregated and applied on every single appliance in the household and other households in the neighbourhood areas.

Applying data mining techniques and predicting data of a particular appliance is essential in this research because: Firstly, predicted data combined with the measured historical data, enhance the overall dataset size to supply a solid and reliable basis to take accurate decisions. For example, if a decision is made to replace an appliance, this decision is considered reliable because it is anchored on a total of nines months of data, in case the energy consumption data is measured for three months and predicted for the next six months. Secondly, there is no need to install the system in every household to measure and record data. By applying the data mining techniques, a model can be built and used to predict data on other households with comparable conditions.

5.4.2. Data Understanding

In this phase of CRISP-DM modelling, data will be taken into the focus, including deciding for the most relevant data variables, then gathering, describing, exploring, and verifying actions to have a robust dataset basis for the next steps. Important to mention that this step has been revisited and adjusted several times during the overall data mining process to ensure having reliable evaluation results.

5.4.2.1. Data's Associated Technical Limitations

As mentioned before in section 2.5. the enormous amount and variety of data retrieved from various resources bring several challenges. The picture was not different while dealing with the data during the implementation phase, where several data limitations were faced. Following is a list of the most important causes of the data limitations were faced during the gathering stage:

1) **Hardware Malfunctioning** – A huge portion of used data were retrieved from appliances and sensors installed within the household environment. Due to technical difficulties, hardware quality issues and unexplainable interruptions, it is noticed that some data were corrupted and suffer from inconstant and inconsistent values that were way beyond the expected ranges. A possible reason for such unexplainable hardware behaviour is the missing the overall integrability checks among the different components manufactured by different vendors, probably some of the combinations of the components were never tested together for this particular purpose. An example is running a Zigbee-based door motion sensor with an Arduino-based cam located inside a cold environment (below 4◦ C) in the refrigerator, synchronized within a Wi-Fi LAN together with many other sensors.

2) **Data Redundancy** – Also data redundancy was noticed in the database, where exact similar readings with the same decimal level were saved several times with the same timestamp. This occurred due to a high load on the server or in the local area network, and the shortage of the proper handling of message queuing on the database level. This issue caused a delay to the whole process because of wasting storage capacities, increasing the processing time due to the increase of the processed data, and consuming longer development time invented to eliminate the repeated data. Extending the LAN and server capabilities would minimize or even remove this shortage, however, this solution is not considered due to costs

restrictions and the need to perform the implementation in an average household environment.

3) **Lack of Native Smart Household Appliances** – The majority of used household appliances are legacy appliances without any smart capabilities. The household environment where the implementation takes place, disposes only of conventional appliances, so because of financial restrictions it is not possible to replace these appliances with smart ones, therefore these appliances were extended with additional smart sensors and components. As mentioned before (section 5.3.1.1, the combination of those devices and the sensors are not always functioning smoothly. The best alternative would be using appliances with natively integrated and tested interfaces to gather and communicate the data.

4) **Lack of standards in this field** – The lack of standards in this relatively evolving field, brings several challenges. This can be seen in the different sensors and components using various protocols such as Zigbee, Z-Wave, Wi-Fi, X10, Insteon, Thread, Bluetooth Low Energy (BLE) and Universal Powerline Bus (UPB). As will be seen in chapter 6, three protocols were used in the implementation, this brought an extra effort to use additional integration and translator plugins to let the hardware based on these protocols perform homogenously within the same environment and unify the resulted data structure retrieved from all different components.

5) **Artificial Intelligence Challenges** – As mentioned before, one of the challenging tasks within the implementation phase was converting the normal and conventional appliances to smart ones by adding smart capabilities. This was achieved by implementing some artificial intelligence approaches such as facial detection to identify the number of occupants inside the household, and parsing images taken from the refrigerator control panel to retrieve the adjusted temperature. Reaching high accuracy levels require massive computing power, and specialised, high-definition equipment such as cameras. In this research, despite the lack of those resources a relatively low however acceptable accuracy levels have been achieved.

Apart from the previously mentioned challenges, the next section provides a detailed description of the data used for both experiments and their sources.

5.4.2.2. Data Sources and Structures

Since data's quality and quantity plays an essential role in the data mining process, it is extremely necessary to decide on relevant data variables by defining selecting criteria, which are collected from available and trustworthy data sources while assembling the required dataset. This section will provide a detailed description of the used data sources, and the data's initial structure, with samples data illustrated in tables and graphs. Basically, both experiments share the same data sources, these are either internal data sources such as sensors, cameras, and processed data, or external APIs, as follows:

External APIs – A number of external APIs were used to collect the relevant data, such as OpenWeatherMap API which delivers data as represented in Table 5.2 and Figure 5.6.

Measured in month	Average temperature (2m overground)	Average temperature (5cm overground)	Average humidity (2m overground)	Weather condition	Records used for average e calculation
January	-1	-2	84	Stormy	13392
February	0,5	-0,8	78	Snowy	12096
March	4,7	4,1	73,9	Rainy	13392
April	8,6	8,3	69,8	Rainy	13518
May	11,7	12,1	76,1	Rainy	17856
June	16,8	17,5	73,7	Rainy	17280
July	16,3	16,7	69,5	Rainy	17856
August	18	18,1	74,3	Sunny	17856
September	11,1	10,5	75,8	Rainy	17280
October	7,9	8,9	80,4	Stormy	17856
November	3,8	3,1	85,4	Rainy	17280
December	1,7	0,7	79,5	Snowy	17856

Table 5.2: Example weather data retrieved from the OpenWeatherMap API for the household located in Recklinghausen – Germany

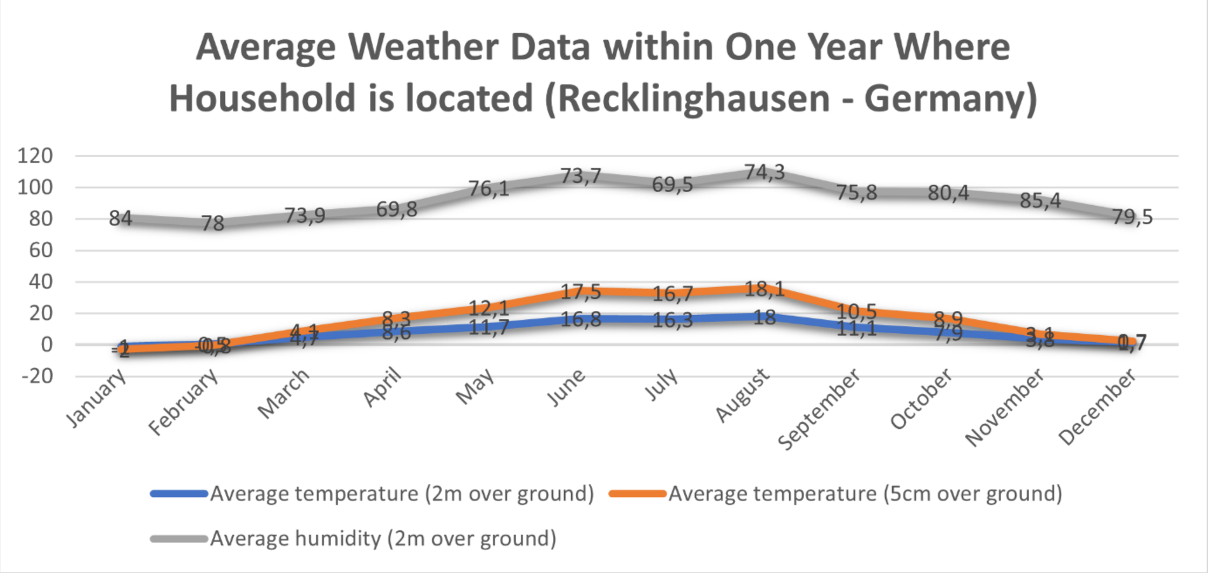


Figure 5.6: Average weather data measured within one year where the household is located in Recklinghausen - Germany

Internal Data Sensors – As previously explained in section 5.3 (Experimental Settings), a number of internal sensor systems were assembled to measure and collect different parameters. As the framework suggests, there are several sensors were implemented in different appliances, these are Energy consumption sensors, internal and external temperature sensors, internal and external humidity sensors, refrigerator door open/close detector, cameras. Table 5.3 shows samples of the collected raw data.

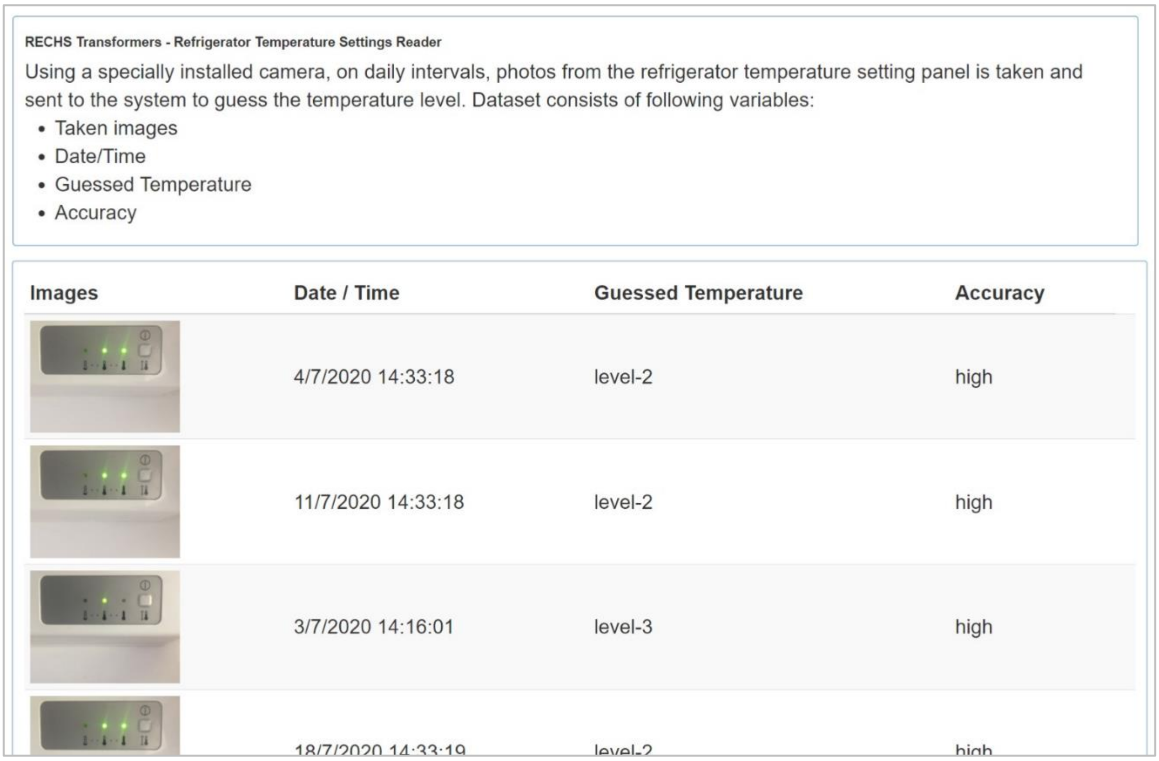
Average Internal temperatur e	Average External temperatur e	kWh	Times Frig door opened per 10min	Total seconds door left opened	Day of the month	No. of records used for the average calculation
21,7	18,8	71	1	17	01	48
19,2	16,8	62	1	9	02	48
19,2	17,2	61	1	9	03	48
21,7	19,1	80	1	17	04	48
23,2	18,6	96	2	23	05	48
21,7	18	75	1	17	06	48
23,2	22,2	97	2	23	07	59
51	14,2	92	2	23	08	72
23,2	13,2	92	2	23	09	72
21,7	14	84	1	17	10	71
18,9	12,2	40	0	4	11	72
21,7	9	83	1	17	12	71
23,2	14,1	99	2	23	13	70
21,7	15,5	87	1	17	14	68

86	16,6	93	2	23	15	72
23,2	15	89	2	23	16	71
21,7	12,5	78	1	17	17	72
21,7	15,4	74	1	17	18	70
23,2	15,2	104	2	23	19	71
23,2	15,6	92	2	23	20	69
23,2	14,8	89	2	23	21	71
21,7	14,6	83	1	17	22	70
21,7	14,6	69	1	17	23	72
19,2	14,4	61	1	9	24	72
21,7	16,1	73	1	17	25	58
21,7	16,4	87	1	17	26	48
23,2	16	90	2	23	27	48
21,7	17,9	75	1	17	28	48
21,7	17,4	80	1	17	29	48
21,7	16,6	70	1	17	30	48
21,7	18,8	78	1	17	31	48

Table 5.3: Example of data collected via assembled sensor system in the household

Machine Learning - Image recognition – This technology will be applied for three different types of data: facial recognition, measuring the refrigerator fullness (Refrigerator fullness detector) and reading the refrigerator temperature settings (Refrigerator temperature settings reader). The facial recognition technology was selected among the number of available biometric technologies such as fingerprint scanners and identifiers, palm print and iris recognition because it is easy to implement, return relatively accurate results, and does not involve complex hardware installations. Its main purpose is to determine the number of occupants of a household and save it in the database. This begins by taking pictures from different corners/rooms inside the household. Images may be taken *regularly* and *on-demand* when the main entrance opens or closes. Images get analysed by detecting the human faces, then capturing those faces by converting the *analogue* faces into digital equivalent information as a file, to initiate the matching process to verify whether faces belong to the same person or different persons. Important to mention that future work may consider using other accurate biometric technologies such as fingerprint, palm print or iris recognition. The second type is measuring the fullness of the refrigerator. The refrigerator fullness percentage will be measured by taking regular photos from beneath every shelf from inside the refrigerator every time the door is opened. Based on the taken images, the empty area, which appears in a brighter tint than the occupied area. Subtracting the occupied area from the previously measured total area gives the estimated fullness. Calculations are done using Python OpenCV and saved in the database. The third type is quite similar where the refrigerator temperature control settings are recorded by the camera fixed against the refrigerator control panel when the door is opened. Images are parsed using the Python OpenCV library to retrieve the adjusted temperature level and gets sent to the database.

Another example of data retrieved from internally running systems can be seen in Figure 5.7, which is a module to take and process images from the refrigerator temperature settings panel to retrieve the frig set temperature. The assembled equipment is based on Arduino single-board computers together with related sensors and uses Python-based software to process and analyse images and convert them to processible integer values.



RECHS Transformers - Refrigerator Temperature Settings Reader

Using a specially installed camera, on daily intervals, photos from the refrigerator temperature setting panel is taken and sent to the system to guess the temperature level. Dataset consists of following variables:

- Taken images
- Date/Time
- Guessed Temperature
- Accuracy

Images	Date / Time	Guessed Temperature	Accuracy
	4/7/2020 14:33:18	level-2	high
	11/7/2020 14:33:18	level-2	high
	3/7/2020 14:16:01	level-3	high
	18/7/2020 14:33:19	level-2	high

Figure 5.7: Transformers: Refrigerator Temperature Settings Panel Reader

System and administrative data – This covers data retrieved automatically from the system such as date/time, day type (weekend, weekday, bank holiday), or entered manually by the system administrators which describe the parameters related to the household and used sample appliances, such as the flat's size, appliances' types, models, colours, and sizes.

As a result of combining data collected from the previously mentioned three resources, a list of initial variables was put together for both experiments, these are:

First experiment: Refrigerator – As mentioned in Table 5.1, one of the policies *Energy Consumption based Appliance Substitution Policy (ECASP)* and *Usage Percentage based Appliance Substitution Policy (UPASP)* will be applied to the sample appliance, the refrigerator, which represents the group of the uninterruptible household appliances. In this experiment, regression techniques are applied due to the fact that both target features need to be predicted; the energy consumption, and the usage percentage, are quantities (numbers) not binaries (yes/no, or true/false). Therefore, algorithms such as kNN, simple linear regression, polynomial regression, random forest and tree regression are considered and evaluated. In this experiment, the initial dataset began with 16 features and ended up with eight features, where several features were removed, merged and altered during the prediction process. Table 5.4 shows the list of the initial data structure and related explanations. The final list of features will be shown and discussed in chapter 6.

Variables	**Source**	**Field Type**	**Example values**	**Description / Source / Related notes**
Datetime	System	Datetime	2021-03-10 13:59:45	Recorded automatically as a timestamp, on the database level while gathering the data.
Internal Temperature	Sensor	Float	Expected range between +10 - +35 C°	Measured by an internal sensor and shows the internal temperature of the kitchen where the refrigerator is located.
External Temperature 5cm	API	Float	The expected range between -15 and +40C°	Retrieved from the external API. Shows the outside temperature in the region where the household is located measured 5cm over the ground
External Temperature 5m	API	Float	The expected range between -20 and +45C°	Retrieved from the external API. Shows the outside temperature in the region where the household is located measured 5meters over the ground
External Temperature Measured	Sensor	Float	The expected range between -20 and +45C°	Retrieved from a sensor located outside the household (on the balcony) and shows the outside temperature.
External Relative Humidity	API	Integer	0-100%	The humidity outside the household retrieved from the external weather API

External Relative Humidity Measured	Sensor	Integer	0-100%	The humidity outside the household measured by a sensor
Internal Relative Humidity Measured	Sensor	Integer	0-100%	The humidity inside the kitchen was measured by a sensor
Weather Condition	API	Varchar	rainy, windy, stormy, snowy, cloudy, sunny,...	Retrieved from the external weather API, and describes the type of weather condition: rainy, windy, stormy, snowy, cloudy, sunny,...
Refrigerator Fullness	AI	Integer	0-100%	Calculated via artificial intelligence techniques by analysing the regular shots taken from refrigerator shelves.
Occupants	AI	Integer		An estimated number of occupants living in the household, calculated automatically by analysing data collected by different sensors and cameras inside the household.
Refrigerator Temperature	AI	Integer	Level 1 – 6	Due to technical restrictions of the refrigerator which is not equipped with digital temperature (see Figure 5.3)
Energy Consumption	Sensor	Float	kWh	Measured by a sensor for a period of time, and predicted for the future in kWh.
Times Door Opened	Sensor	Integer		Measured by a sensor to register the number of times the refrigerator gets opened in 24-hours time
Duration Door Left Opened	Sensor	Integer	X seconds	It measures the time the door is left opened every time it gets opened. This information is important to deliver an approximate value of the wasted energy.
Day Type	System	Varchar	Weekend, weekday	Calculated based on the recorded timestamp. It is either weekend, bank holiday, school holiday or regular weekday

Table 5.4: Refrigerator's energy consumption prediction's initial variables, sources, database field type, format, short description, possible data range and some examples.

Second experiment – Water Immersion Heater – The second experiment will be applied to a sample appliance that belongs to the schedulable appliances category, using both policies: Automatic Scheduling of Running Periods Policy (ASRPP). This device, shown in Figure 5.8, has a temperature regulator that varies between 35 and 85 degrees, used to define the desired water temperature. When the appliance is turned on, it runs continuously to keep the water temperature always on the desired level, even when the device is not in use. The basic idea is to bring the amount of consumed energy by the device to the lowest level, by preventing heating up the water during periods when the device is not used. This will be achieved by tracking the running periods of the appliance, together with other different variables for a period of time, then predicting the periods when the heater will be used, so the device can be switched off when it is not needed, for instance at night, or when household's occupants are not at home.

Figure 5.8: The Immersion Water Heater used in the experiment [204]

Table 5.5 shows the list of the initial data structure and related explanation. The final list of features will be shown and discussed in chapter 6.

Variables	Source	Field Type	Example values	Description / Source / Related notes
Timestamp	System	Datetime	2021-03-10 13:59:45	Recorded automatically as a timestamp, on the database level while gathering the data.
Heater Target Temperature	AI(*)	Integer		Due to technical restrictions of the immersion heater which is not equipped with digital temperature. The temperature is calculated via artificial intelligence techniques by analysing the regular images taken from the heater's temperature regulator.
Occupants	AI(*)	Integer		An estimated number of occupants living in the household, calculated automatically by analysing data collected by different sensors and cameras inside the household.
Internal Temperature	Sensor	Float	Expected range between +10 - +35 C°	Measured by an internal sensor and shows the internal temperature of the kitchen where the immersion heater is located.
External Temperature 5 Meters over the ground	API/Sensor	Integer	The expected range between -20 and +45C°	This is relevant because it affects the incoming water temperature. Retrieved from the external API and a sensor. Shows the outside temperature in the region where the

				household is located measured 5meters over the ground
External Temperature 15 meters over the ground	API/Sensor	Integer	The expected range between -20 and +45C°	Retrieved from the external API and a sensor. Shows the outside temperature in the region where the household is located measured 15cm over the ground
Wind speed	API	Integer		This may affect the temperature of the incoming water
Incoming water temperature	Sensor	Integer	The expected range between +0 and +45C°	Retrieved from a sensor mounted on the pipeline feeding the immersion heater.
Weather condition	API	Varchar	rainy, windy, stormy, snowy, cloudy, sunny,…	Retrieved from the external weather API, and describes the type of weather condition: rainy, windy, stormy, snowy, cloudy, sunny,…
External Humidity	API	Integer	0-100%	The humidity outside the household retrieved from the external weather API
Internal Humidity	Sensor	Integer	0-100%	The humidity inside the kitchen was measured by a sensor
Off day (weekends, holidays, annual leave, …)	System	Varchar	Yes, no	Calculated by the system based on the timestamp, and also can be taken from the personal calendar.
Energy consumption	Sensor	Float	kWh	Measured by a sensor for a period of time, and predicted for the future in kWh.
Appliance On/Off	DM(**)	Binary	1/0	This field is guessed based on the energy consumption field.
Water consumption	Sensor	Integer	Litre(s)/sec.	Measured by a sensor for a period of time, and predicted for the future in Litre(s)/second.
Traffic situation For First Occupant	API	Varchar	*on-time, slightly-delayed* or *excessively-late*	Calculated via an external map service (such as OpenStreetMap) to guess the delay. It is categorized into *on-time, slightly-delayed* or *excessively-late*

Traffic Delay in Minutes for First Occupant	API/System	Integer	Minutes	Calculated via an external map service (such as OpenStreetMap) to guess the delay in minutes
Traffic situation For Second Occupant	API	Varchar	*on-time, slightly-delayed or excessively-late*	See *Traffic situation For First Occupant*
Traffic Delay in Minutes for Second Occupant	API/System	Integer	Minutes	See *Traffic Delay in Minutes for First Occupant*
Traffic situation For the Third Occupant	API	Varchar	*on-time, slightly-delayed or excessively-late*	See *Traffic situation For First Occupant*
Traffic Delay in Minutes for the Third Occupant	API/System	Integer	Minutes	See *Traffic Delay in Minutes for First Occupant*

Table 5.5: Immersion water heater energy consumption prediction's initial variables, sources, database field type, format, possible data range and some examples, with applied pre-processing actions. [AI(*): Artificial Intelligence, DM(**): Data Mining]

The data which were collected during the measurement phase of energy consumption and water rate data are illustrated in both Figure 5.9 and Figure 5.10 accordingly.

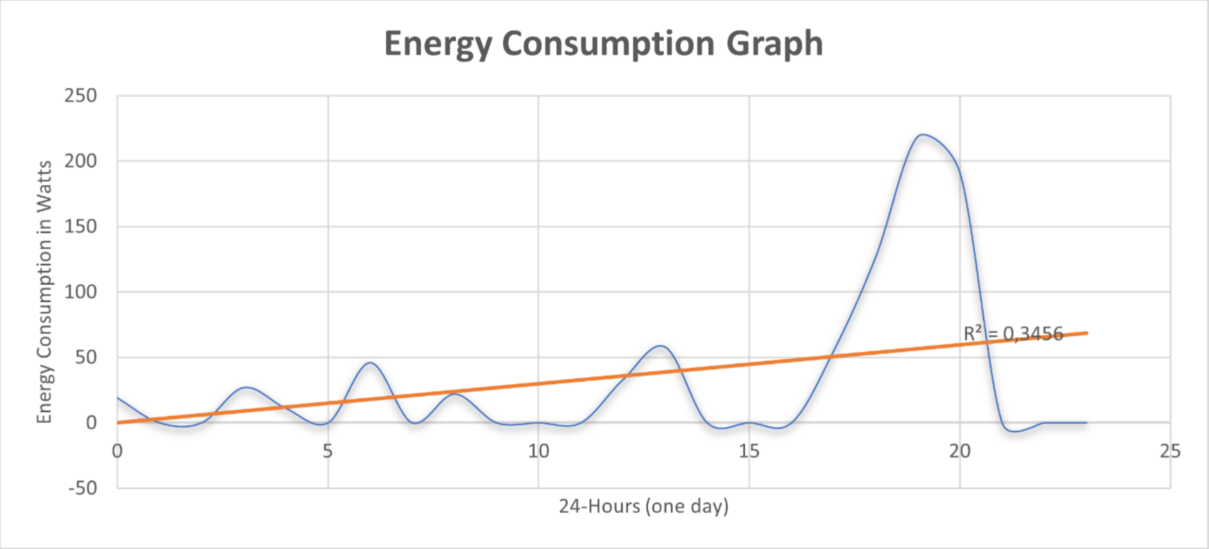


Figure 5.9: Measured Average Energy Consumption in 24-hours (in kWh), with Squared-R Value (Coefficient of determination) that shows the proportion of the variance in variables

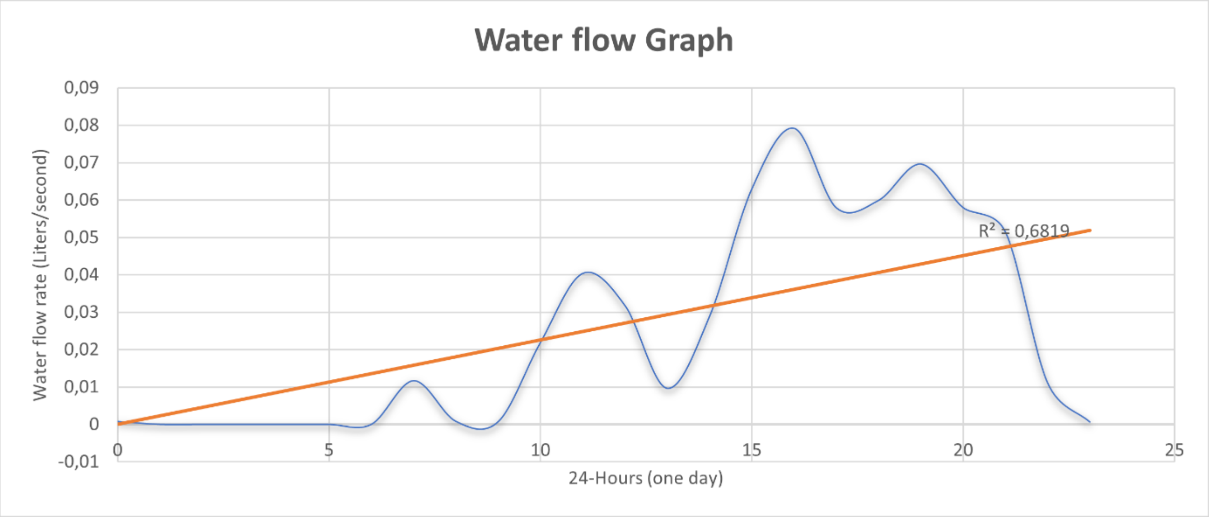


Figure 5.10: Measured Water Flow Rate in 24-hours (in Litre/second), with Squared-R Value (Coefficient of determination) that shows the proportion of the variance in variables

5.4.3. Data Preparation

This phase is considered one of the most important phases in CRISP-DM data mining because it has a direct impact on prediction quality and accuracy. In this phase, data runs through several preparation iterations including selecting, cleaning, constructing, integrating and formatting. The motivation behind preparing features is the high impact on the resulted model performance because removing irrelevant features results in an easy-to-understand, better performing and faster running model. Not all these steps are applicable, rather it depends on the nature of the case. The final decision is taken based on the final model, it went through several iterations till having satisfying results. The acceptance of the final results was measured using the usual regression metrics, which will be explained in detail in the next section, including Mean Square Error (MSE), Root Mean Square Error (RMSE), Mean Absolute Error (MAE) and the Coefficient of Determination (R2). The following steps were applied to the collected dataset.

5.4.3.1. High Percentage of Missing Values

Due to technical shortage of the used humidity sensor and the nature of the room where it is installed, the kitchen, which was rapidly changing on an hourly basis, it seems that a huge number of observations in the internal humidity feature is either N/A, null, zero or has unreadable corrupted values, therefore this feature was removed. Important to mention that binding the external humidity data retrieved from external

API with the internal humidity was not an option due to the huge difference between those values.

5.4.3.2. Low Variation Rates

In this stage features with almost the same value are detected and dropped. The variation rates are calculated automatically via MATLAB and Orange 3, based on the equation explained in section 3.3.7.

Inspecting the values reveals that some of them have a strong variance score, such as energy consumption, however other features were scoring a very low variance score such as the weather condition, which scored a 0.24877371, therefore this feature will be removed from the dataset.

5.4.3.3. Merging, Splitting Features

Increasing the data quality could be obtained by merging data from two or more features. It is not necessarily required that these features have a high feature-wise correlation. A clear example was detected while inspecting the external temperature values. External temperatures were retrieved from the weather API and also measured via a sensor. The externally collected data were separated into two values: temperatures measured on 5cm and 5 meters high. Surprisingly values were varying in some cases up to 150%. The best solution was to take the average of the three features: *External_Temperature_5cm*, *External_Temperature_5m*, and *External_Temperature_Measured*. The new feature *External_Temperature* was left because of the low correlation rate.

5.4.3.4. Correlation

There are two different types of correlations. Feature-wise and target-correlation. The feature-wise correlation occurs when two or more independent features show a high correlation rate. In this case, to achieve better performance and computing times, it is recommended to remove the feature with the lowest correlation level. However, the second type represents a high correlation between an independent feature(s) and the dependent target feature. High correlation levels are required to have good prediction results, where weak correlations, close to zero, are not relevant in the prediction process.

To obtain reliable data mining results it is necessary to inspect the relationship's strength among features in the dataset to find out the association or correlation score. This covers the correlation among features on one side, and between features and the target on the other side. Regardless of which formula is applied, the result must be always either negative, positive or zero-correlation, ranging between -1 and +1. In the literature, there are two main methods: Pearson and Spearman correlations. In fact, there is a third method called Kendall, however, it is similar to the Spearman method. The main difference between both approaches is the way how variables are handled in the pre-processing phase. The Pearson approach inspects the linear relationship of two or more values, so the value of the variable stands in the heart of the process, this leads to the fact that having any outliers or any data abnormality will affect the result. Therefore, it is necessary to apply different pre-processing actions before applying this method. Since the Spearman approach is not based on evaluating the values of the variables, rather on their rank-order, the values tend to change parallelly but not automatically at a steady rate. The correlation value ranges between -1 and +1. Where +1 is considered a perfect positive value that describes an exact rate of value change in the same direction between two variables. Any value bigger than 0.8 is considered a strong correlation, where the value bigger than 0.4 is judged as high. When the value stands between 0.4 and 0.2 it is described as correlated. Anything below 0.1 is not correlated, where 0 value describes the independent relationship between inspected values. The same classification is valid for the same negative values. Important to mention that negative values differ only in the direction, in other words, a -0.8 score means that both variables have a strong correlation where the first value decreases when the second increases.

As a result of the correlation analysis, it is expected to record a high correlation between the consumed energy and some variables such as the refrigerator fullness, the adjusted refrigerator temperature, the average total seconds of how long the refrigerator's door was opened per hour, and the internal temperature inside the household. Moreover, it is expected to measure a weak correlation between energy consumption and the external temperature, or the internal humidity. It is not expected to measure any negative correlation. Table 5.6 shows the resulted list of features after applying the previous data preparation techniques, with both correlations.

Variable	**Pearson's Correlation / Type**	**Spearman's Correlation / Type**
The average total seconds the door left opened per hour	+0.946 Strong correlation	+0.953 Strong correlation
Refrigerator's target (adjusted) temperature	+0.918 Strong correlation	+0.952 Strong correlation
Refrigerator fullness	+0.861 Strong correlation	+0.857 Strong correlation
The average time the refrigerator's door left opened per hour	+0.840 Strong correlation	+0.832 Strong correlation
Current number of occupants	+0.525 High correlation Note: This is corrected to (+0.923) after removing outliers	+0.919 Strong correlation
Household's internal temperature	+0.178 Not strong correlation Note: This is corrected to (+0.947) after removing outliers	+0.923 Strong correlation
Datetime	-0.142 Not strong correlation	-0.164 Not strong correlation
External temperature (outside the household)	-0.044 Does not correlate	-0.040 Does not correlate

Table 5.6: Correlation Analysis: Pearson and Spearman calculated for some variables related to the refrigerator experiment

As seen in the table the score's evaluation based on both methods is quite similar for most variables. Some differences are seen related to the *Household's internal temperature* and *Current number of occupants*. The reason is related to the number of outliers in both variables' datasets.

5.4.3.5. Anomaly Detection and Outliers

A visual data observation and calculating minimum, maximum and average of the features gave an indicator of having some anomalies and outliers, as seen in Figure 5.11 which illustrates two values: the number of occupants calculated using different machine learning techniques, and the internal temperature measured via an internal sensor. The peaks are the outliers which are shown in the graph.

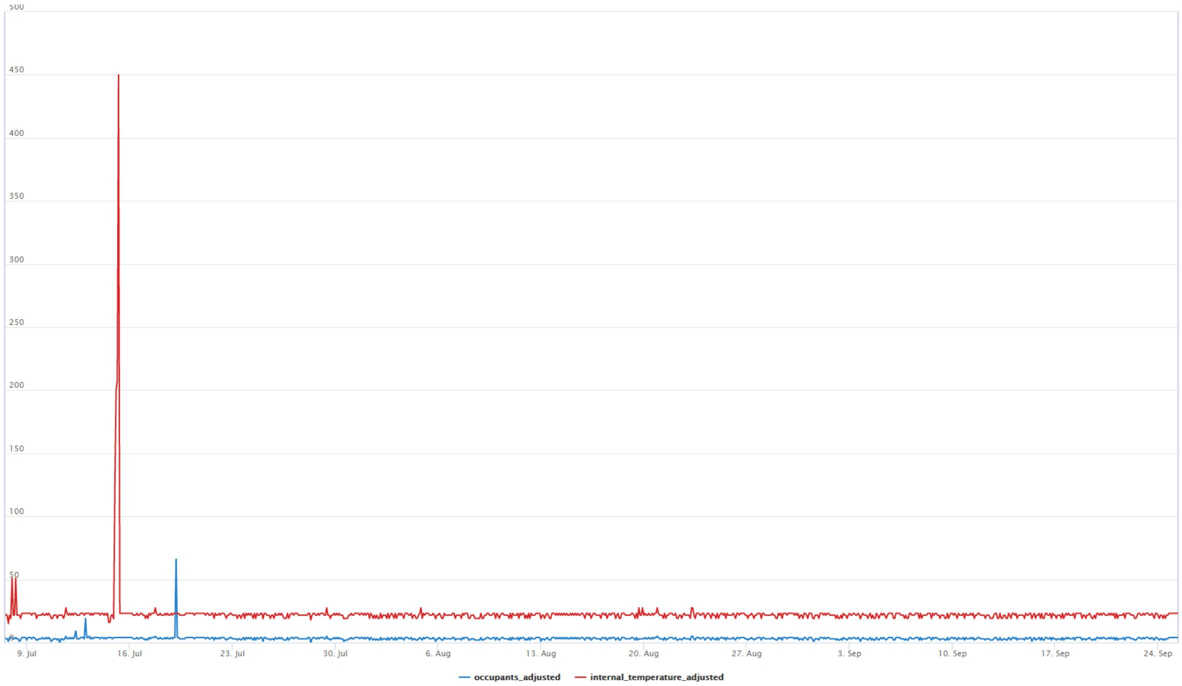


Figure 5.11: Number of occupants (blue) and the internal household temperature (red) with the peaks representing the outliers

Figure 5.12 shows the detected outliers for two features *Number of occupants* and *Household's internal temperature*, the data is shown on the right side after removing the outliers using the Covariance Estimator method, with a 2% contamination supported by a 1,0 fraction. Important to mention that re-calculating the target correlation with both features after removing the outliers showed much higher scores, *Household's internal temperature* Pearson scored +0.947, and *Current number of occupants* changed to +0.923, therefore both can be considered in the further data mining process.

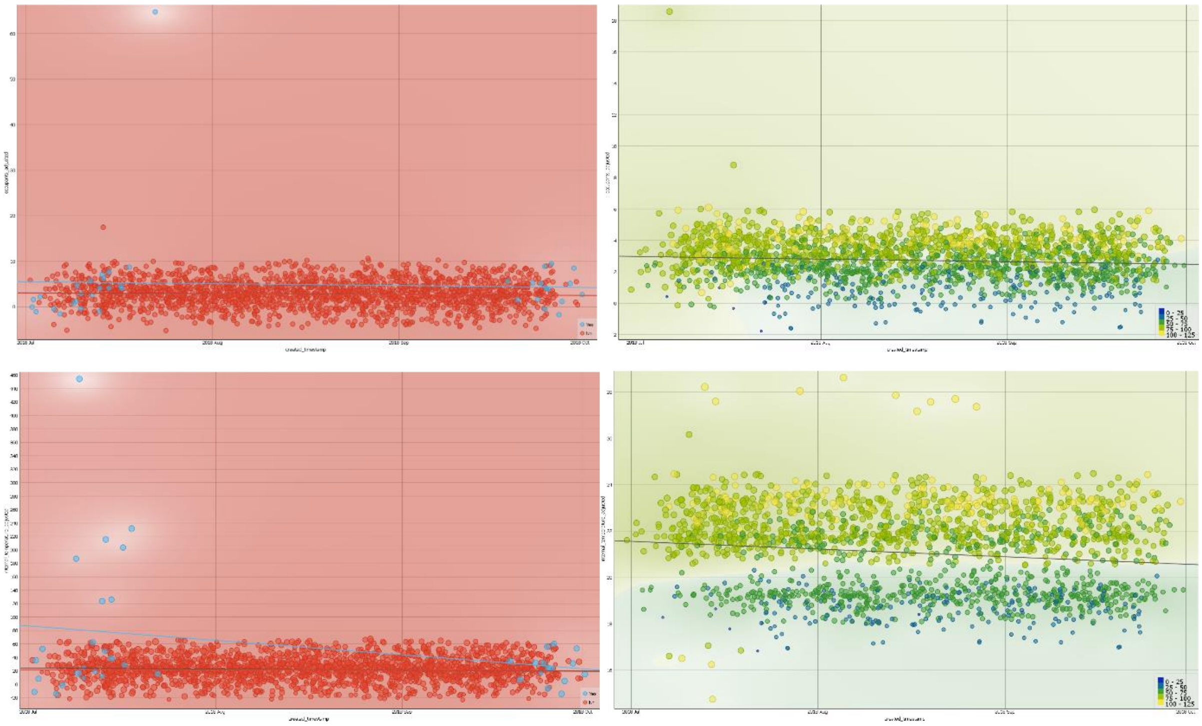


Figure 5.12: The top-left graph shows the outliers of the Number of occupants using the Covariance Estimator method, with a 2% contamination supported by a 1,0 fraction. Outliers (blue), Number of occupants (red). The graph on the top-right side shows the data after removing the outliers. The same thing repeated for the Household's Internal Temperature on the bottom-left and bottom-right graphs.

Further anomaly detection was carried on for the rest of the features, this was processed using an open-source data visualization, machine learning and data mining toolkit called Orange 3, developed by the University of Ljubljana [188] shown in Figure 5.13. The Outliers widget applies one of the four methods for outlier detection. All methods apply classification to the dataset. One-class SVM with non-linear kernels (RBF) performs well with non-Gaussian distributions, while a Covariance estimator works only for data with Gaussian distribution. One efficient way to perform outlier detection on moderately high dimensional datasets is to use the Local Outlier Factor algorithm. The algorithm computes a score reflecting the degree of abnormality of the observations. It measures the local density deviation of a given data point concerning its neighbours. Another efficient way of performing outlier detection in high-dimensional datasets is to use random forests (Isolation Forest). Results obtained from applying the mentioned workflow can be seen in the next chapter in section 6.2.1.3.

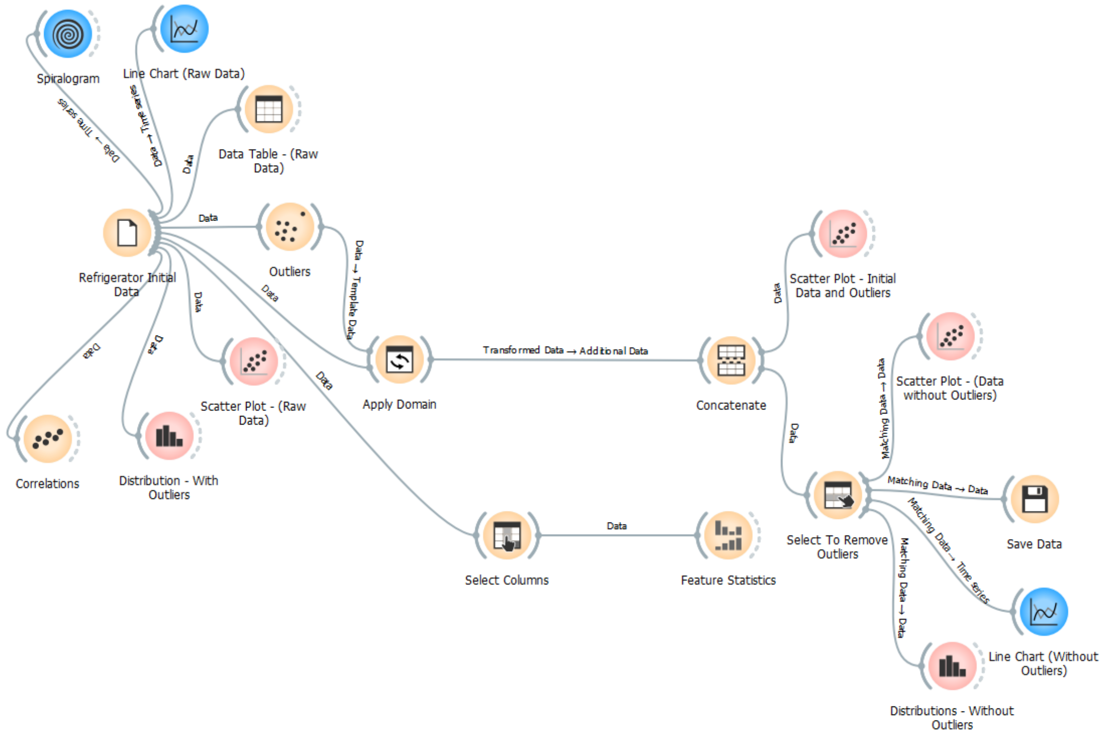

Spiralogram
Line Chart (Raw Data)
Data → Time series
Data → Time series
Data
Data Table - (Raw Data)
Data
Outliers
Data → Template Data
Refrigerator Initial Data
Data
Data
Data
Data
Data
Scatter Plot - (Raw Data)
Apply Domain
Transformed Data → Additional Data
Concatenate
Data
Scatter Plot - Initial Data and Outliers
Correlations
Distribution - With Outliers
Matching Data → Data
Scatter Plot - (Data without Outliers)
Data
Matching Data → Data
Save Data
Select To Remove Outliers
Matching Data → Time series
Select Columns
Data
Feature Statistics
Matching Data → Data
Line Chart (Without Outliers)
Distributions - Without Outliers

Figure 5.13: Orange 3 Workflow to remove Outliers

After applying the third method Local Outlier Factor, to detect and remove outliers in the target feature Figure 5.14 shows the results:

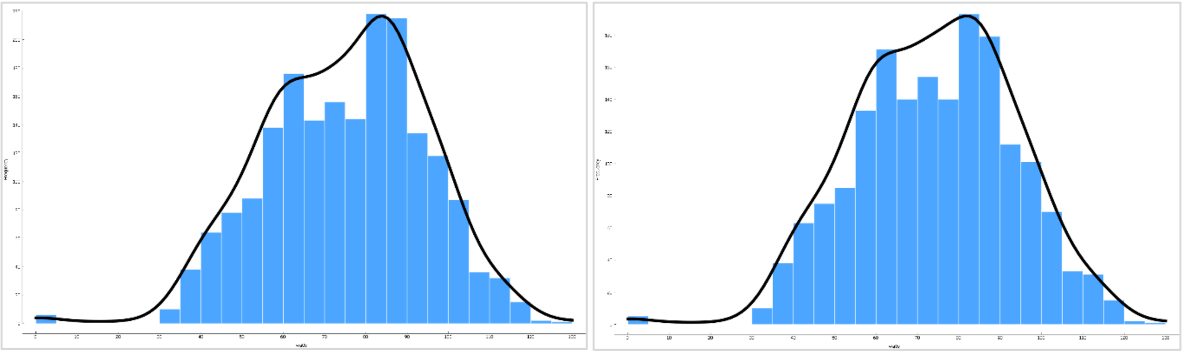


Figure 5.14: Refrigerator Energy Consumption with outliers (left), without outliers (right) presented using Kernel Density Distribution.

One of the possible problems which may reduce the model quality is the fact that the model is adapting itself to the training data, or what is called *overfitting*. It is highly required to identify, detect and prevent this problem because not taking any action against it makes the model tailored for the training data and has a very weak ability to correctly predict the target when having new different observations. This has been dealt with by splitting the data into 10 folds and using them all for training and validation. Another approach was to start with building a simple model then increasing the complexity by adding more and more features. This approach helps to detect and avoid underfitting and reach the good fit target. Early stopping may help prevent the overfitting problem. This can be done by detecting the point where the gap between prediction error of both training and validation is as small as possible and stopping the process.

5.4.4. Modelling and Applying Machine Learning Algorithms

The data source for the three implementation instances which were carried on for two sample appliances within the household are obtained from both observed and predicted data. Observed datasets were collected from various sensors, artificial intelligence techniques, and external APIs. However, predicted datasets were obtained from applying data mining and machine learning algorithms on the observed data to create a model with a high accuracy rate to produce predictions and discover patterns and mutual relationships. As mentioned in section 3.3, there are a number of different data mining techniques, each one has its own features, application area, nature of problems and algorithms. Important to mention that some algorithms can be applied for more than techniques such as Neural Networks which can be used for regression and classification problems. The application of a particular technique depends on the targeted feature and the nature of the dataset. For example, classification techniques are better in predicting labels, such as yes/no, on/off, however, regression techniques are specialised in predicting quantities. Following is the description of the applied policies and experiments.

Data preparation offered an insight into the exact problem description and defined the character of the needed prediction problem. In the previous sections, it is seen that there are several independent features with different correlations rates and diverging impacts on the dependent target feature (energy consumption). The required prediction should deliver continuous dependent values based on independent features, where the target is not dichotomous. This matches the definition of a regression problem where independent features are labelled and the target feature is a continuous variable, not classified. The next step is applying the prepared features on several candidate regression algorithms to obtain the required model. For this purpose following algorithms will be applied using different settings to have the best match to the problem:

1. Generalized Linear Regression – which is only used for regression problems, with applying different regularization parameters and strength such as Ridge Regression (L2), Lasso regression (L1) and Elastic net regression. Apart from the fact that multiple LR expects several features, both are identical.
2. Polynomial Regression – shows the regression line for multiple regressors. Polynomial expansion is a regulation of the degree of the polynomial that is used to transform the input data and has an effect on the shape of a curve. If polynomial expansion is set to 1 it means that untransformed data are used in the regression.
3. Tree – This algorithm is a simple algorithm, a precursor to Random Forest, that splits the data into nodes by class purity and can be used for both classification and regression problems.
4. Random Forest – is an ensemble learning method used for classification, regression and other tasks. Random Forest builds a set of decision trees. Each tree is developed from a bootstrap sample from the training data. When developing individual trees, an arbitrary subset of attributes is drawn (hence the term *Random*), from which the best attribute for the split is selected. The final model is based on the majority vote from individually developed trees in the forest.
5. Finally, the k-Nearest Neighbour (kNN) – Searches for K closest training examples in feature space and uses their average as a prediction.

One of the possible problems which may reduce the model quality is the fact that the model is adapting itself to the training data, or what is called *overfitting*. It is highly required to identify, detect and prevent this problem because not taking any action against it makes the model tailored for the training data and has a very weak ability to correctly predict the target when having new different observations. This has been dealt with by splitting the data into 10 folds and using them all for training and validation. Another approach was to start with building a simple model then increasing the complexity by adding more and more features. This approach helps to detect and avoid underfitting and reach the good fit target. Early stopping may help to prevent the overfitting problem. This can be done by detecting the point where the gap between prediction error of both training and validation is as small as possible and stopping the process.

5.4.5. Cross-Validation of Models

As mentioned in section 3.3.7., regression algorithms are suitable for prediction problems where a continuous value or a quantity should be predicted, however, the classification techniques are more suitable to predict discrete class labels such as On/Off, Yes, No. Each machine learning approach has its own validation criteria, as explained in the following sections.

5.4.5.1. Regression Techniques Cross-Validation Metrics

Predicting energy consumption for the first experiment is considered a regression problem predicted by applying regression algorithms because it attempts to predict continuous values [189] [190] [191], therefore only regression-specific evaluation metrics can give an insight on the suitability of a regression approach. For this reason, commonly known performance metrics such as confusion matrix, accuracy, and ROC Curve, which are suitable for classification algorithms, will not be used [192]. However, as explained in section 3.3.7, other mathematical equations are used, such as Mean-Squared-Error (MSE), Root-Mean-Square-Error (MRSE), Mean-Absolute-Error (MAE) and Coefficient of Determination (R2).

Calculating the previously mentioned metrics were performed using different software, MATLAB and Orange 3. Figure 5.15 shows the workflow designed for this purpose.

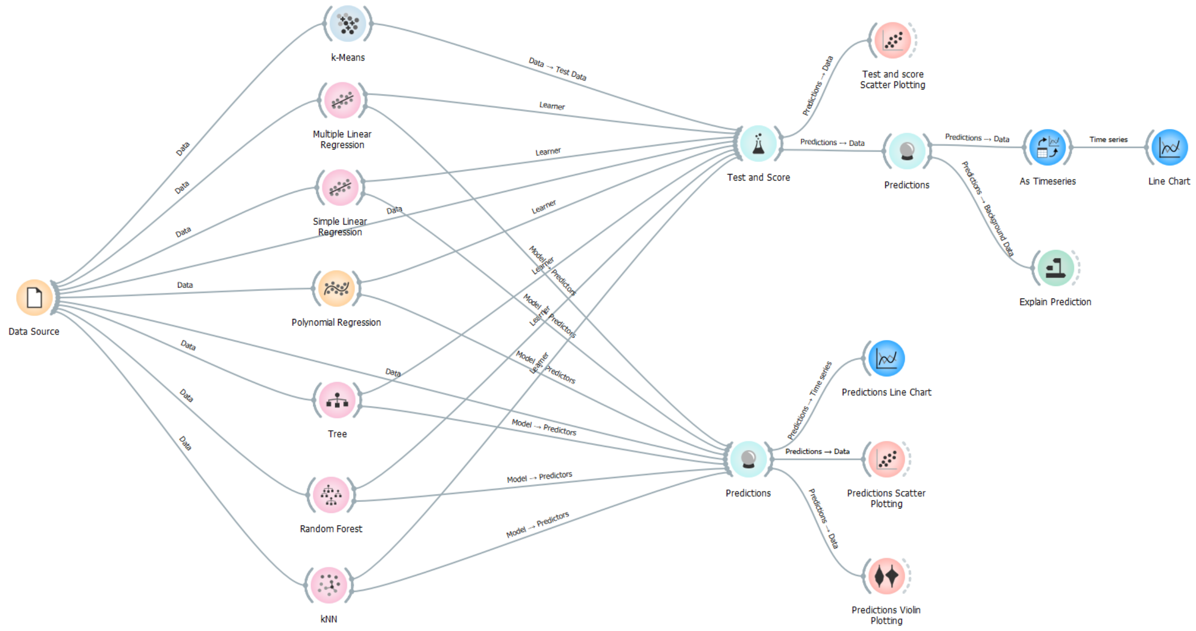


Figure 5.15: Orange 3 Workflow to Apply Various Regression Algorithms and Calculate Validation Metrics for the Refrigerator

All these metrics were calculated for the wattage variable, results will be illustrated in chapter 6 (*case study*).

5.4.5.2. Classification Techniques Cross-Validation Metrics

This machine learning approach is used for the second experiment where the discrete class label of the running periods of the immersion heater (switched on/off) needs to be predicted. Figure 5.16 shows the Orange 3 workflow used to feed the initial data, and the algorithms and the final validation criteria.

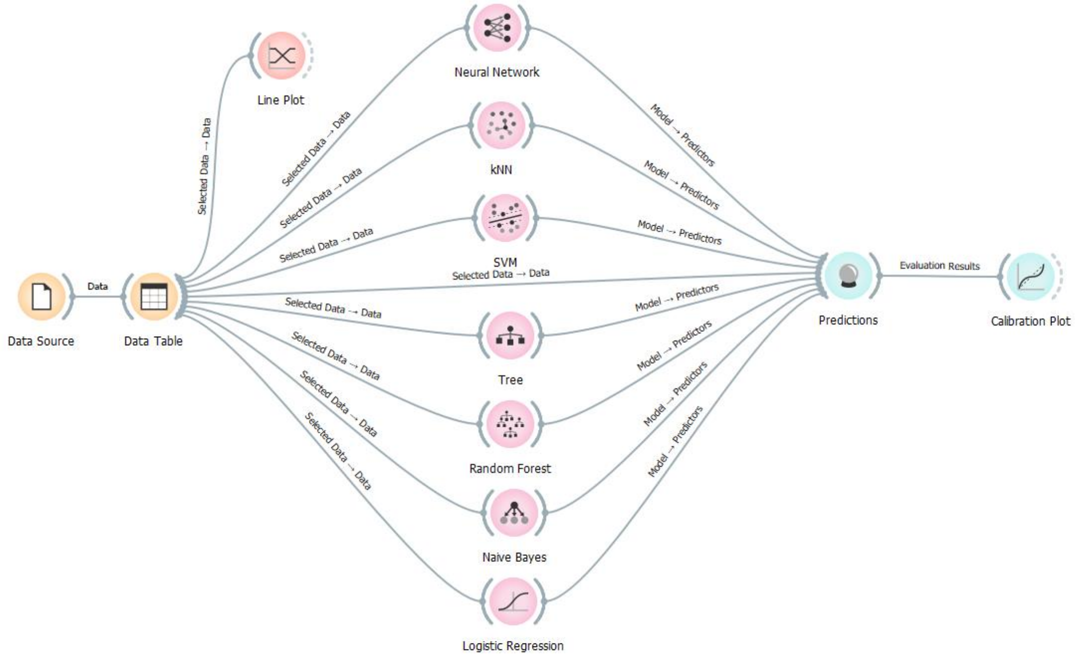
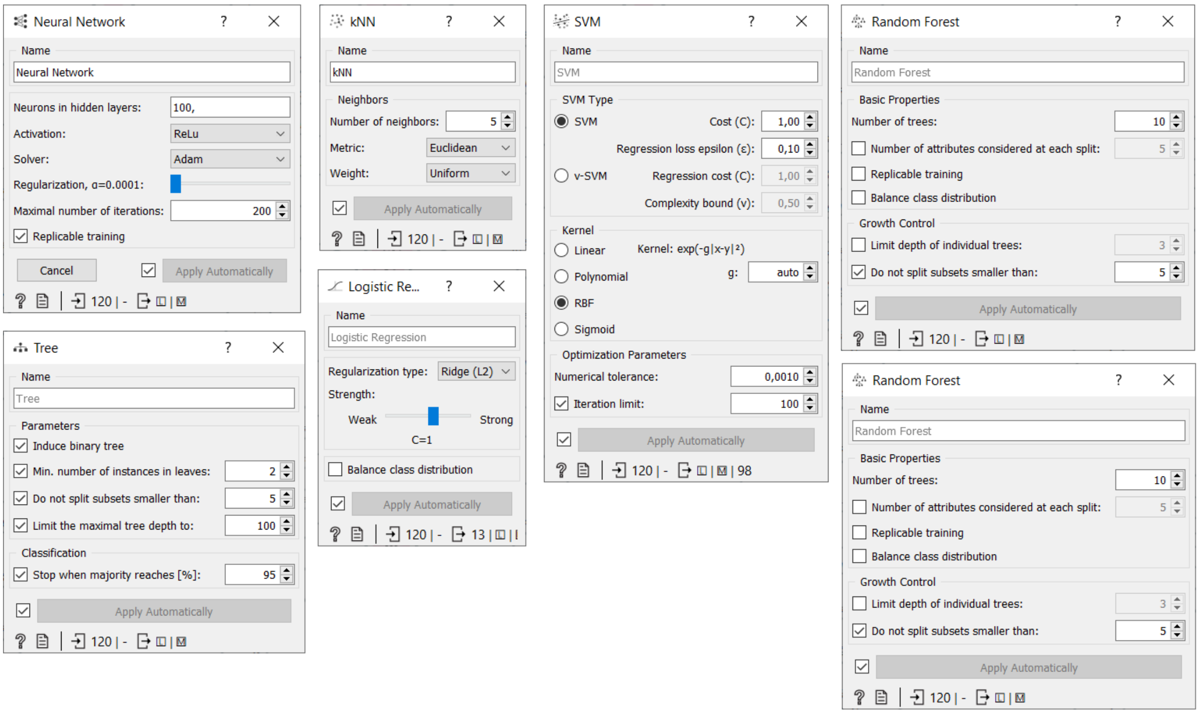


Figure 5.16: Orange 3 Workflow to Apply Various Classification Algorithms and Calculate Validation Metrics Fort he Immersion Heater

Figures 5.17 illustrate the default settings used in the evaluation software for the applied algorithms.



Neural Network

Name: Neural Network

Neurons in hidden layers: 100,

Activation: ReLu

Solver: Adam

Regularization, α=0.0001:

Maximal number of iterations: 200

☑ Replicable training

Tree

Name: Tree

Parameters

☑ Induce binary tree

☑ Min. number of instances in leaves: 2

☑ Do not split subsets smaller than: 5

☑ Limit the maximal tree depth to: 100

Classification

☑ Stop when majority reaches [%]: 95

kNN

Name: kNN

Neighbors

Number of neighbors: 5

Metric: Euclidean

Weight: Uniform

Logistic Re...

Name: Logistic Regression

Regularization type: Ridge (L2)

Strength: Weak — Strong, C=1

☐ Balance class distribution

SVM

Name: SVM

SVM Type

● SVM — Cost (C): 1,00; Regression loss epsilon (ε): 0,10

○ ν-SVM — Regression cost (C): 1,00; Complexity bound (ν): 0,50

Kernel

○ Linear — Kernel: exp(-g|x-y|²)

○ Polynomial — g: auto

● RBF

○ Sigmoid

Optimization Parameters

Numerical tolerance: 0,0010

☑ Iteration limit: 100

Random Forest

Name: Random Forest

Basic Properties

Number of trees: 10

☐ Number of attributes considered at each split: 5

☐ Replicable training

☐ Balance class distribution

Growth Control

☐ Limit depth of individual trees: 3

☑ Do not split subsets smaller than: 5

Random Forest

Name: Random Forest

Basic Properties

Number of trees: 10

☐ Number of attributes considered at each split: 5

☐ Replicable training

☐ Balance class distribution

Growth Control

☐ Limit depth of individual trees: 3

☑ Do not split subsets smaller than: 5

Figure 5.17: Orange 3 default settings of the applied classification algorithms

		Actual	
		Positive	Negative
Predicted	Positive	True Positive	False Positive
	Negative	False Negative	True Negative

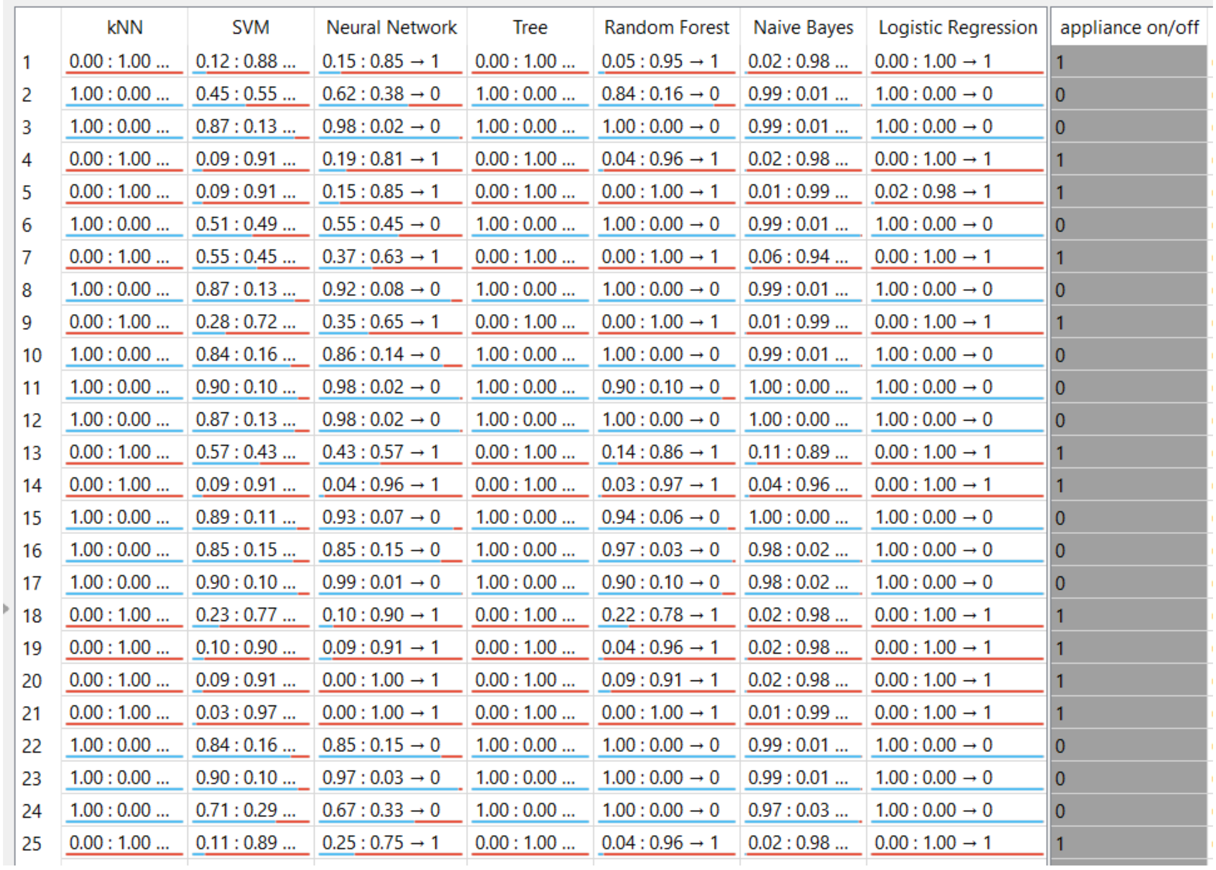
Table 5.7: Predictions Possible Outcomes Compared to the Real Values

Here we notice that the validation metrics are different from the ones in the previous regression refrigerator experiment. These metrics are:

Area Under ROC Curve (AUC) – As explained in section 3.3.3., This metric simply represents the probability to rank a randomly chosen positive example higher than a randomly chosen negative example. Referring to Table 5.7, this metric depends on calculating two different values: True Positive Rate (TPR), and False Positive Rate (FPR), then plotting these results on an XY dimension. AUC is the resulting area beneath the line.

Classification Accuracy (CA) – This is considered the typical metric which is calculated by taking the proportion of true outcomes divided by the entire quantity of all observed cases. A detailed explanation can be seen in section 3.3.3.

Results are considered good when it approaches the 100% target. As will be shown in chapter 6, every CA result that goes over 95% is considered good enough to consider the algorithm. Figure 5.18 shows part of the used data and resulted predictions using the mentioned algorithms (kNN, SVM, Neural Network, Tree, Random Forest, Naïve Bayes and Logistic Regression). Comparing the value in the column *appliance on/off* with the predicted value in each algorithm resulted in showing red underlines which indicate the mistaken prediction and blue underlines which indicate a correct prediction.



	kNN	SVM	Neural Network	Tree	Random Forest	Naive Bayes	Logistic Regression	appliance on/off
1	0.00 : 1.00 ...	0.12 : 0.88 ...	0.15 : 0.85 → 1	0.00 : 1.00 ...	0.05 : 0.95 → 1	0.02 : 0.98 ...	0.00 : 1.00 → 1	1
2	1.00 : 0.00 ...	0.45 : 0.55 ...	0.62 : 0.38 → 0	1.00 : 0.00 ...	0.84 : 0.16 → 0	0.99 : 0.01 ...	1.00 : 0.00 → 0	0
3	1.00 : 0.00 ...	0.87 : 0.13 ...	0.98 : 0.02 → 0	1.00 : 0.00 ...	1.00 : 0.00 → 0	0.99 : 0.01 ...	1.00 : 0.00 → 0	0
4	0.00 : 1.00 ...	0.09 : 0.91 ...	0.19 : 0.81 → 1	0.00 : 1.00 ...	0.04 : 0.96 → 1	0.02 : 0.98 ...	0.00 : 1.00 → 1	1
5	0.00 : 1.00 ...	0.09 : 0.91 ...	0.15 : 0.85 → 1	0.00 : 1.00 ...	0.00 : 1.00 → 1	0.01 : 0.99 ...	0.02 : 0.98 → 1	1
6	1.00 : 0.00 ...	0.51 : 0.49 ...	0.55 : 0.45 → 0	1.00 : 0.00 ...	1.00 : 0.00 → 0	0.99 : 0.01 ...	1.00 : 0.00 → 0	0
7	0.00 : 1.00 ...	0.55 : 0.45 ...	0.37 : 0.63 → 1	0.00 : 1.00 ...	0.00 : 1.00 → 1	0.06 : 0.94 ...	0.00 : 1.00 → 1	1
8	1.00 : 0.00 ...	0.87 : 0.13 ...	0.92 : 0.08 → 0	1.00 : 0.00 ...	1.00 : 0.00 → 0	0.99 : 0.01 ...	1.00 : 0.00 → 0	0
9	0.00 : 1.00 ...	0.28 : 0.72 ...	0.35 : 0.65 → 1	0.00 : 1.00 ...	0.00 : 1.00 → 1	0.01 : 0.99 ...	0.00 : 1.00 → 1	1
10	1.00 : 0.00 ...	0.84 : 0.16 ...	0.86 : 0.14 → 0	1.00 : 0.00 ...	1.00 : 0.00 → 0	0.99 : 0.01 ...	1.00 : 0.00 → 0	0
11	1.00 : 0.00 ...	0.90 : 0.10 ...	0.98 : 0.02 → 0	1.00 : 0.00 ...	0.90 : 0.10 → 0	1.00 : 0.00 ...	1.00 : 0.00 → 0	0
12	1.00 : 0.00 ...	0.87 : 0.13 ...	0.98 : 0.02 → 0	1.00 : 0.00 ...	1.00 : 0.00 → 0	1.00 : 0.00 ...	1.00 : 0.00 → 0	0
13	0.00 : 1.00 ...	0.57 : 0.43 ...	0.43 : 0.57 → 1	0.00 : 1.00 ...	0.14 : 0.86 → 1	0.11 : 0.89 ...	0.00 : 1.00 → 1	1
14	0.00 : 1.00 ...	0.09 : 0.91 ...	0.04 : 0.96 → 1	0.00 : 1.00 ...	0.03 : 0.97 → 1	0.04 : 0.96 ...	0.00 : 1.00 → 1	1
15	1.00 : 0.00 ...	0.89 : 0.11 ...	0.93 : 0.07 → 0	1.00 : 0.00 ...	0.94 : 0.06 → 0	1.00 : 0.00 ...	1.00 : 0.00 → 0	0
16	1.00 : 0.00 ...	0.85 : 0.15 ...	0.85 : 0.15 → 0	1.00 : 0.00 ...	0.97 : 0.03 → 0	0.98 : 0.02 ...	1.00 : 0.00 → 0	0
17	1.00 : 0.00 ...	0.90 : 0.10 ...	0.99 : 0.01 → 0	1.00 : 0.00 ...	0.90 : 0.10 → 0	0.98 : 0.02 ...	1.00 : 0.00 → 0	0
18	0.00 : 1.00 ...	0.23 : 0.77 ...	0.10 : 0.90 → 1	0.00 : 1.00 ...	0.22 : 0.78 → 1	0.02 : 0.98 ...	0.00 : 1.00 → 1	1
19	0.00 : 1.00 ...	0.10 : 0.90 ...	0.09 : 0.91 → 1	0.00 : 1.00 ...	0.04 : 0.96 → 1	0.02 : 0.98 ...	0.00 : 1.00 → 1	1
20	0.00 : 1.00 ...	0.09 : 0.91 ...	0.00 : 1.00 → 1	0.00 : 1.00 ...	0.09 : 0.91 → 1	0.02 : 0.98 ...	0.00 : 1.00 → 1	1
21	0.00 : 1.00 ...	0.03 : 0.97 ...	0.00 : 1.00 → 1	0.00 : 1.00 ...	0.00 : 1.00 → 1	0.01 : 0.99 ...	0.00 : 1.00 → 1	1
22	1.00 : 0.00 ...	0.84 : 0.16 ...	0.85 : 0.15 → 0	1.00 : 0.00 ...	1.00 : 0.00 → 0	0.99 : 0.01 ...	1.00 : 0.00 → 0	0
23	1.00 : 0.00 ...	0.90 : 0.10 ...	0.97 : 0.03 → 0	1.00 : 0.00 ...	1.00 : 0.00 → 0	0.99 : 0.01 ...	1.00 : 0.00 → 0	0
24	1.00 : 0.00 ...	0.71 : 0.29 ...	0.67 : 0.33 → 0	1.00 : 0.00 ...	1.00 : 0.00 → 0	0.97 : 0.03 ...	1.00 : 0.00 → 0	0
25	0.00 : 1.00 ...	0.11 : 0.89 ...	0.25 : 0.75 → 1	0.00 : 1.00 ...	0.04 : 0.96 → 1	0.02 : 0.98 ...	0.00 : 1.00 → 1	1

Figure 5.18: Part of target Feature appliance on/off and the resulted prediction using different classification algorithms (kNN, SVM, Neural Network, Tree, Random Forest, Naïve Bayes and Logistic Regression). Red underlines indicate the mistaken prediction, whereas blue underlines indicate a correct prediction.

5.5. RECHS Application

Due to the nature of the implementation which covers several areas inside the household, and communication with external systems and platforms, there is a need to have a central point to put all related systems, APIs, GUIs and interfaces which are required to implement the various strategies mentioned previously, in one place to enhance the modularity and the encapsulation, besides keeping the maintenance efforts in its minimum level. This central software application is called RECHS which stands for Reduction of Energy Consumption in Household Sector.

The software application is implemented using the waterfall project management approach, therefore describing, documenting, and prototyping the requirements precisely was considered as an essential milestone during the project's life cycle, which included the justification of the built artefact, description of methods, aims, deliverables and plans, covering functional and non-functional requirements, documenting detailed user's requirements via use-cases, sequences and activity diagrams. Technically, the application consists of a farm of microservices built on a Java Spring Boot version 2.0.3, the communication was established over RESTful API specifications that interact with the frontend via a middleware built via PHP 7, jQuery 3 and Bootstrap v5.0. The application's infrastructure overview and general software architecture are illustrated in Figure 5.19.

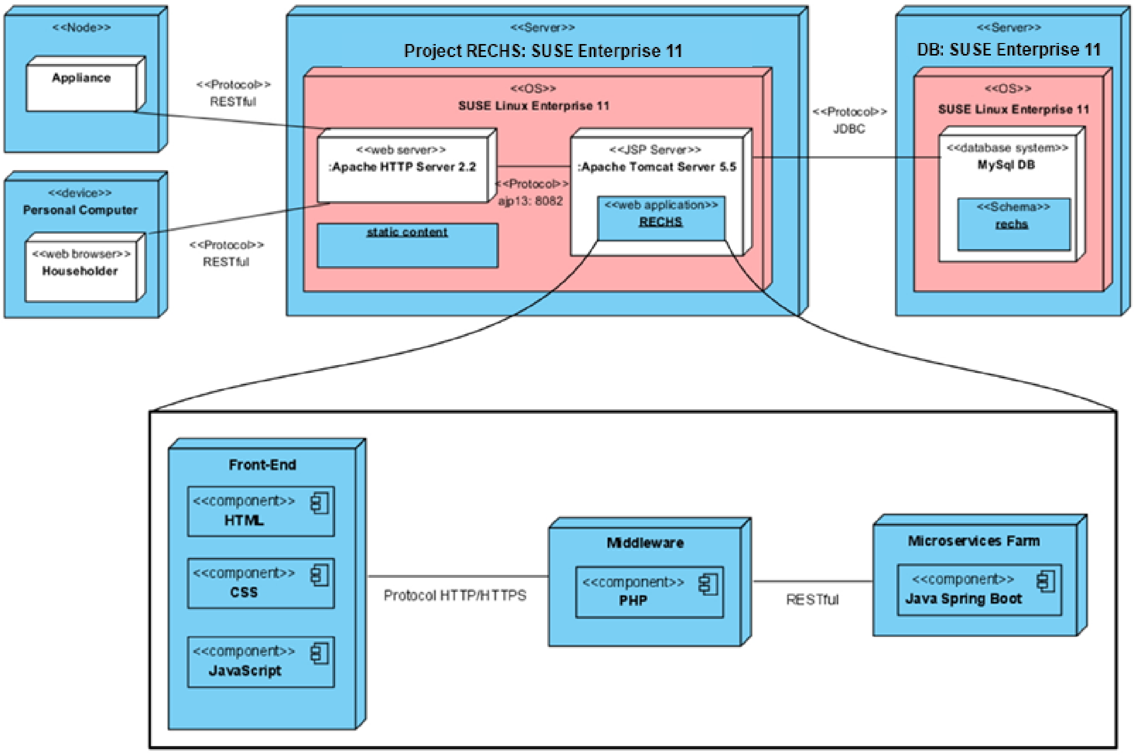
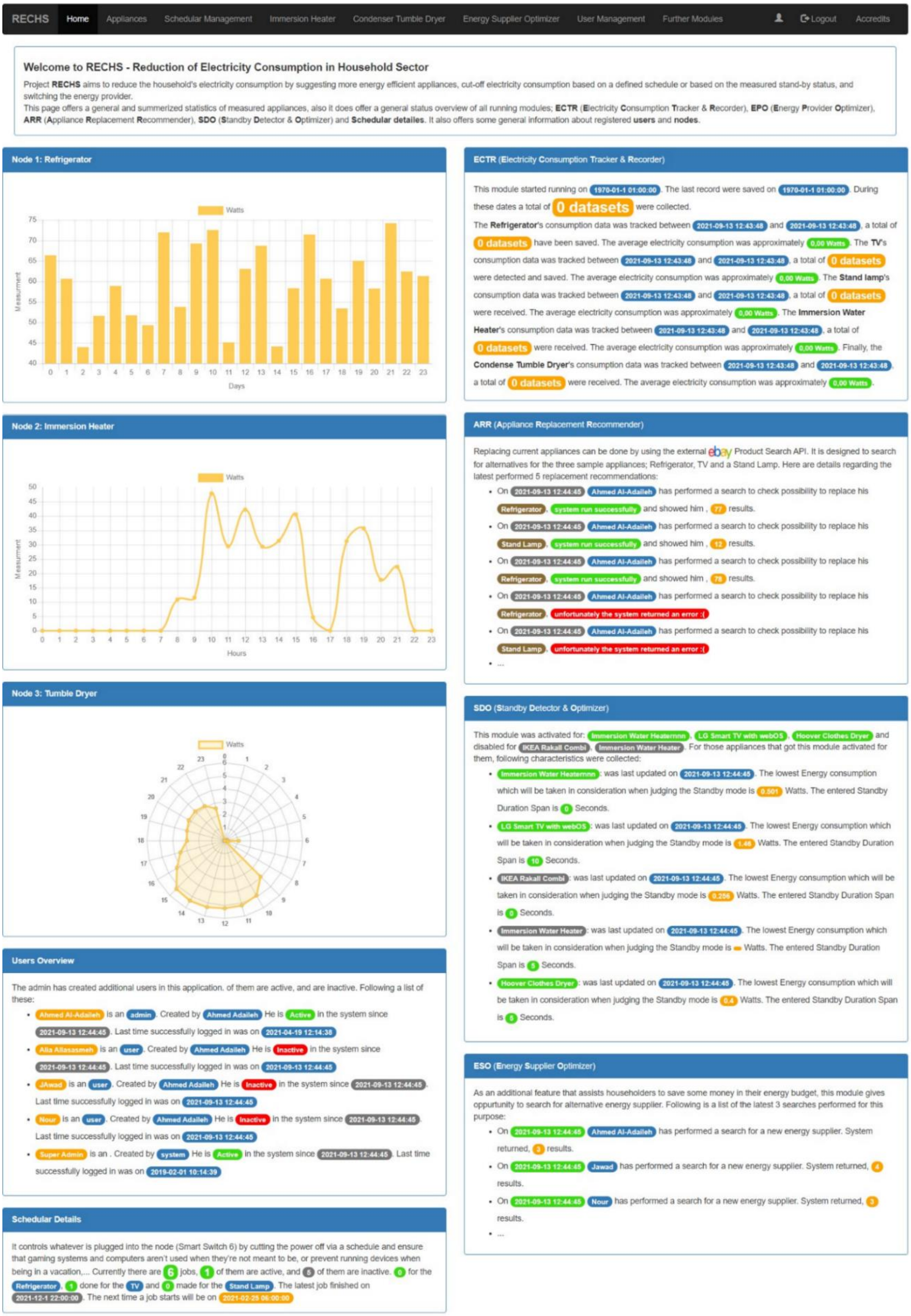


Figure 5.19: RECHS System Architecture

Figures 5.20, 5.21 and 5.22 illustrate some sections of the RECHS application. Including the home page, the appliances overview which shows some details of the tracked appliances. Important to mention that some of the details are entered by administrators, and the rest were automatically measured and collected via different sensors. Besides these components, the application has its own user management, schedule management (Figure 5.23) and local energy suppliers management.



Welcome to RECHS - Reduction of Electricity Consumption in Household Sector

Project **RECHS** aims to reduce the household's electricity consumption by suggesting more energy efficient appliances, cut-off electricity consumption based on a defined schedule or based on the measured stand-by status, and switching the energy provider.
This page offers a general and summerized statistics of measured appliances, also it does offer a general status overview of all running modules; **ECTR** (**E**lectricity **C**onsumption **T**racker & **R**ecorder), **EPO** (**E**nergy **P**rovider **O**ptimizer), **ARR** (**A**ppliance **R**eplacement **R**ecommender), **SDO** (**S**tandby **D**etector & **O**ptimizer) and **Schedular detailes**. It also offers some general information about registered **users** and **nodes**.

Node 1: Refrigerator

Node 2: Immersion Heater

Node 3: Tumble Dryer

Users Overview

The admin has created additional users in this application. of them are active, and are inactive. Following a list of these:

- Ahmed Al-Adaileh is an admin. Created by Ahmed Adaileh He is Active in the system since 2021-09-13 12:44:45. Last time successfully logged in was on 2021-04-19 12:14:38
- Alia Allasasmeh is an user. Created by Ahmed Adaileh He is Inactive in the system since 2021-09-13 12:44:45. Last time successfully logged in was on 2021-09-13 12:44:45
- JAwad is an user. Created by Ahmed Adaileh He is Inactive in the system since 2021-09-13 12:44:45. Last time successfully logged in was on 2021-09-13 12:44:45
- Nour is an user. Created by Ahmed Adaileh He is Inactive in the system since 2021-09-13 12:44:45. Last time successfully logged in was on 2021-09-13 12:44:45
- Super Admin is an . Created by system He is Active in the system since 2021-09-13 12:44:45. Last time successfully logged in was on 2019-02-01 10:14:39

Schedular Details

It controls whatever is plugged into the node (Smart Switch 6) by cutting the power off via a schedule and ensure that gaming systems and computers aren't used when they're not meant to be, or prevent running devices when being in a vacation,... Currently there are 6 jobs, 1 of them are active, and 5 of them are inactive. 0 for the Refrigerator, 1 done for the TV and 0 made for the Stand Lamp. The latest job finished on 2021-12-1 22:00:00. The next time a job starts will be on 2021-02-25 06:00:00

ECTR (**E**lectricity **C**onsumption **T**racker & **R**ecorder)

This module started running on 1970-01-1 01:00:00. The last record were saved on 1970-01-1 01:00:00. During these dates a total of 0 datasets were collected.

The **Refrigerator**'s consumption data was tracked between 2021-09-13 12:43:48 and 2021-09-13 12:43:48, a total of 0 datasets have been saved. The average electricity consumption was approximately 0,00 Watts. The **TV**'s consumption data was tracked between 2021-09-13 12:43:48 and 2021-09-13 12:43:48, a total of 0 datasets were detected and saved. The average electricity consumption was approximately 0,00 Watts. The **Stand lamp**'s consumption data was tracked between 2021-09-13 12:43:48 and 2021-09-13 12:43:48, a total of 0 datasets were received. The average electricity consumption was approximately 0,00 Watts. The **Immersion Water Heater**'s consumption data was tracked between 2021-09-13 12:43:48 and 2021-09-13 12:43:48, a total of 0 datasets were received. The average electricity consumption was approximately 0,00 Watts. Finally, the **Condense Tumble Dryer**'s consumption data was tracked between 2021-09-13 12:43:48 and 2021-09-13 12:43:48, a total of 0 datasets were received. The average electricity consumption was approximately 0,00 Watts.

ARR (**A**ppliance **R**eplacement **R**ecommender)

Replacing current appliances can be done by using the external ebay Product Search API. It is designed to search for alternatives for the three sample appliances; Refrigerator, TV and a Stand Lamp. Here are details regarding the latest performed 5 replacement recommendations:

- On 2021-09-13 12:44:45 Ahmed Al-Adaileh has performed a search to check possibility to replace his Refrigerator, system run successfully and showed him , 77 results.
- On 2021-09-13 12:44:45 Ahmed Al-Adaileh has performed a search to check possibility to replace his Stand Lamp, system run successfully and showed him , 12 results.
- On 2021-09-13 12:44:45 Ahmed Al-Adaileh has performed a search to check possibility to replace his Refrigerator, system run successfully and showed him , 78 results.
- On 2021-09-13 12:44:45 Ahmed Al-Adaileh has performed a search to check possibility to replace his Refrigerator, unfortunately the system returned an error :(
- On 2021-09-13 12:44:45 Ahmed Al-Adaileh has performed a search to check possibility to replace his Stand Lamp, unfortunately the system returned an error :(
- ...

SDO (**S**tandby **D**etector & **O**ptimizer)

This module was activated for: Immersion Water Heaternnn, LG Smart TV with webOS, Hoover Clothes Dryer and disabled for IKEA Rakall Combi, Immersion Water Heater. For those appliances that got this module activated for them, following characteristics were collected:

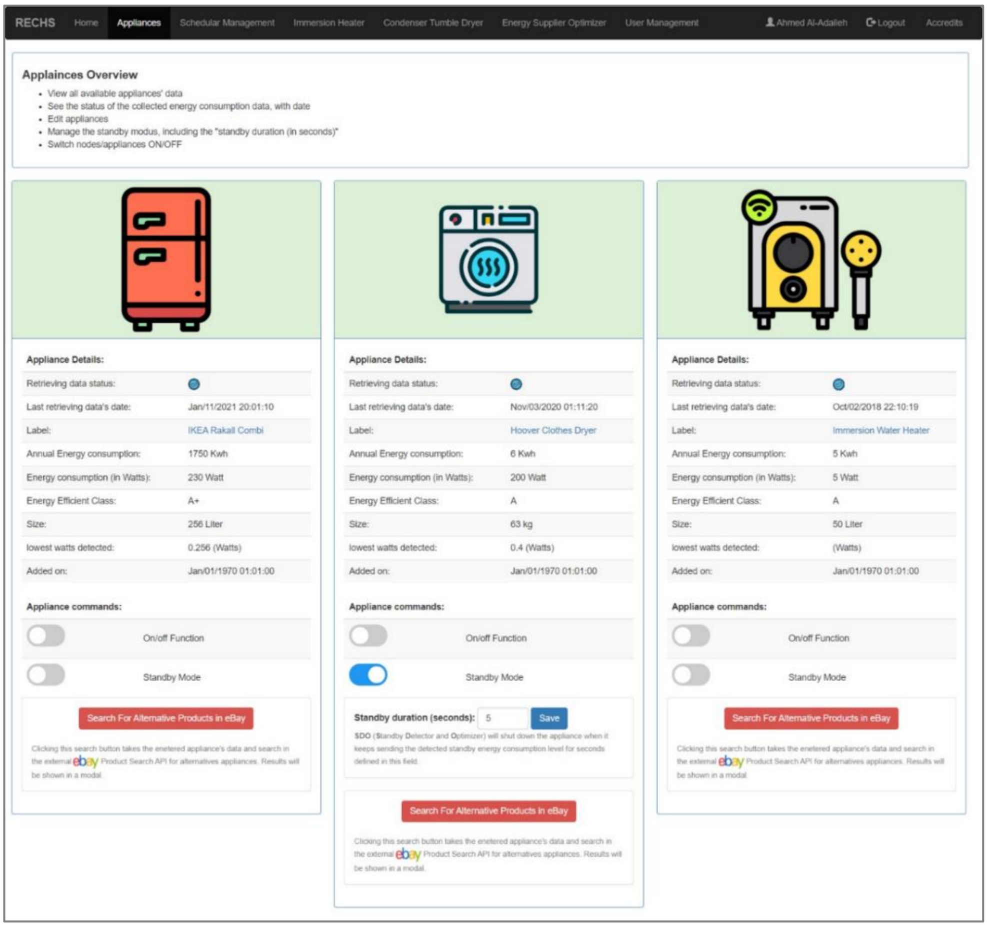
- Immersion Water Heaternnn: was last updated on 2021-09-13 12:44:45. The lowest Energy consumption which will be taken in consideration when judging the Standby mode is 0.501 Watts. The entered Standby Duration Span is 0 Seconds.
- LG Smart TV with webOS: was last updated on 2021-09-13 12:44:45. The lowest Energy consumption which will be taken in consideration when judging the Standby mode is 1.46 Watts. The entered Standby Duration Span is 10 Seconds.
- IKEA Rakall Combi: was last updated on 2021-09-13 12:44:45. The lowest Energy consumption which will be taken in consideration when judging the Standby mode is 0.256 Watts. The entered Standby Duration Span is 0 Seconds.
- Immersion Water Heater: was last updated on 2021-09-13 12:44:45. The lowest Energy consumption which will be taken in consideration when judging the Standby mode is Watts. The entered Standby Duration Span is 5 Seconds.
- Hoover Clothes Dryer: was last updated on 2021-09-13 12:44:45. The lowest Energy consumption which will be taken in consideration when judging the Standby mode is 0.4 Watts. The entered Standby Duration Span is 5 Seconds.

ESO (**E**nergy **S**upplier **O**ptimizer)

As an additional feature that assists householders to save some money in their energy budget, this module gives oppurtunity to search for alternative energy supplier. Following is a list of the latest 3 searches performed for this purpose:

- On 2021-09-13 12:44:45 Ahmed Al-Adaileh has performed a search for a new energy supplier. System returned, 3 results.
- On 2021-09-13 12:44:45 Jawad has performed a search for a new energy supplier. System returned, 4 results.
- On 2021-09-13 12:44:45 Nour has performed a search for a new energy supplier. System returned, 3 results.
- ...

Figure 5.20: RECHS Homepage Overview



Applainces Overview

- View all available appliances' data
- See the status of the collected energy consumption data, with date
- Edit appliances
- Manage the standby modus, including the "standby duration (in seconds)"
- Switch nodes/appliances ON/OFF

Appliance Details:

Retrieving data status:	●
Last retrieving data's date:	Jan/11/2021 20:01:10
Label:	IKEA Rakall Combi
Annual Energy consumption:	1750 Kwh
Energy consumption (in Watts):	230 Watt
Energy Efficient Class:	A+
Size:	256 Liter
lowest watts detected:	0.256 (Watts)
Added on:	Jan/01/1970 01:01:00

Appliance commands:

○ On/off Function

○ Standby Mode

Search For Alternative Products in eBay

Clicking this search button takes the enetered appliance's data and search in the external ebay Product Search API for alternatives appliances. Results will be shown in a modal.

Appliance Details:

Retrieving data status:	●
Last retrieving data's date:	Nov/03/2020 01:11:20
Label:	Hoover Clothes Dryer
Annual Energy consumption:	6 Kwh
Energy consumption (in Watts):	200 Watt
Energy Efficient Class:	A
Size:	63 kg
lowest watts detected:	0.4 (Watts)
Added on:	Jan/01/1970 01:01:00

Appliance commands:

○ On/off Function

● Standby Mode

Standby duration (seconds): 5 Save

SDO (**S**tandby **D**etector and **O**ptimizer) will shut down the appliance when it keeps sending the detected standby energy consumption level for seconds defined in this field.

Search For Alternative Products in eBay

Clicking this search button takes the enetered appliance's data and search in the external ebay Product Search API for alternatives appliances. Results will be shown in a modal.

Appliance Details:

Retrieving data status:	●
Last retrieving data's date:	Oct/02/2018 22:10:19
Label:	Immersion Water Heater
Annual Energy consumption:	5 Kwh
Energy consumption (in Watts):	5 Watt
Energy Efficient Class:	A
Size:	50 Liter
lowest watts detected:	(Watts)
Added on:	Jan/01/1970 01:01:00

Appliance commands:

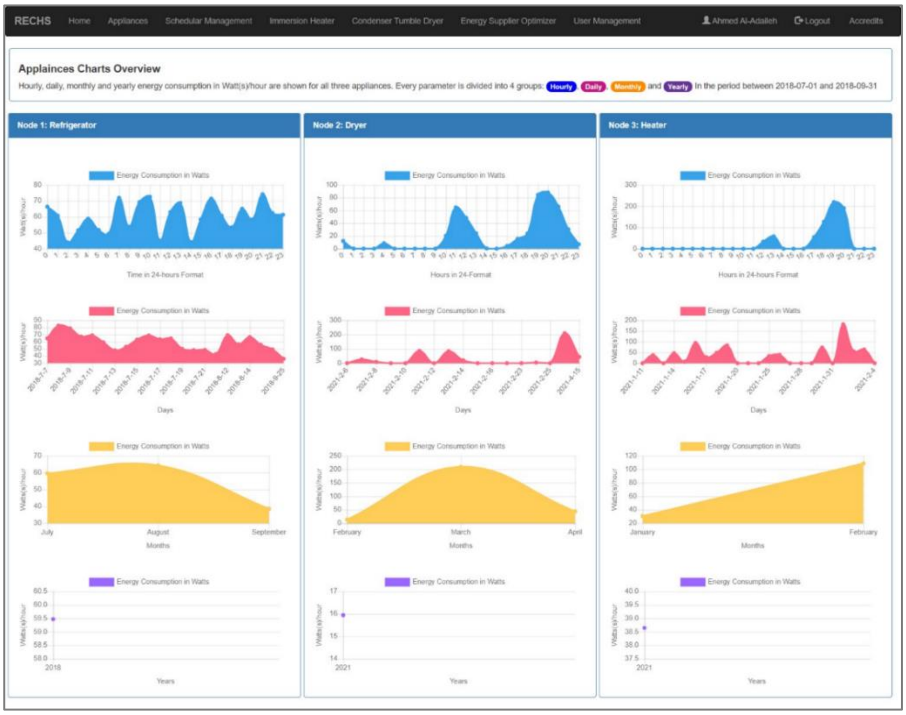
○ On/off Function

○ Standby Mode

Search For Alternative Products in eBay

Clicking this search button takes the enetered appliance's data and search in the external ebay Product Search API for alternatives appliances. Results will be shown in a modal.

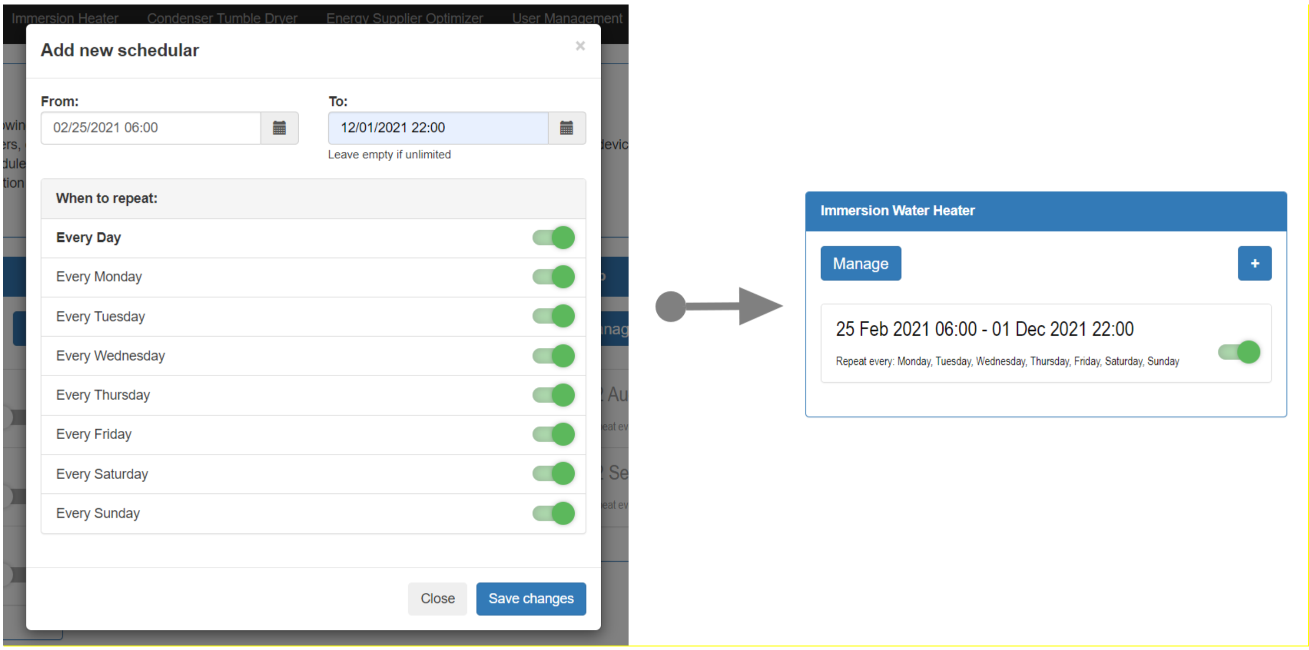
Figure 5.21: RECHS Appliances Overview where the substitution function can be triggered.



Applainces Charts Overview

Hourly, daily, monthly and yearly energy consumption in Watt(s)/hour are shown for all three appliances. Every parameter is divided into 4 groups: Hourly, Daily, Monthly and Yearly In the period between 2018-07-01 and 2018-09-31

Figure 5.22: RECHS Application – Appliances historical energy consumption overview grouped by for different periods: 24-hours, days, months and years



Add new schedular

From: 02/25/2021 06:00

To: 12/01/2021 22:00

Leave empty if unlimited

When to repeat:

- Every Day ●
- Every Monday ●
- Every Tuesday ●
- Every Wednesday ●
- Every Thursday ●
- Every Friday ●
- Every Saturday ●
- Every Sunday ●

Immersion Water Heater

25 Feb 2021 06:00 - 01 Dec 2021 22:00 ●

Repeat every: Monday, Tuesday, Wednesday, Thursday, Friday, Saturday, Sunday

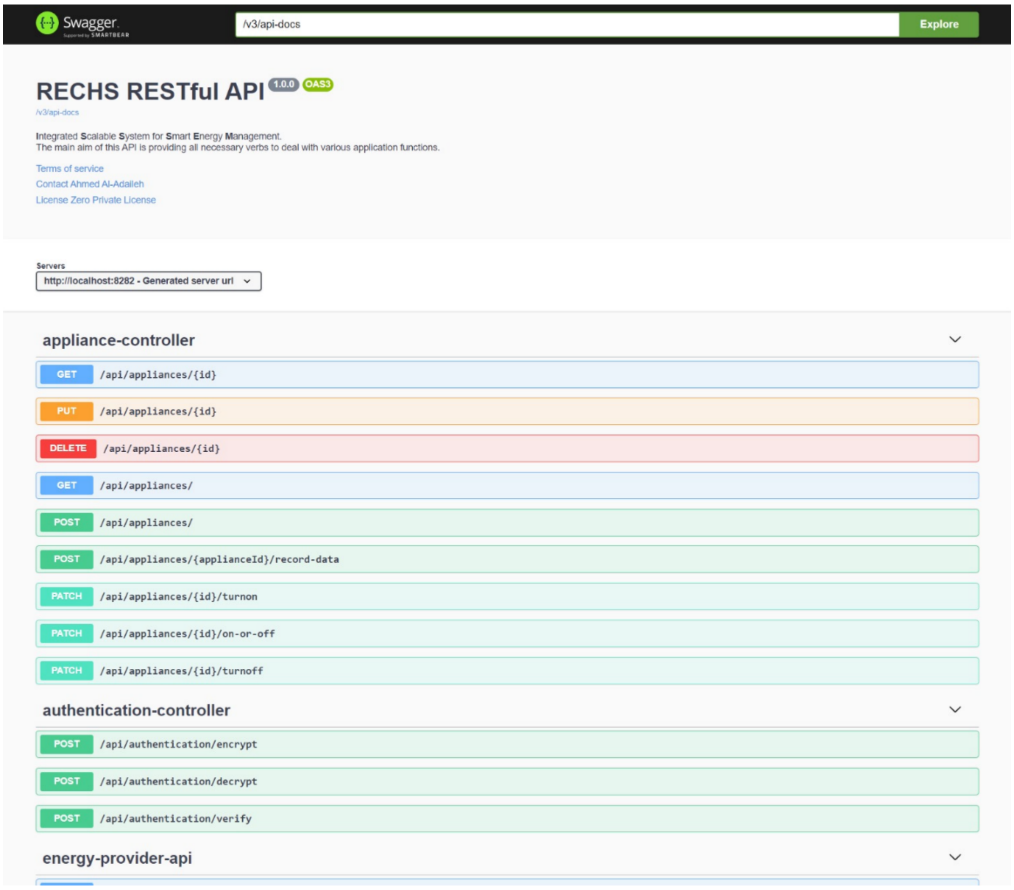
Figure 5.23: Create Manual Schedule to run the Immersion Water Heater

5.6. Framework's Additional Components

The framework consists of several other components than previously mentioned. These are integrated into the overall system to support the accomplishment of major and minor functions such as enhancing the encapsulation, supporting the integration and data assimilation which combines numeric models with observations. In this section, a number of the most important components will be highlighted.

5.6.1. Microservices and The RESTful API

As explained in section 3.2. microservices are split up into a set of distributed services, mainly categorised into five main groups: appliances, authentication, energy providers, APIs management and basic error management. Important to mention that the developed farm of microservices differs from the Service Oriented Architecture SOA approach in many ways, firstly it is smaller in size, has abounded context and serves a single purpose, designed to perform a high degree of independence. Figure 5.24 illustrates the farm of microservices developed for this purpose using Java SpringBoot, and figure 5.25 shows an example RESTful verb responsible for recording energy consumption for any appliance.



RECHS RESTful API

/v3/api-docs

Integrated Scalable System for Smart Energy Management.
The main aim of this API is providing all necessary verbs to deal with various application functions.

Terms of service
Contact Ahmed Al-Adaileh
License Zero Private License

Servers: http://localhost:8282 - Generated server url

appliance-controller

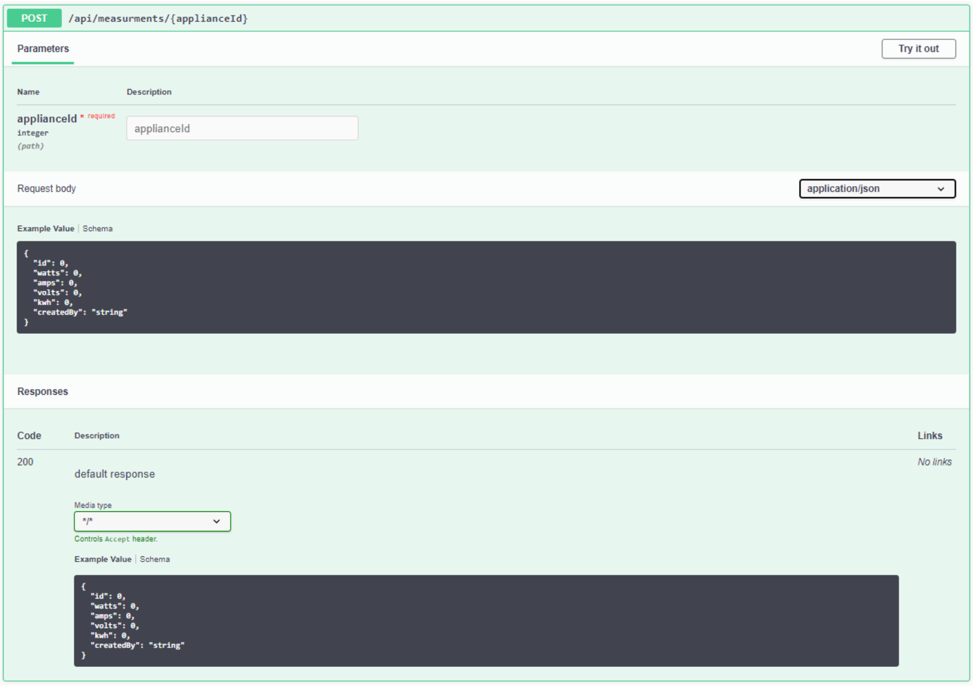
- GET /api/appliances/{id}
- PUT /api/appliances/{id}
- DELETE /api/appliances/{id}
- GET /api/appliances/
- POST /api/appliances/
- POST /api/appliances/{applianceId}/record-data
- PATCH /api/appliances/{id}/turnon
- PATCH /api/appliances/{id}/on-or-off
- PATCH /api/appliances/{id}/turnoff

authentication-controller

- POST /api/authentication/encrypt
- POST /api/authentication/decrypt
- POST /api/authentication/verify

energy-provider-api

Figure 5.24: Part of the Microservices farm and some example RESTFUL Verbs



POST /api/measurments/{applianceId}

Parameters

Name	Description
applianceId * required integer (path)	applianceId

Request body: application/json

```
{
  "id": 0,
  "watts": 0,
  "amps": 0,
  "volts": 0,
  "kwh": 0,
  "createdBy": "string"
}
```

Responses

Code	Description	Links
200	default response	No links

Media type: */*

Controls Accept header.

```
{
  "id": 0,
  "watts": 0,
  "amps": 0,
  "volts": 0,
  "kwh": 0,
  "createdBy": "string"
}
```

Figure 5.25: Example RESTful Verb: Recording energy consumption for any appliance

As mentioned in section 4.2. (Quality factors), performance can be broken down into three main factors: response time, throughput and utilisation. The following load tests were carried on using a load testing tool called *JMeter* to measure the response time by simulating simultaneous requests sent by 5000 users in parallel, to show the suitability of this paradigm to support one of the most important attributes of the proposed framework which is the scalability and to handle the performance of the microservices. Figure 5.26 shows the response time measured for two sample webservices *authentication* and *energy consumption wattage recording* done by 5000 users against one instance of the related webservice. In this figure it is seen that the response time is almost zero for all requests done in the first 80 seconds (from 13:31:30 till 13:32:50), the response time measured in milliseconds starts increasing gradually for both webservices till reaching approximately ~28 milliseconds for the *authentication* and ~20 milliseconds for the *energy consumption wattage recording* after about 23 minutes and 20 seconds, at 13:54:50. From this point forward till the end of the measurement period which ended at 13:55:30, the response varies for both webservices up and down. This behaviour may have occurred due to the resources-sharing of the laptop where both client and server and the JMeter tool are running together with other applications. Important to mention that the *authentication* webservice requires a longer response time than the *energy consumption wattage recording* because of the nature of the SQL enquiry, the *authentication* requires running SELECT statements, which imply processing and waiting time, where the *energy consumption wattage recording* relies only on INSERT action which runs much faster since it only requires adding a new line to the table without inquiring any existing data.

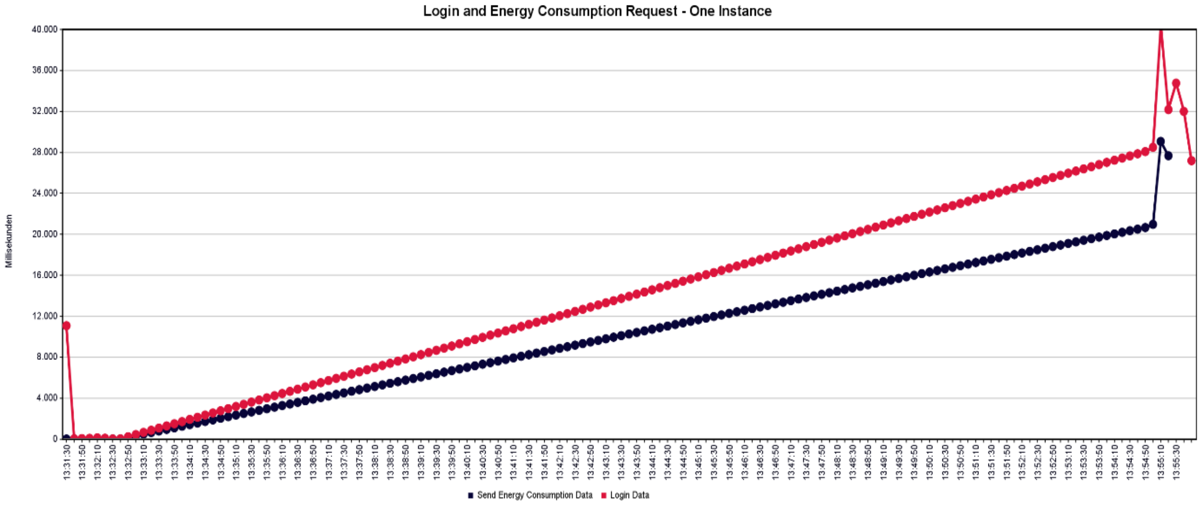


Figure 5.26: Response time measurement of two requests login and energy consumption wattage recording measured with one instance of the related webservices

Comparing this figure with the next figure 5.27, which measure the same parameters except for the number of instances which is increased to 5 instances. We notice a similar behaviour of both webservices in the first 23minutes and 20 seconds (from 13:31:30 till ~13:54:50), where the response time is almost zero for both webservices in the first 100 seconds, then it rapidly increases till reaching its maximum after ~24 minutes (at ~13:55:40). At this point, when the response time reaches 28 milliseconds, the other 4 instances get allocated to process the coming requests, so we start observing a rapid decrease of the response time, which means an increase in the performance. This turnaround behaviour is seen nearly till the end of the experiment time (at 14:27:20). Important to notice that deploying the 4 instances (at ~13:55:40) did have more effect on the *authentication* than the *energy consumption wattage recording* because of the internal caching mechanism on the database level, where the result of SELECT statements is cached and used without a need to repeatedly running the query. As seen in the previous figure 5.27, towards the end of the experiment an irregularity of the response time can be observed for both services, this might occur when the same hardware, in this case, the laptop, is shared among clients, server and other applications running parallelly.

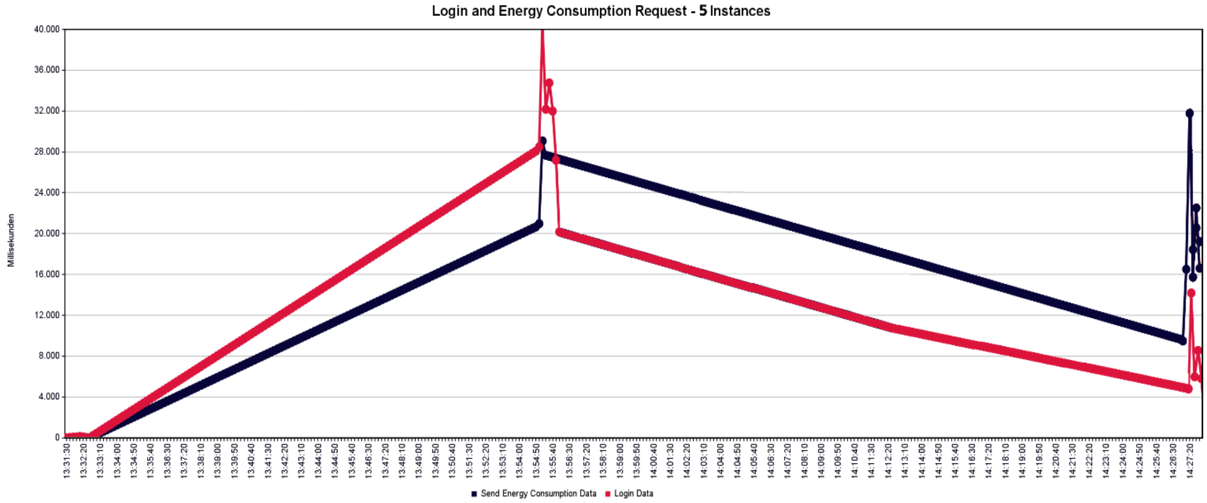


Figure 5.27: Response time measurement of two requests login and energy consumption wattage recording measured with 5 instances of the related webservices

Figure 5.28 illustrates a comparison of aggregated data parameters including average, median, the minimum and maximum response time (measured in milliseconds) for both requests *login* and *energy consumption recorder*.

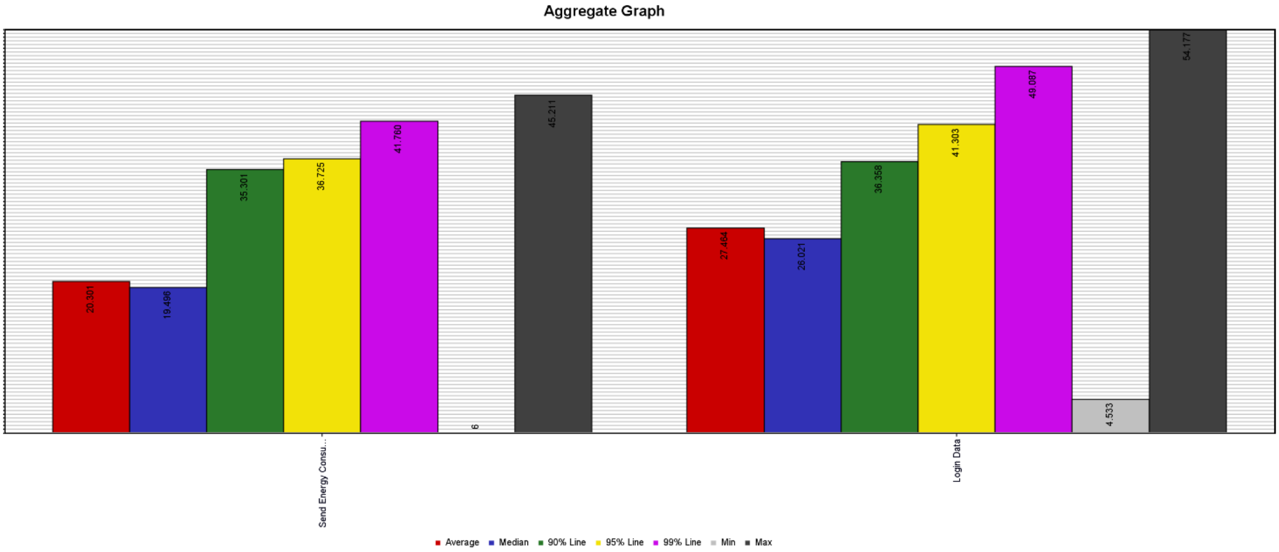


Figure 5.28: A comparison of aggregated data parameters including Average, Median, Minimum and Maximum Response time (measured in milliseconds) for both requests login and energy consumption recorder

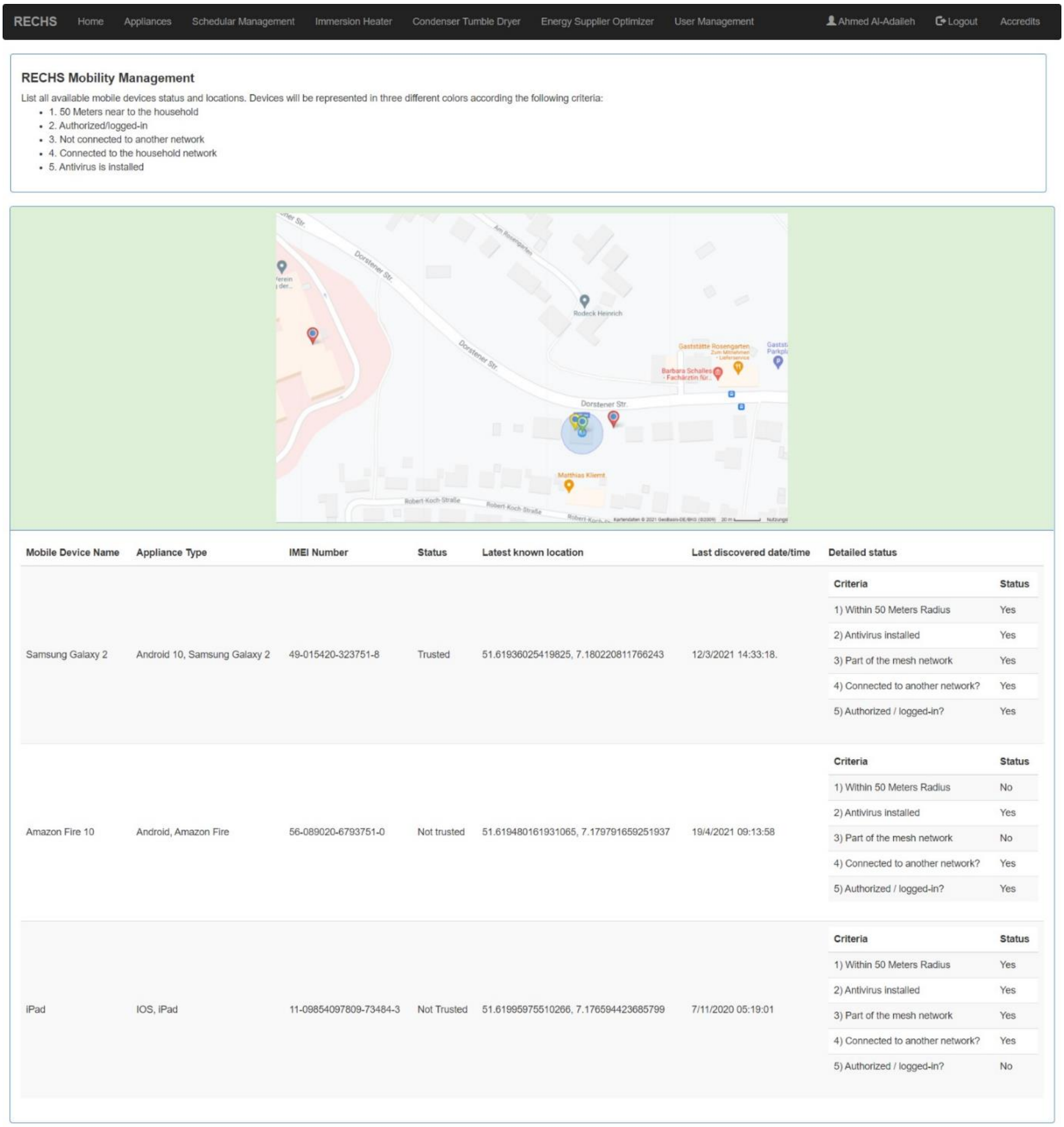
5.6.2. Data Transformers

Due to the fact that the proposed framework can handle all data collected from different sources having different formats, varying from images, to binaries, or even ASCII with different data structures, it is essential to implement a number of data transformers to assure a proper data flow and reliable results. Moreover, the framework is built to support almost all kinds of household appliances, the conventional and digital ones, so there is a need to obtain and process output parameters from these appliances. This approach supports the strength point of the framework to deal with smart and conventional appliances. Where smart appliances can directly generate processible data, and images can be taken from conventional appliances settings panels then converted to processible ASCII formats using these data transformers. An example is shown in Figure 5.3 to read the refrigerator temperature setting panel and convert it into an integer.

5.6.3. Mobility Management

Running a system based on a mesh network requires a lot of attention to the security simple because each node must be trusted, and therefore any attack either physically by stealing the node, or digitally over the LAN may threaten the whole system. For this purpose, securing the mobile devices gained huge importance because these mobile devices could be part of other networks, and they are parallelly used for other purposes such as making phone calls. The module shown in Figure 5.29 used for this purpose categorize the tracked devices into *TRUSTED* or *NOT TRUSTED* based on evaluating the following five questions, where a device is considered *NOT TRUSTED* if at least one question is answered with *NO*.

1. Is the device physically located within 50 meters radius of the household?
2. Is the device currently part of the mesh network related to the household?
3. Is the device currently *only* connected to the mesh network?
4. Is there antivirus software installed?
5. Is the device currently logged in/authorized?



RECHS Mobility Management

List all available mobile devices status and locations. Devices will be represented in three different colors according the following criteria:

- 1. 50 Meters near to the household
- 2. Authorized/logged-in
- 3. Not connected to another network
- 4. Connected to the household network
- 5. Antivirus is installed

Mobile Device Name	Appliance Type	IMEI Number	Status	Latest known location	Last discovered date/time	Detailed status	
Samsung Galaxy 2	Android 10, Samsung Galaxy 2	49-015420-323751-8	Trusted	51.61936025419825, 7.180220811766243	12/3/2021 14:33:18.	Criteria	Status
						1) Within 50 Meters Radius	Yes
						2) Antivirus installed	Yes
						3) Part of the mesh network	Yes
						4) Connected to another network?	Yes
						5) Authorized / logged-in?	Yes
Amazon Fire 10	Android, Amazon Fire	56-089020-6793751-0	Not trusted	51.619480161931065, 7.179791659251937	19/4/2021 09:13:58	Criteria	Status
						1) Within 50 Meters Radius	No
						2) Antivirus installed	Yes
						3) Part of the mesh network	No
						4) Connected to another network?	Yes
						5) Authorized / logged-in?	Yes
iPad	IOS, iPad	11-09854097809-73484-3	Not Trusted	51.61995975510266, 7.176594423685799	7/11/2020 05:19:01	Criteria	Status
						1) Within 50 Meters Radius	Yes
						2) Antivirus installed	Yes
						3) Part of the mesh network	Yes
						4) Connected to another network?	Yes
						5) Authorized / logged-in?	No

Figure 5.29: The module used to track and evaluate the trusted mobile devices

5.7. Conclusion

This chapter has presented a detailed description of the process which will be followed to implement the proposed framework on two different sample appliances within the household. It started with providing analysis of the household appliances categories: uninterruptible, instant, and schedulable devices, and explained a number of strategies and policies applied to reduce the energy consumption which are: energy-based appliance substation policy, usage percentage-based appliance substitution policy, and automatic scheduling of running periods policy. Then, provided a detailed description of the experimental settings, covering all self-assembled systems, off-the-shelf hardware and software which will be utilised during the implementation phase.

Since the data is the main pillar in this research, a detailed review of the data structure, data gathering, and processing methodology CRISP-DM was introduced. This included several stages starting from understanding the business, then understanding the data including mentioning the data limitations and challenges faced, together with the data sources used across the entire implementation phase, followed by a listing of the initial data structure of both experiments: refrigerator experiment, and water immersion heater experiment. The next sections were describing the methods used to prepare the data in a way to make it ready for the next modelling phase. The data preparation phase has concentrated on several steps: firstly, detect and eliminate the variables with high missing values rate. Secondly, detect and eliminate variables with low variation rates. Thirdly, merging or splitting variables to increase the data quality and score good points during the evaluation stage will be done separately. Fourthly, calculating the correlation between different columns, which lead to removing feature with high correlations, finally describing the anomaly detection and outliers detection done to spot the unusual and bizarre values and eliminate them accordingly.

After the data was prepared, the turn comes to applying the related machine learning algorithms to produce models which will be used for predicting data. Due to the different nature of both experiments, the regression algorithms were applied in the first refrigerator experiment because the required prediction should deliver continuous dependent values based on independent features, where the target is not dichotomous, which is the amount of the consumed energy, however, in the second water immersion heater experiment, the classification algorithms were applied because the discrete class label of the running periods of the immersion heater (switched on/off) needs to be predicted. A number of workflows designed using the Orange 3 platform to perform data analysis and visualization, were illustrated. Finally, a number of evaluation metrics for both regression and classification were discussed.

The last part of this chapter covered several components beginning with the core Java-based application, called RECHS, which was developed to merge all components and modules under one umbrella. Then explaining the load testing tool called *JMeter* which simulates requests and measures a number of important performance parameters. Moreover, covering the microservices paradigm and RESTful API approach and their role to support the framework’s main characteristics: integrability and scalability. As will be seen in the next case study chapter, a number of data transformers are explained to illustrate the framework’s capability of transforming data from any format to an understandable and processable format, such as images to figures. Finally, the mobility management and it's five questions to determine whether a node is secure or not, offered by the framework is explained.

6. Case Study: Reduction of Energy Consumption of Two Sample Appliances

6.1. Introduction

The framework is built around the idea of reducing energy consumption in the household area while keep offering the same level of comfort, by applying different strategies and policies, such as delivering recommendations to replace appliances with more efficient ones based on the energy consumption or usage, or automatically adjusting operating periods of certain appliances based on predicted parameters. Moreover, the dynamicity, integrational nature of the framework opens avenues for an indirect impact on a number of industrial fields such as providing the local energy suppliers with the aggregated energy quantities consumed by households within a neighbourhood or city. Also, providing the appliances' manufacturers realistic energy consumption figures measured under real-life conditions.

This chapter consists of a detailed explanation of two main experiments designed to offer an example implementation of the proposed framework on two sample household appliances: the refrigerator and the immersion water heater. Both experiments will follow the same road map, which all begins with categorising the appliance in one of the three categories: uninterruptable, instant or schedulable, in order to allocate the appropriate applicable policy. For the first experiment related to the refrigerator, which belongs to the *Uninterruptable appliances* category, the *Energy Consumption based Appliance Substitution Policy (ECASP)* will be applied, however, the *Automatic Scheduling of Running Periods Policy (ASRPP)* will be applied for the second experiment related to the immersion heater.

In both experiments, data mining and predictions play an essential role to extend the overall dataset volume to become more precise approximate to the true decisions. Also, to offer an insight into the futuristic behaviour of appliances, so it is possible to act in advance. For this purpose, both MATLAB and Orange Data Mining will be used to follow the CRISP-DM approach, to apply several steps starting from understanding the business, then defining the data and preparing it in the way to reach high accuracy levels during the application of various data mining algorithms. The preparation phase will pass the data through several actions such as: removing the predictors with a high percentage of missing data, removing non-relevant data, detecting anomalies and outliers and dealing with them properly, then assessing correlations among predictors and target-wise. Once the first draft of the final data structure is set up, depending on the nature of the approaches, algorithms from both types of regression or classification will be considered for each experiment; where regression is chosen for the refrigerator experiment because there is a need to predict a continues value (the energy consumption), and the classification was picked up for the immersion heater experiment because it is required to predict (or classify) the discrete value *appliance on/off*. Important to mention that the final datasets were adjusted repeatedly after applying the algorithms to ensure reaching acceptable accuracy levels when applying a number of relevant evaluation metrics. The final step will be applying the correspondent policy to reduce the energy consumption and examine the obtained results which are presented and explained at the end of each experiment.

6.2. First Experiment: Reduction of Energy Consumption of the Refrigerator

The first experiment will be carried on the refrigerator, a sample appliance belonging to the uninterruptible household's appliances category, where the Energy Consumption-based Appliance Substitution Policy (ECASP) will be applied. As a first step data must be gathered, cleaned, then the prediction process starts to forecast the future energy consumption, which makes it possible to use external APIs to search and replace the appliance with a more efficient one, as will be explained in section 6.2.1. The initial data structure was presented in section 5.4.2.2. (Table 5.4) which shows the list of all potential independent variables that might be considered. The data has gone through a list of cleaning and preparation steps, as described in the next section.

6.2.1. Refrigerator Data Preparation

The following five steps were applied in this phase. Each step covers a different aspect of the data and aims to put the data into proper shape as a preparation for the following step related to choosing the appropriate modelling algorithm.

6.2.1.1. Removing Independent Variables with a High Percentage of Missing Values

As mentioned in section 5.4.3.1. the humidity dataset had a lot of missing values, so this independent variable has been excluded from the final data set to increase the overall prediction accuracy. Data were inspected for approximately three months is shown via the blue area, where the missing data records are represented by the gaps shown in Figure 6.1 which cover approximately over 30% of the measured data. In this figure, the internal humidity measured via a sensor is shown in blue, where gaps between the blue areas represent the periods when there is no data recorded.

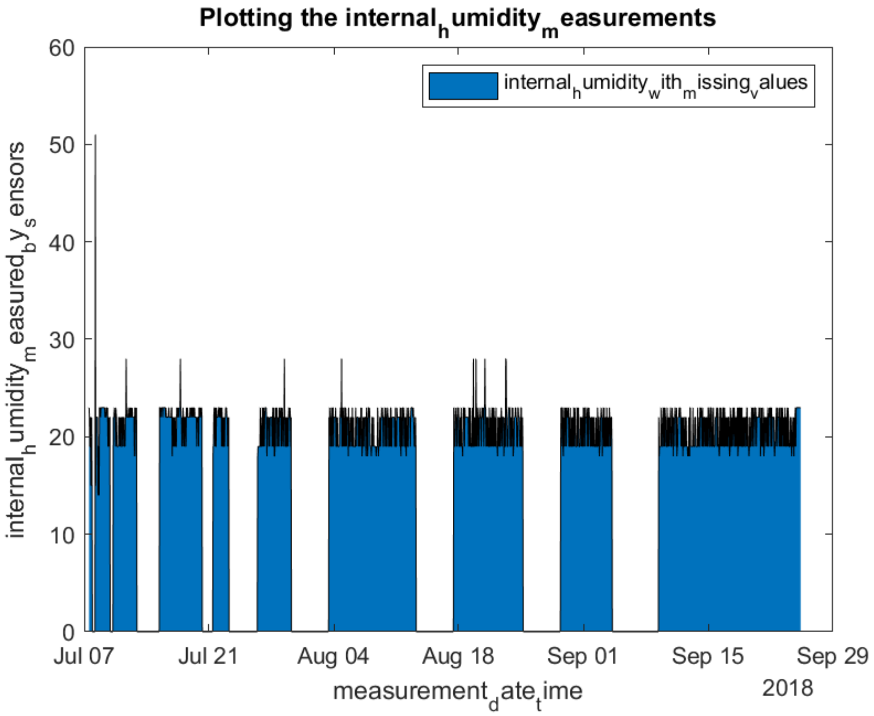


Figure 6.1: Internal humidity measured via a sensor, showing a huge number of missing values

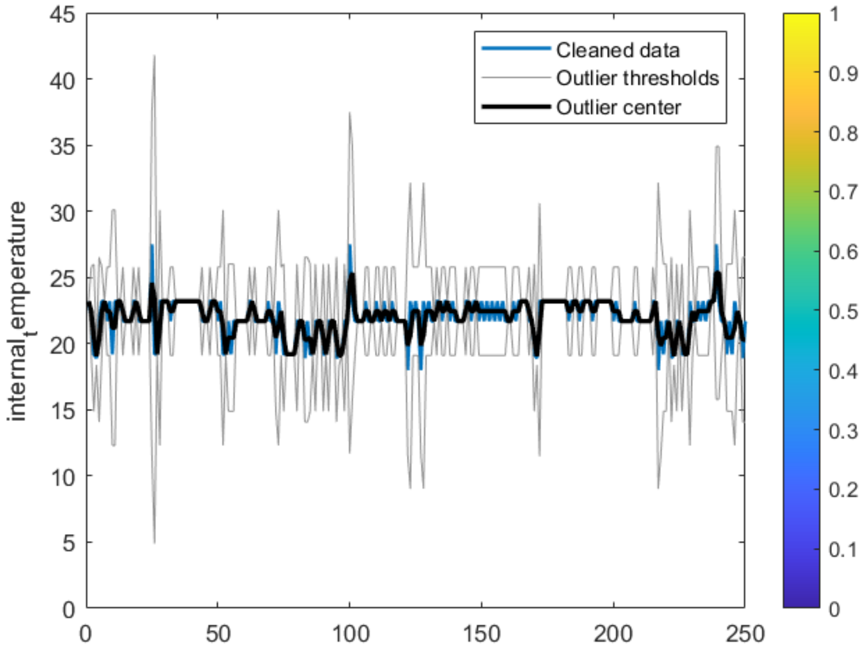
6.2.1.2. Removing of non-relevant Independent Variables

As will be seen later, the nature of the dependent target variable requires utilising regression algorithms, therefore some independent variables which are not relevant such as the DateTime should be removed. This kind of time-bounded variable could be relevant if the nature of the data mining is time series.

6.2.1.3. Anomalies, outliers' detection and smoothing data

The next step is reducing anomalies and outliers which exists in the dataset due to different reasons which were mentioned in section 5.4.2.1. Each feature is inspected and, when possible, cleaned by either removing the anomalies or smoothing the data peaks in the way to achieve an acceptable accuracy rate. Figures 6.2, 6.3 and 6.4 show three different plotting for four sample independent variables: *internal temperature*, *fullness*, and *times refrigerator door's open*. It is important to mention that only 250 records were considered to enhance the readability and clarity of the figures.

In Figure 6.2 we see in the top image, four three parameters: outliers threshold, outliers centre which show the boundaries of the data including the peaks (in grey), and its average (in black). The cleaned data line represents the resulting data after removing the outliers. The second bottom-left figure illustrates the data's local maxima and minima, which helps to identify the turning points where data flow increases or decreases. The third figure shows two lines representing the input data in light blue, and the resulting data after applying the smoothing method *Moving-mean* with the factor 0.5 (bold blue line).



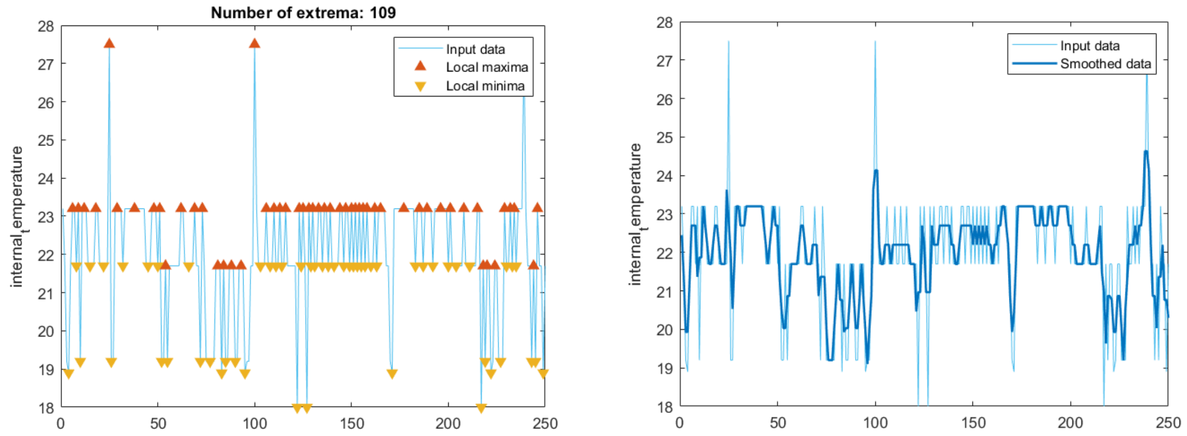
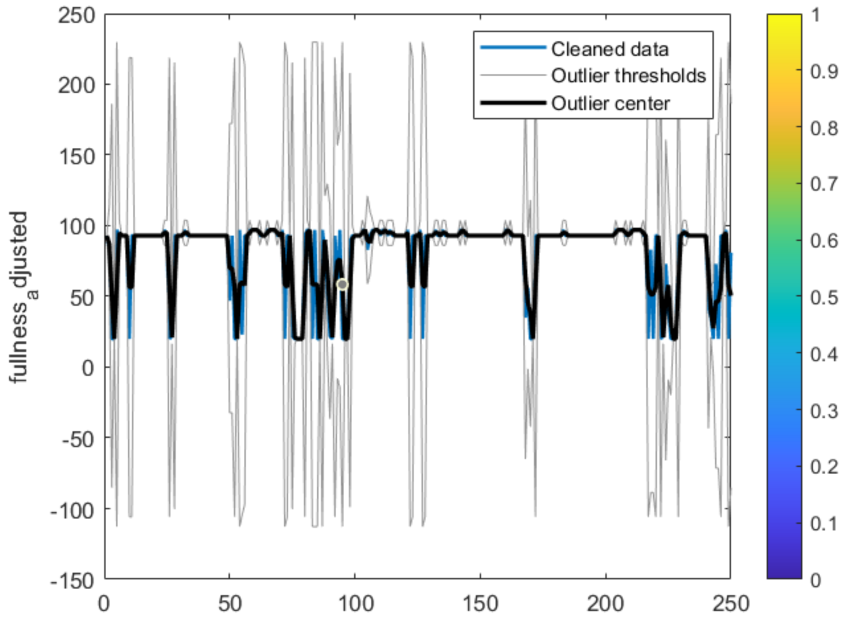


Figure 6.2: Internal Temperature Anomalies detection and data smoothing sample plotting for the first 250 data records

A similar approach can be seen in Figure 6.3 for the *Fullness* independent variable, which describes the percentage fullness of the refrigerator measured regularly. Due to the massive data disparity, seen in the bottom-left graph, a higher factor (0.7) was used for the data smoothing process.



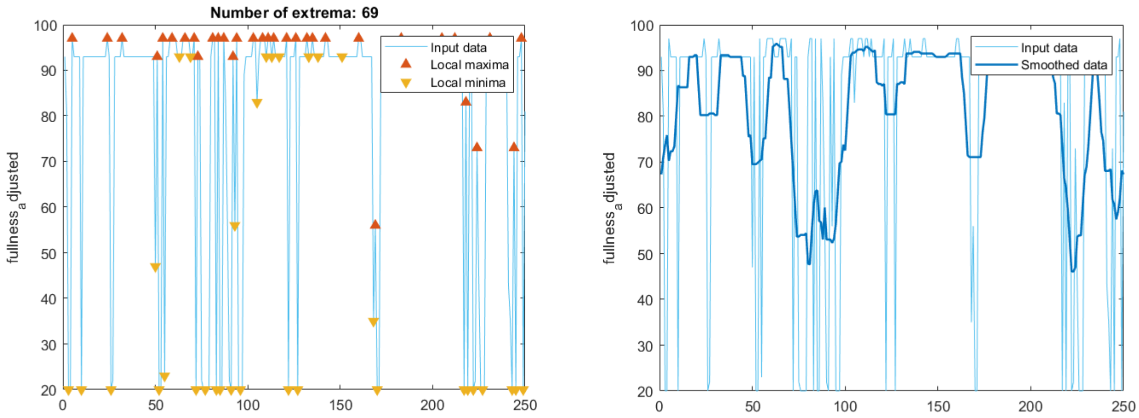


Figure 6.3: Three figures illustrate the anomalies detection, data smoothing for the refrigerator fullness independent variable

Figure 6.4 shows the same picture for the independent variable *times door is opened* which indicates how many times the refrigerator door was opened within 10 minutes.

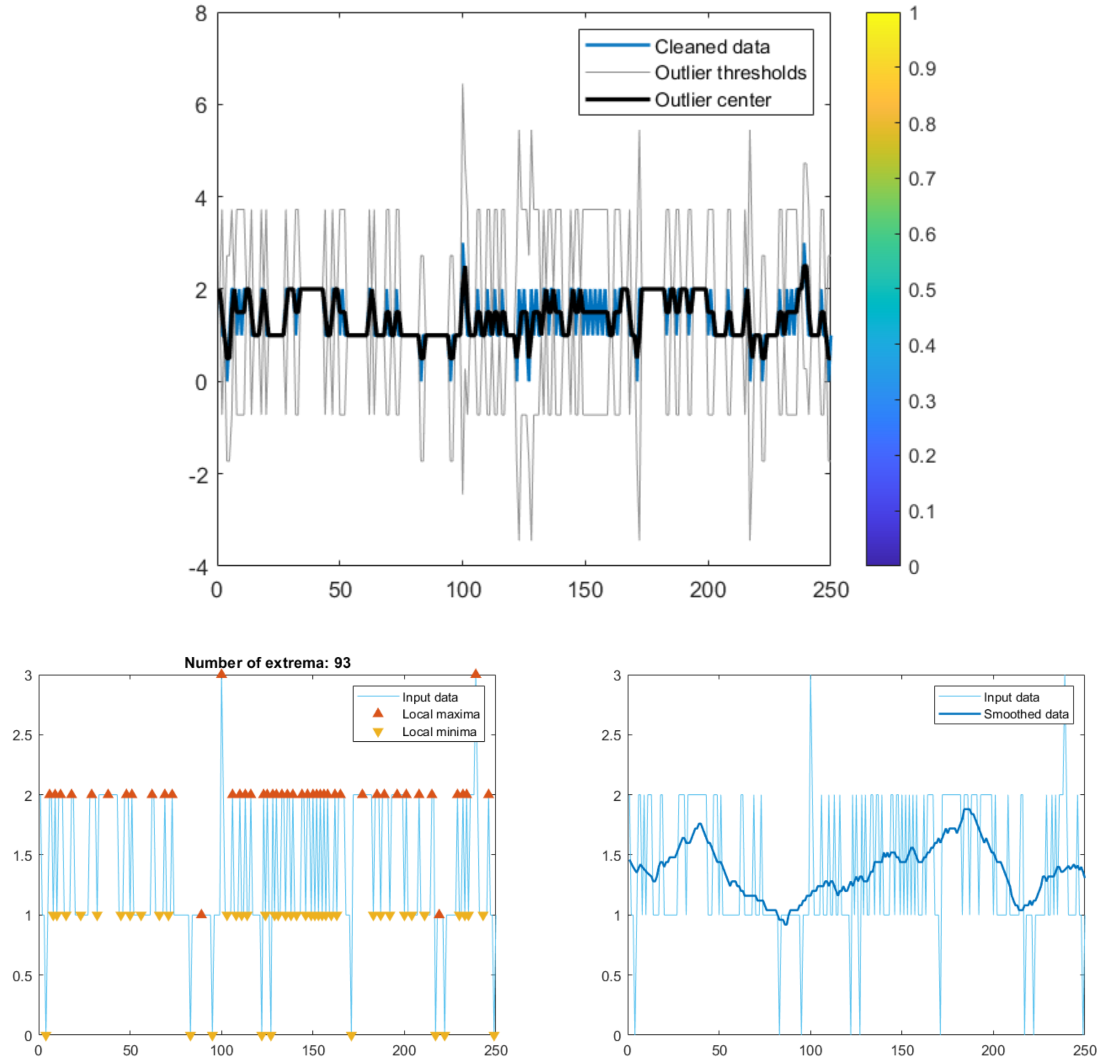


Figure 6.4: Plotting the independent variable times the refrigerator's door is opened

6.2.1.4. Correlation computation

As mentioned in section 5.4.3.4. both correlation types were examined, features showing high levels of feature-wise correlation were eliminated, and high levels of feature-target correlation were kept. This is calculated for several features using Spearman's method, as shown in Table 6.1.

Feature	**Correlation Type (*)**	**Feature 2 (**)**	**Correlation rate (Result)**
Internal Temperature	Feature-Target	Wattage	0.92 (Strong)
External Temperature 5cm	Feature-Target, Feature-Feature	Wattage, External Temperature Measured	0.09 (Weak), 0.87 (High)
External Temperature 2m	Feature-Target, Feature-Feature	Wattage, External Temperature Measured	0.15 (Weak), 0.9 (High)
External Temperature Measured	Feature-Target	Wattage	0.14 (Weak)
External Relative Humidity	Feature-Target, Feature-Feature	External Relative Humidity Measured	-0.07 (Weak), 0.86 (High)
External Relative Humidity Measured	Feature-Target	--	-0.06 (Weak)
Refrigerator Fullness	Feature-Target	--	0.85 (Strong)
Refrigerator Temperature	Feature-Target	--	0.92 (Strong)
Times Door Opened	Feature-Target	--	0.84 (Strong)
Duration Door Left Opened	Feature-Target	--	0.95 (Strong)
Occupants	Feature-Target	--	0.52 (Middle)

Table 6.1: Results of the calculated correlations for all relevant features/predictors [(*): Shows the type of the relation, either between a feature and the target (Feature-Target) or between 2 features (Feature-Wise), (**): Relevant only if the feature-wise correlation is examined

Table 6.1 contains a list of all relevant features or predictors for which the correlation calculation took place. Correlation type *Feature-Target* was calculated for the relevant feature against the target feature *Wattage* to determine whether to keep it or not, because, on one hand, showing a strong-correlation rate means that the examined feature is strongly affecting the target feature, on the other hand when a feature shows a weak-correlation rate with the target feature, the feature will be eliminated because it has not any considerable impact on the to-be-predicted target. Another correlation type is examined which is the *Feature-Feature*, where the correlation rate is examined for two features to assess the possibility to eliminate one of them to save processing power and storage. Features are eliminated when two features have a high *Feature-Feature* correlation rate. Features are divided into four categories:

1. **Temperature** – This category includes: 1) Internal temperature (InTp), 2) External temperature 5cm over the ground (XTp5c), 3) External temperature 2m over the ground (XTp5m) and 4) External temperature measured by a sensor (XTpM). Figure 6.5 illustrates the calculated correlation rates among features on one hand, and between the features and the target feature on the other hand. According to the figure, a high correlation is noticed between the target and internal temperature, and less correlation between target and external temperature, which supports the obvious fact that external temperature does not affect the energy consumption of a refrigerator.

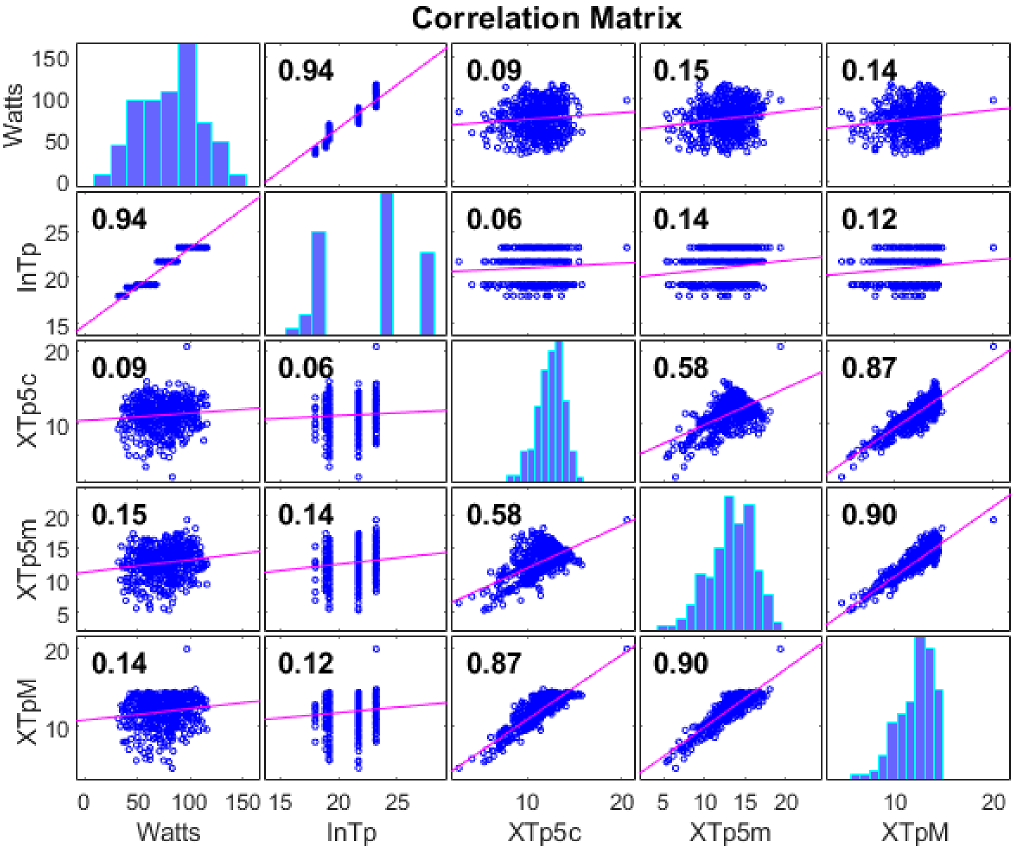


Figure 6.5: Correlation Matrix to illustrate the correlation rates among the external temperature features on one hand, and between features and target feature on the other hand.

2. **Humidity Measurements** – It includes both external humidity (HmdEx) and measured humidity by a sensor (HmdM), as shown in Figure 6.6, as expected - we see a weak correlation between both features and the target because there is no

direct relation between the wattage consumed by a refrigerator and the humidity figures. So both features should not be considered in the final dataset. Important to mention that the mutual correlation between both features is high (0.86) which means that it is possible to eliminate one of them.

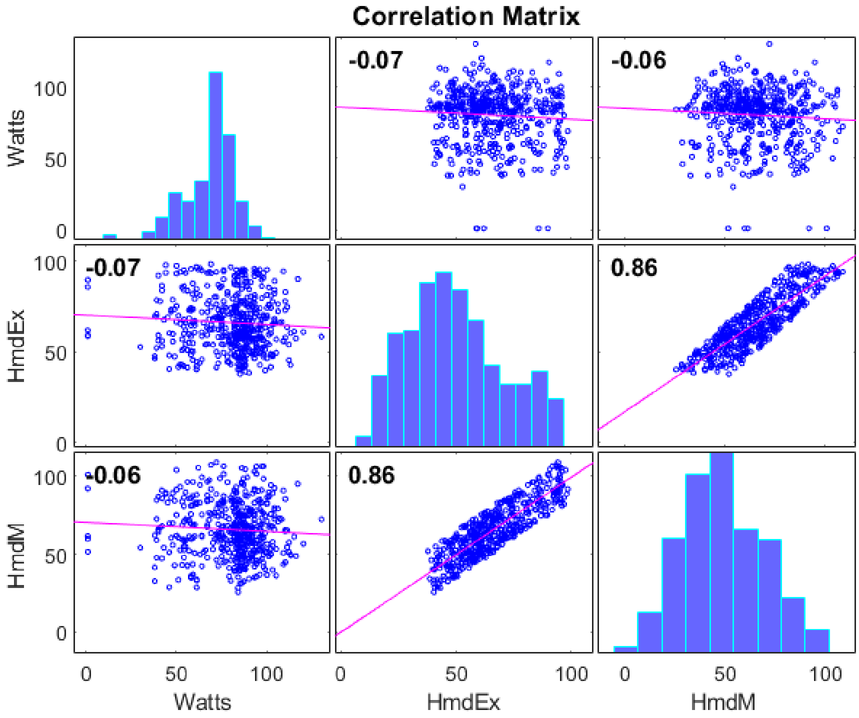


Figure 6.6: Plotting the mutual correlation rates among humidity features and the target feature.

3. **Refrigerator Parameters** – This category contains all independent variables that are directly measured from the refrigerator. It is expected that these features show a high correlation rate with energy consumption, which is proven by the resulting correlation rates shown in Figure 6.7. The calculated features are 1) Refrigerator fullness (Full), 2) Refrigerator target temperature (FrgTp), 3) Times the refrigerator door is opened every 10 minutes (TmDor), and finally 4) Seconds the door was left opened every time it gets opened (SecDo). Looking at the first line of the plot show that there is a strong correlation rate between the target feature *watts* and all other features; *fullness* (0.85), *target temperature* (0.92), *times door opened* (0.84) and *seconds the door left opened* (0.95).

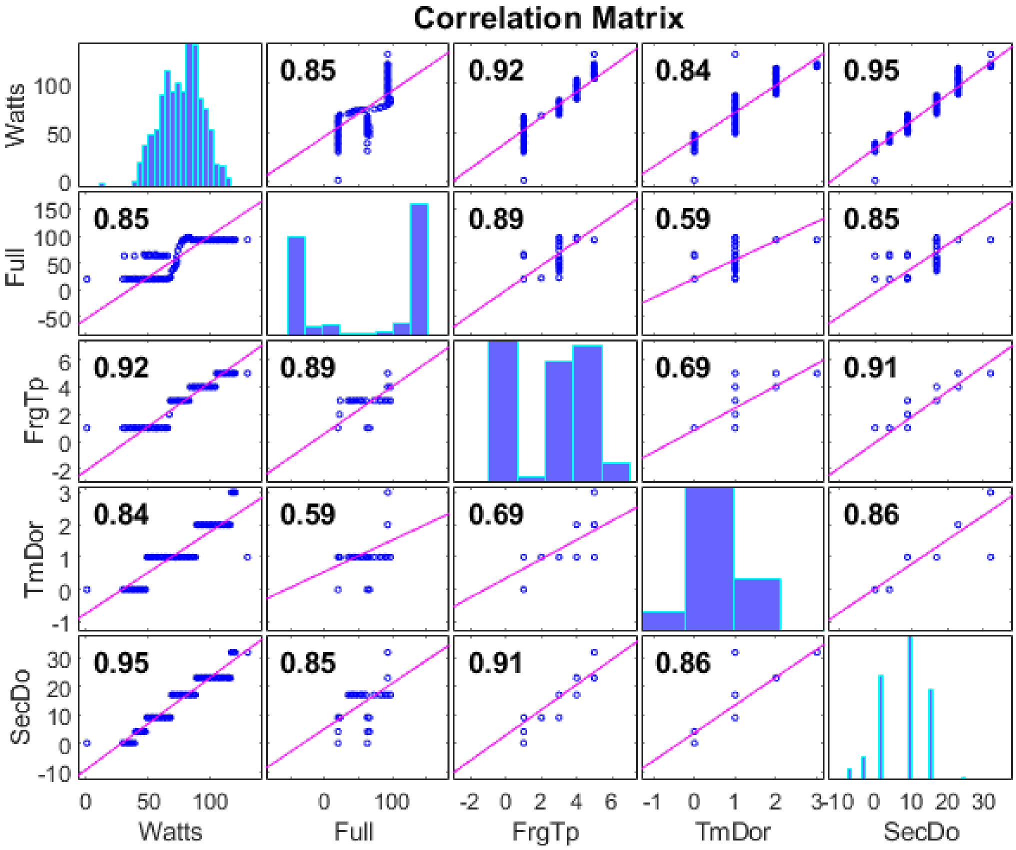


Figure 6.7: Illustration of features from the refrigerator parameters. fullness (Full), frig internal temperature (FrgTp), times the frig door is opened in 10 minutes (TmDor), seconds the door left open each time it gets opened (SecDo)

4. **Additional Parameters** – Figure 6.8 illustrates some additional parameters measured from the surrounding environment, these are the number of occupants who have been in the household (Occpt) and the type of the day (DayTp), which is divided into *working-day*, or *none-working-day* (such as weekend, holiday, bank holiday, etc.). Although the number of occupants does not show a high correlation rate with the target, it did score 0.52, it is considered because – logically – the number of household occupants must have a direct impact on the energy consumed by the refrigerator. Important to mention that a high mutual correlation rate between Occpt and DayTp was expected, however the rate was 0.00 because occupants number should increase in none-working-days, which was not the case in this experiment.

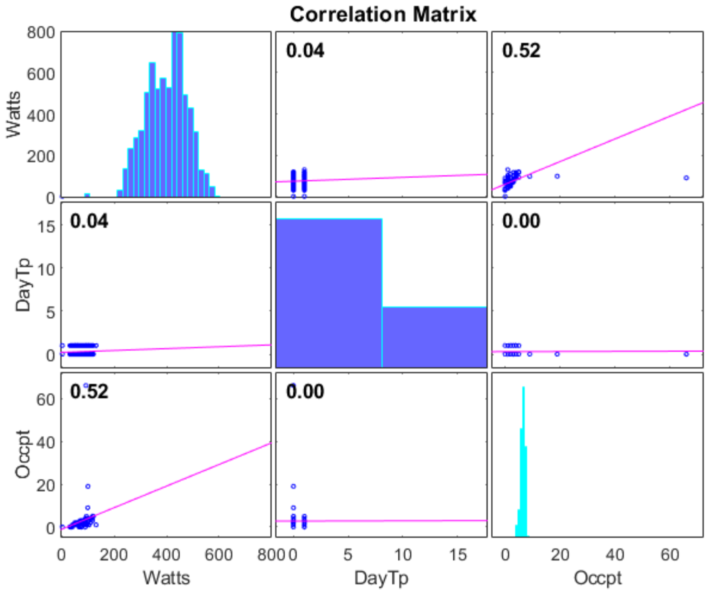


Figure 6.8: Correlation matrix of the additional parameters category

6.2.1.5. Variance calculation

The *weather-condition* is removed from the dataset because it has a very low variance score (0.24877371).

6.2.1.6. Data Final Structure

Table 6.2 shows the final structure after completing all previously explained steps during the preparation process.

Independent Variables	**Source**	**Field Type**	**Example values**	**Action / Justification**
Datetime	System	Datetime	2021-03-10 13:59:45	Removed / Not relevant for regression algorithms predictions
Internal Temperature	Sensor	Float	Expected range between +10 - +35 C°	Accepted / Shows high correlation with the target-feature
External Temperature 5cm	API	Float	The expected range between -15 and +40C°	Removed / Low target-correlation
External Temperature 2m	API	Float	The expected range between -20 and +45C°	Removed / See *External Temperature 5cm*
External Temperature Measured	Sensor	Float	The expected range between -20 and +45C°	Removed / See *External Temperature 5cm*
External Relative Humidity	API	Integer	0-100%	Removed / Low correlation score

External Relative Humidity Measured	Sensor	Integer	0-100%	Removed / due to low correlation score
Internal Relative Humidity Measured	Sensor	Integer	0-100%	Removed / A high percentage of missing data
Weather Condition	API	Varchar	rainy, windy, stormy, snowy, cloudy, sunny	Removed / Low Variance score (0.24877371)
Refrigerator Fullness	AI	Integer	0-100%	Accepted / It shows a high correlation score with the target-dependent feature
Occupants	AI	Integer		Accepted / The correlation score is not high, but also not low enough to discard this feature
Refrigerator Temperature	AI	Integer	Level 1 – 6	Accepted / A high correlation score was calculated.
Energy Consumption	Sensor	Float	kWh	Accepted / This is the target-dependent feature to be predicted measured in kWh.
Times Door Opened	Sensor	Integer		Accepted / Despite high feature-wise correlation with *Duration Door left Opened*, the field is kept to get better predictions
Duration Door Left Opened	Sensor	Integer	X seconds	Accepted / See *Times Door Opened*
Day Type	System	Varchar	Weekend, weekday	Altered / to (weekend: 0,1) because the effect of the day type affects the number of occupants staying at the household. The day type itself does not affect the prediction.

Table 6.2: Refrigerator's energy consumption prediction's final independent variables list after running the preparation phase

6.2.2. Modelling & Evaluation

To enhance the dataset of the refrigerator's energy consumption, it is required to predict the continuous dependent variable *wattage*, therefore, as discussed in section 3.3.7., regression is the most suitable approach for this purpose. As explained in section 5.4.4. a total of 5 different algorithms were examined and evaluated to decide the most suitable model. This is accomplished by applying the model to a portion of data to validate and test it. Figures 6.9 and 6.10 illustrate two examples *random forest* and *polynomial regression* of the resulting relationship between the test value represented by the blue bubbles, and the predicted value shown in yellow. The error between both values is represented with the orange line.

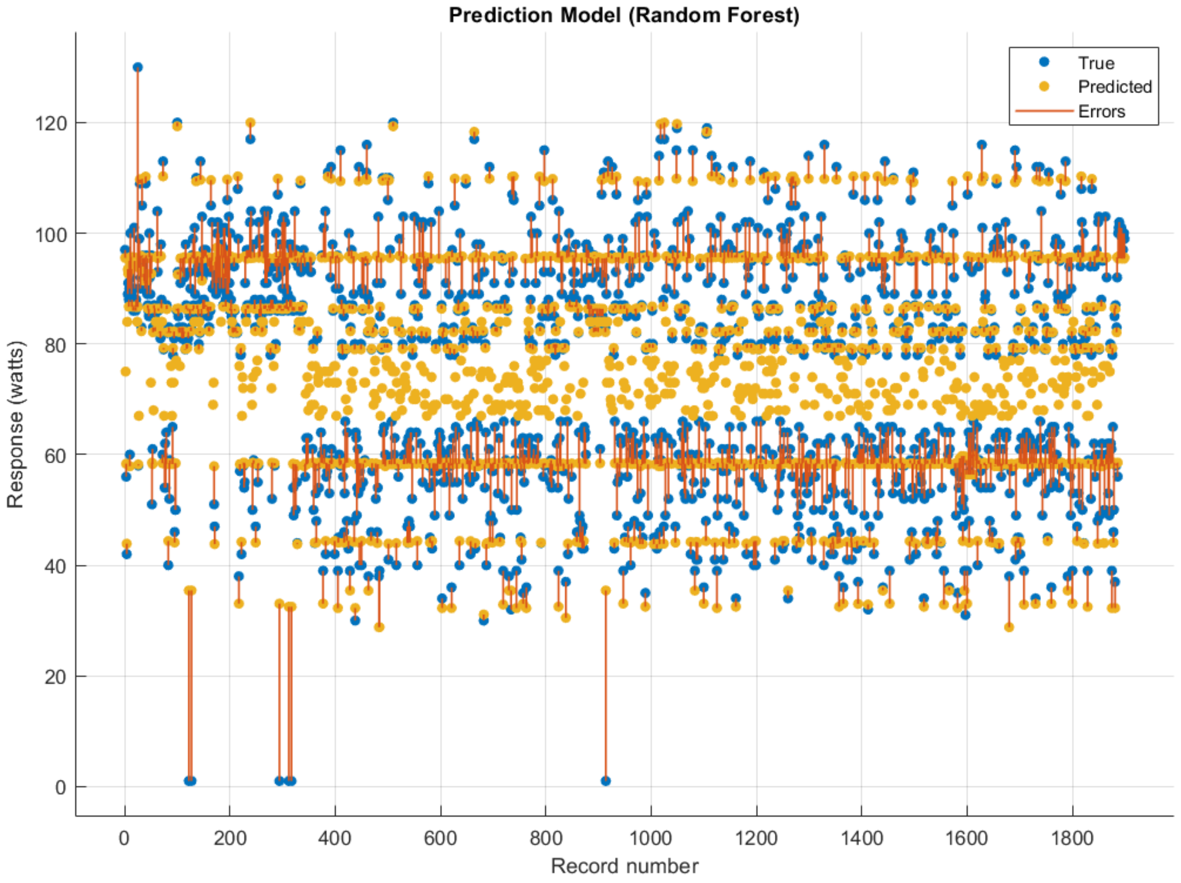


Figure 6.9: Prediction model obtained from applying the Random Forest algorithm, with a good RMSE (2.892)

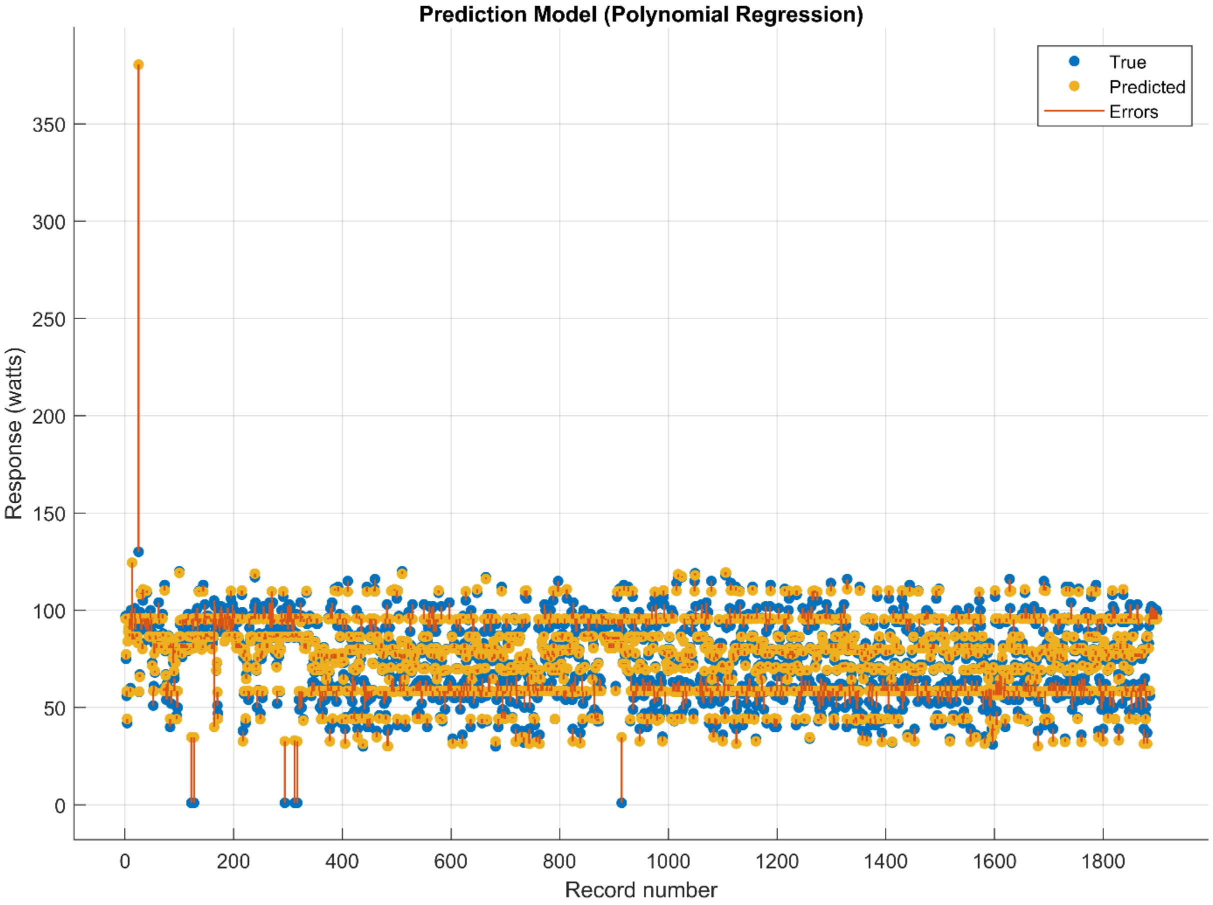
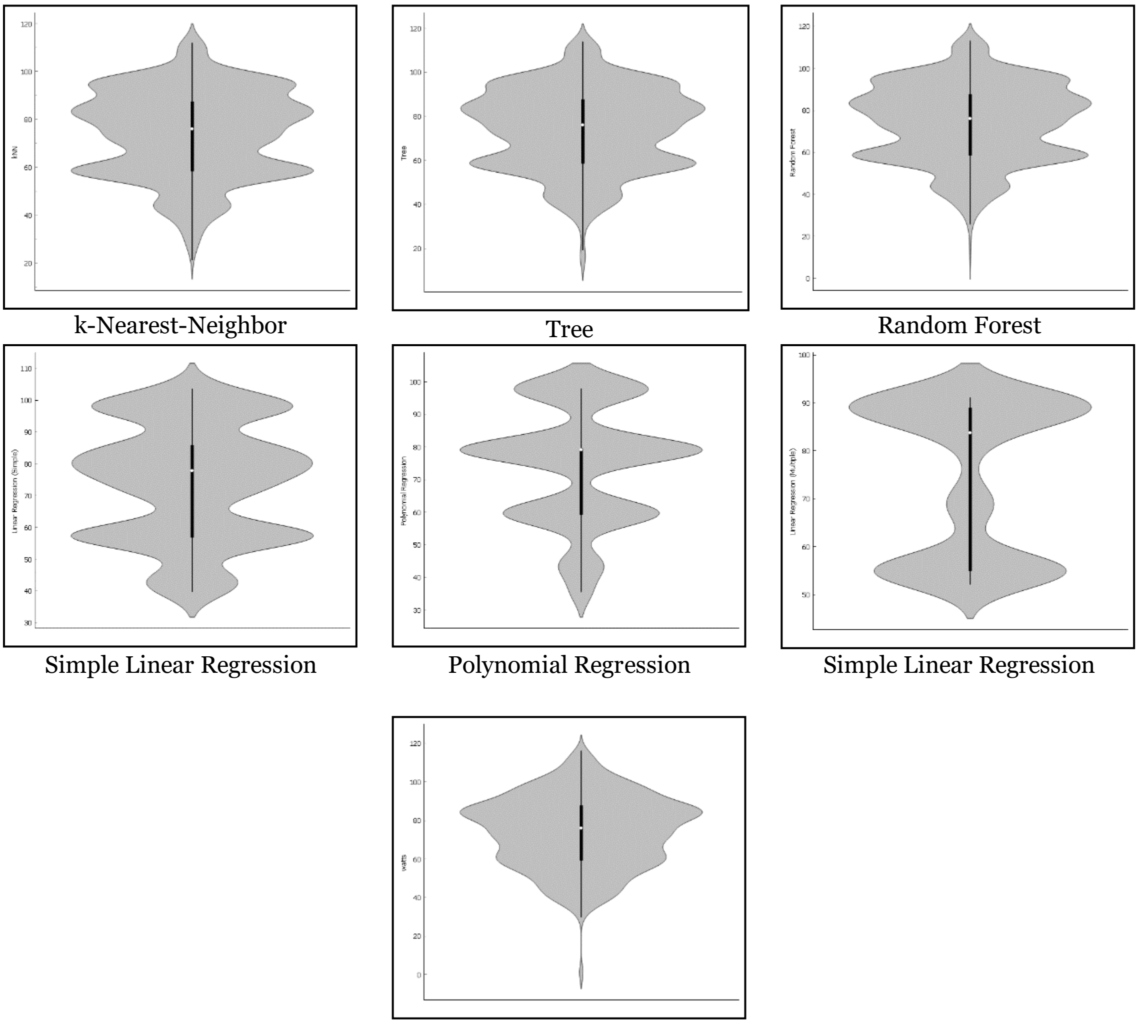


Figure 6.10: Prediction model obtained from applying the Polynomial Regression (RMSE 14.681)

Model	MSE	RMSE	MAE	R2
Polynomial Regression	215.542	14.681	9.871	0.397
Linear Regression (Multiple)	430.948	20.759	15.519	-0.205
Random Forest	8.363	2.892	1.708	0.977
Tree	8.332	2.886	1.685	0.977
Linear Regression (Simple)	22.513	4.745	3.359	0.937
k-Nearest Neighbour	13.519	3.677	2.254	0.962

Table 6.3: Different evaluation metrics related to the applied regression algorithms

A quick look at Table 6.3 reveals that Random Forest and Tree are performing well to predict the wattage. This can be clearly seen in the violin plots shown in Figure 6.11 simply by visually comparing the shape of the initial test wattage, and the predicted wattage done by Tree and Random Forest algorithms. Important to mention that kNN is doing well with an R2 value of (0.962), however it has been excluded because other models were more accurate.



k-Nearest-Neighbor

Tree

Random Forest

Simple Linear Regression

Polynomial Regression

Simple Linear Regression

Initial Wattage

Figure 6.11: Violin Plotting of the initial Wattage and the predicted one with different algorithms

The same fact is emphasized in Figure 6.12 and 6.13 that show the evolving MSE value during the data training process for both random forest and polynomial regression. In Figure 6.12 both estimated (light blue) and observed (dark blue) MSE are quite identical, however, in figure 6.13 those lines indicate a perceptible difference, where it is clearly seen that the polynomial regression algorithm has difficulties delivering a proper and reliable model.

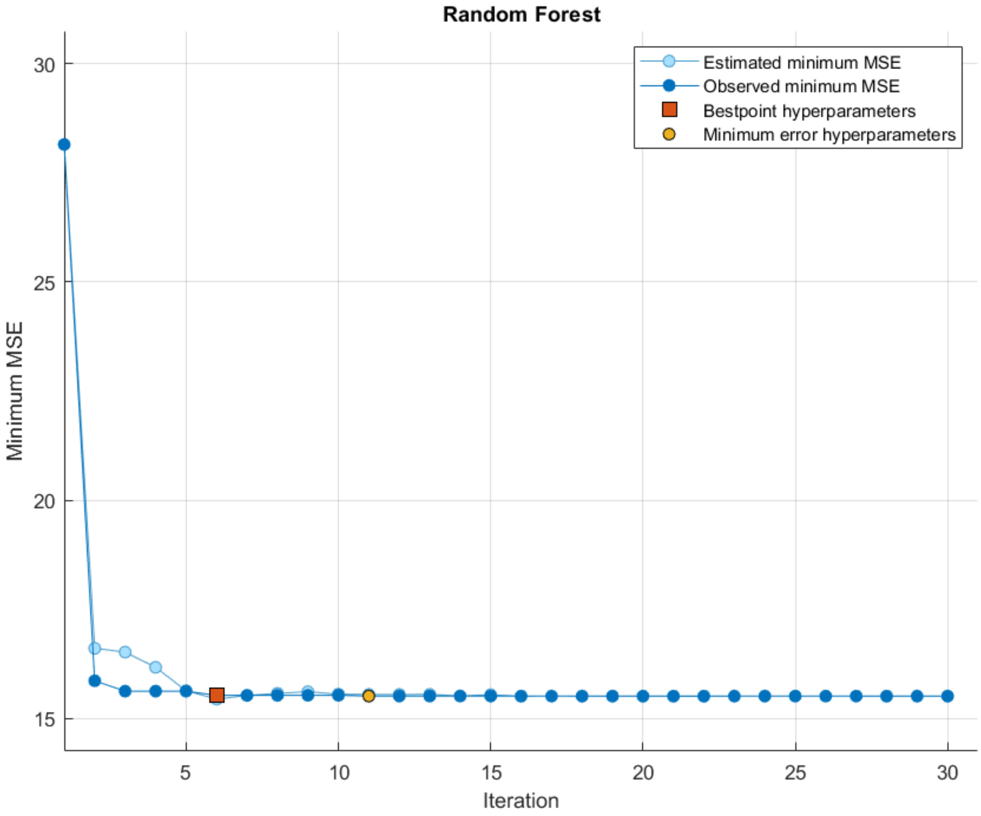


Figure 6.12: A graph showing the evolving of the MSE values during the data training process while applying the random forest algorithm

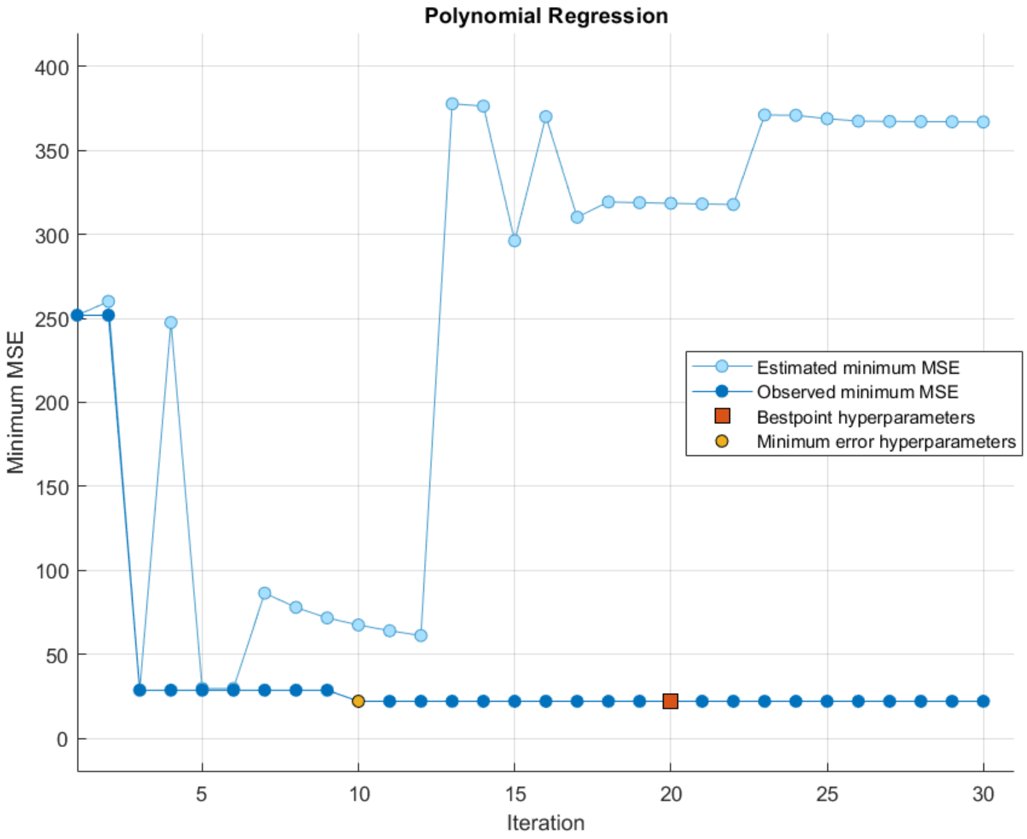
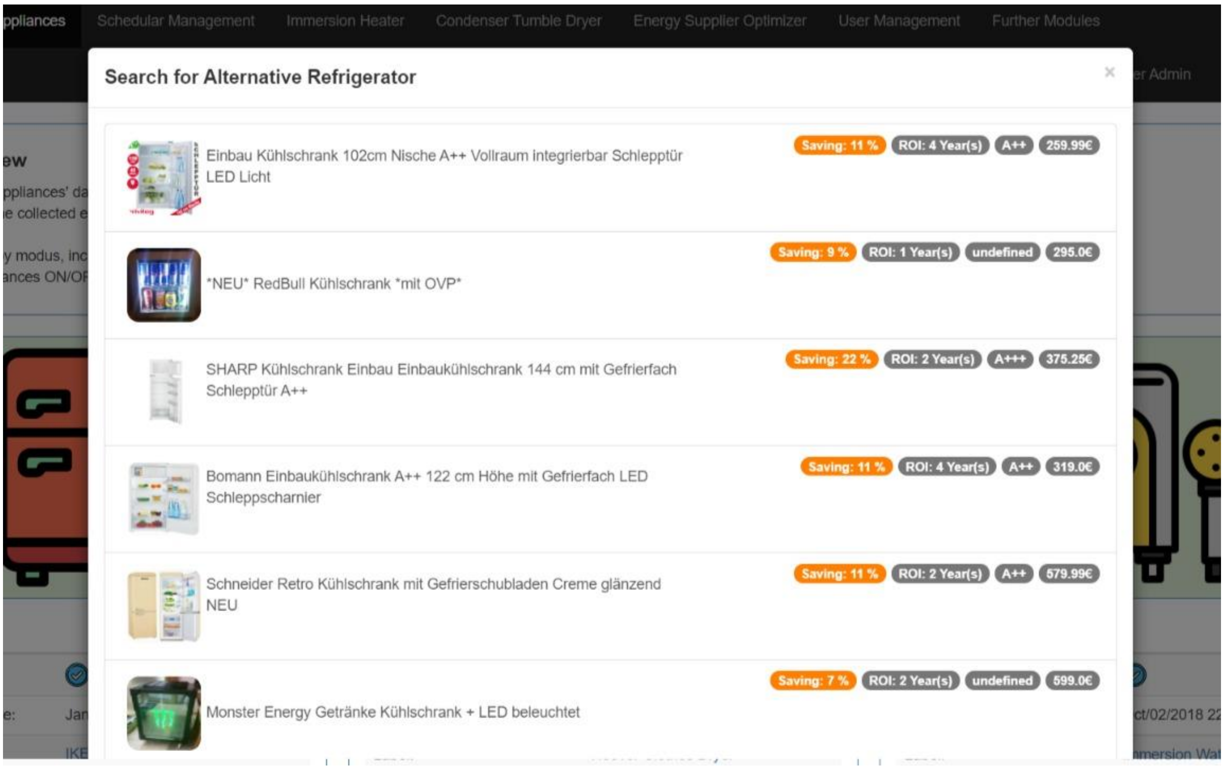


Figure 6.13: The MSE journey during the data training process emphasizes the fact the polynomial regression algorithm has difficulties to deliver a proper and reliable prediction model

In the next section, the dataset which is gathered from the observation and predictions will be utilised during the application of the energy consumption-based appliance substitution policy.

6.2.3. The Application of The Energy Consumption based Appliance Substitution Policy

The RECHS application, mentioned in section 5.5., implements the substitution strategy to reduce the consumed energy by using the dataset consisting of both historical and predicted wattage of the appliance to determine the exact and real-life energy consumption of the appliance, then it uses an external product API (from eBay Product Search API) to search for more efficient appliances while keep matching the same features. This approach can be initiated either manually, by starting the search by an admin, or automatically where an observation system keeps watching and measuring different parameters of the used appliance's energy consumption, compare it with results from the external API, then notify the admins with the suitable substitution recommendations. Following is an explanation of how an approximate saving of 22% of the energy consumption was achieved for the sample appliance.



Search for Alternative Refrigerator

Einbau Kühlschrank 102cm Nische A++ Vollraum integrierbar Schlepptür LED Licht — Saving: 11 % | ROI: 4 Year(s) | A++ | 259.99€

NEU RedBull Kühlschrank *mit OVP* — Saving: 9 % | ROI: 1 Year(s) | undefined | 295.0€

SHARP Kühlschrank Einbau Einbaukühlschrank 144 cm mit Gefrierfach Schlepptür A++ — Saving: 22 % | ROI: 2 Year(s) | A+++ | 375.25€

Bomann Einbaukühlschrank A++ 122 cm Höhe mit Gefrierfach LED Schleppscharnier — Saving: 11 % | ROI: 4 Year(s) | A++ | 319.0€

Schneider Retro Kühlschrank mit Gefrierschubladen Creme glänzend NEU — Saving: 11 % | ROI: 2 Year(s) | A++ | 579.99€

Monster Energy Getränke Kühlschrank + LED beleuchtet — Saving: 7 % | ROI: 2 Year(s) | undefined | 599.0€

Figure 6.14: Screenshot is taken from the RECHS application showing the returned matching list of equivalent refrigerators with more efficient energy consumption

Figure 6.14 shows part of the list of refrigerators with similar features to the sample refrigerator with better energy efficiency, which was returned by eBay product search API. The total average saving of each item in the result's list is calculated in the RECHS application using the three equations explained in section 5.2., and will be shown highlighted in orange within the list. Following is a sample efficiency calculation of the most efficient candidate *Sharp Refrigerator* shown in the third place on the list to replace the current refrigerator that proves that a refrigerator with the energy efficiency class A+++ may save energy up to 22%

$$Measured\ and\ predicted\ energy\ consumption\ of\ the\ current\ appliance = \ 0{,}081\ kWh$$

$$Measured\ and\ predicted\ energy\ consumption\ of\ the\ new\ appliance = \ 0{,}063\ kWh$$

$$Amount\ of\ \ hours\ the\ refrigerator\ runs\ daily\ in\ average = 8\ hours$$

$$E_1 = \ 0{,}081 * \ 8 = 0{,}648\ kWh$$

$$E_2 = \ 0{,}063 * \ 8 = 0{,}504\ kWh$$

$$Energy\ Saving = \frac{(0{,}648 * 365) - (0{,}504 * 365)}{0{,}648 * 365} * 100\% = 22{,}2\%$$

The next section will describe the second experiment related to the immersion heater and the applied policies to reduce energy consumption.

6.3. Second Experiment: Reduction of Energy Consumption of an Immersion Heater

The energy consumed when water is not used, considered wasted energy and should be cut off. For this purpose, the experiment should be built around predicting the periods when the appliance is used, in other words when the water flow rates are not zero, and cut-off the energy completely to prevent the re-heating cycles and thus save energy. As explained in section 5.4.2.2, several parameters will be measured and collected to assist in predicting the running periods of the heater (when the heater will be used). Having this valuable information allows switching the device ON shortly before using it, and OFF when it is not used, for instance at night.

Energy consumption (in kWh) and water usage (in Litres/second) are considered the most important variables which have a direct impact on the prediction. Figure 6.15 shows the measured average energy consumption over three months within 24-hours cycles. There the energy consumption represented in the blue line, shows a peak in consumption between 16:00 and 21:00 clock, which is the period when households come back home and start their evening activities (cooking, dishwashing, …). The same behaviour can be seen around mid-day between 11:00 and 14:00 clock, and in the early morning between 05:00 and 7:00.

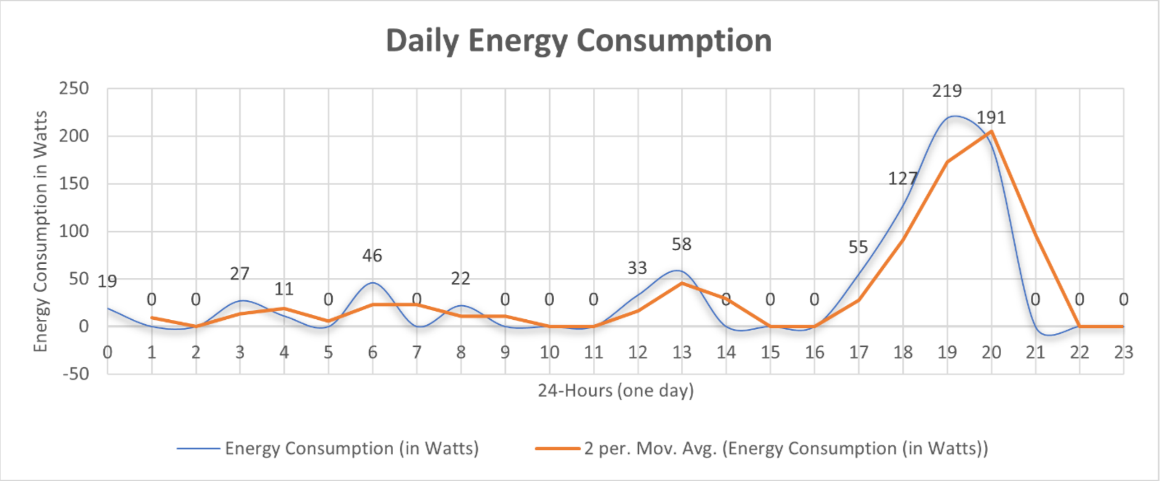


Figure 6.15: Measured Average Energy Consumption in 24-hours (in kWh), showing the moving average line (in orange)

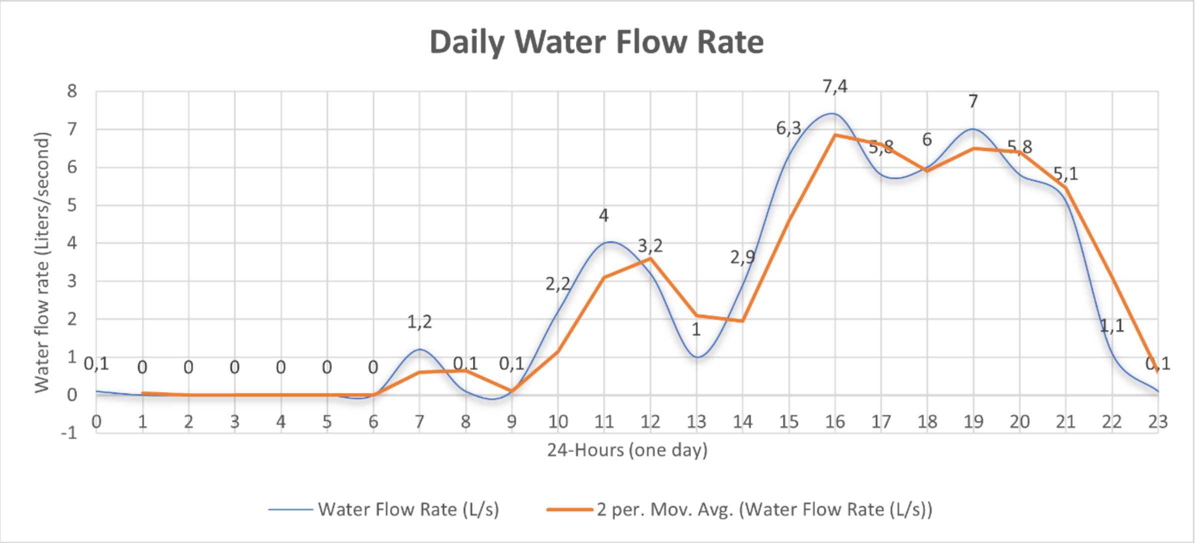


Figure 6.16: Measured Water Flow Rate in 24-hours (in Litre/second) with the orange line showing the moving average

Comparing both figures is shown in the usage and energy consumption indicator graph in Figure 6.17, where the red line represents energy consumption, and the blue line represents the water flow. Important to mention that these lines do not show the quantity, rather shows *one* when there is a value, and *zero* when there is no value. it emphasizes the obvious idea that both peaks in the energy consumption and water usage are match in the periods between 6:00 and 23:00 clock, taking into consideration that the heater volume of 10 litres plays a role in buffering hot water so there is no need for continuous heating up the water. In other words -roughly- the energy consumption increases when the water flow increases. However, according to Figure 6.17, although the water usage is zero in the time between midnight and 6:00 clock in the morning, there are noticeable energy consumption values (or peaks). This occurs because the device attempts to keep the water on the same required temperature level when the water temperature drops.

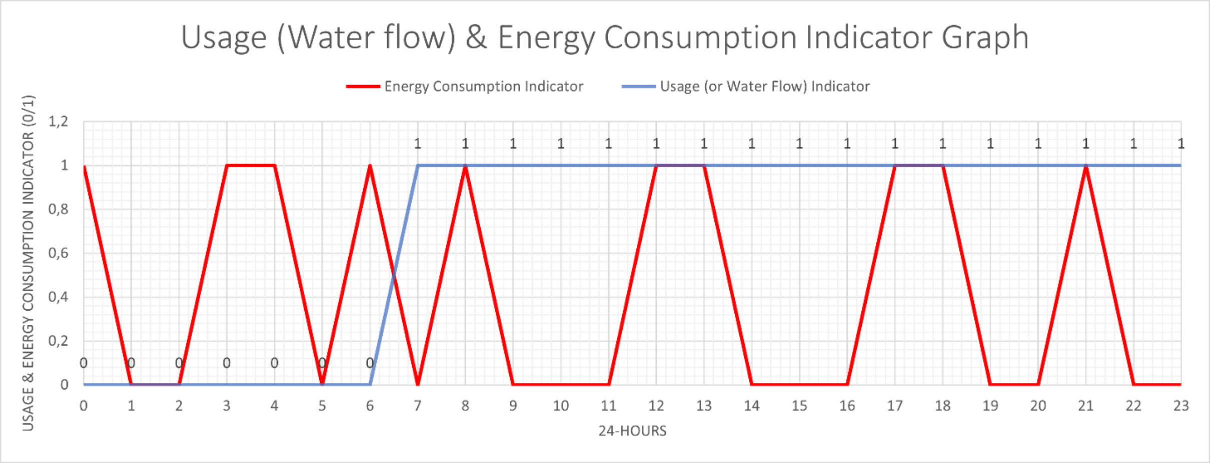


Figure 6.17: A comparison between the energy consumption (in kWh) and the Water Flow Rate or usage (in a litre(s)/sec.)

Similar to the previous experiment, it all begins with defining and preparing the data, then applying it to the appropriate algorithms to predict the running periods to allow applying the ASRPP policy. The initial data structure which was previously discussed in section 5.4.2.2. in Table 5.5, went through a number of verification and cleaning processes explained in the next sections.

6.3.1. Immersion Heater Data Preparation

To reduce the processing power and time, and to achieve high accuracy rates, data was prepared to decide for the most relevant and suitable independent variables, moreover, the records were running through several cleaning procedures to eliminate data noise, inconsistency, and irrelevant data records. These iterations are described as follows:

6.3.1.1. Shared Independent Variables

Because both sample appliances used in both experiments operate within the same environment, they share a number of similar independent variables. Following are a list of these variables which were prepared and discussed in the first experiment related to the refrigerator. These includes:

- **Internal humidity** – removed because of the missing values as explained in section 6.2.1.1.

- **Timestamp** – Automatically generated date/time to define the time sequence which will be used in the applied sequence classification with recurrent neural network data mining technique.
- **Internal temperature** – An important feature for both refrigerator and water immersion heater, because it has a direct impact on the loss of heat occurred due to lack of adequate isolation. As seen in section 6.2.1.3. and Figure 6.2, anomalies and outliers detection were applied to smooth the data and increase prediction's accuracy.
- **External temperature and external humidity measurements** – As explained in the correlation calculations in section 6.2.1.4., all external temperature and external humidity independent variables are removed because they show low correlation rates with the target feature (Feature-Target), and high correlation with other features (Feature-Wise)
- **Surrounding additional independent variables from the household** – Number of occupants and the type of day (weekend/weekday) are common variables that are valid for all appliances in both experiments.

6.3.1.2. Detecting Outliers' Detection and Smoothing Data

Similar to the approach applied in the first experiment, explained in section 6.2.1.3., anomalies detection is one of the basic data preparation processes in this experiment which aims to remove all outliers and clean the data from extreme values. Although this process has been applied to all variables, in this section only one example will be explained to avoid repetition. Figure 6.18 shows three graphs, the top one indicates outliers (represented with letter *x*) in the dataset of the variable *water flow rate*. The final cleaned data is represented in the bold blue line. Both bottom graphs also explain the process of finding the extreme points (bottom-left) and the smoothed data by applying the *moving mean* method (bottom-right).

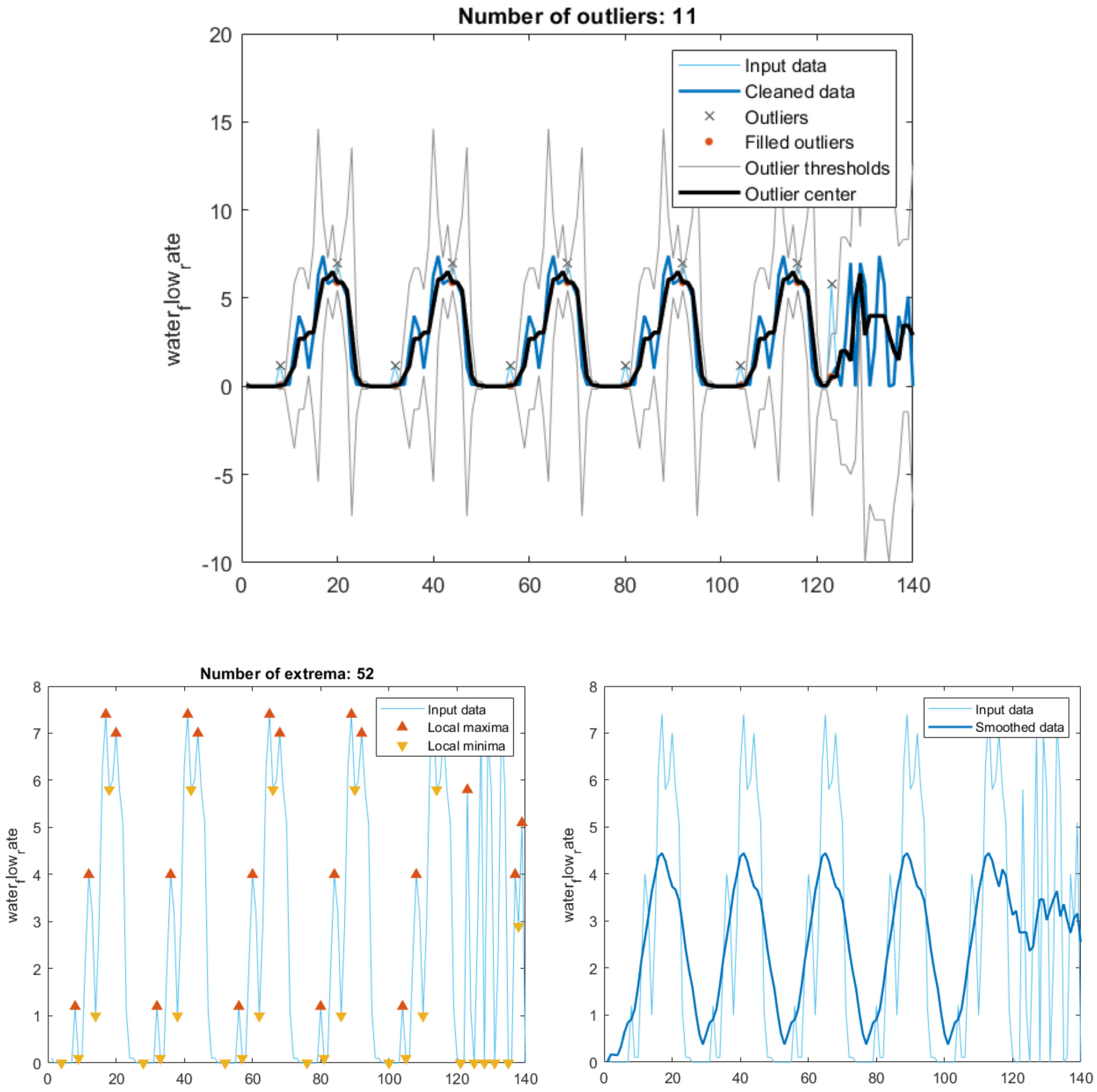


Figure 6.18: Three figures show the anomalies detection for the variable Water Flow Rate

6.3.1.3. Merging, Splitting, Aggregating Data

Sometimes it is necessary to merge several variables into one average variable to enhance the data correctness, and to reduce the calculation time. This approach is applied to both external temperature, humidity and traffic delay fields, as follows:

- ➔ **External Temperature** – Both fields *External Temperature 5cm over the ground* and *External Temperature 2 meters over the ground* are merged into *External Temperature* to enhance the accuracy and the overall temperature.

- **External Humidity** – The same approach is done for the external humidity, where the average of both fields *External Relative Humidity* and *External Relative Humidity Measured* is put into one field to combine both data resources and ultimately improve the data correctness.

- **Traffic-Related Variables** – Consists of two variables; firstly, *Traffic delay in minutes* which determines when the first person will arrive home, so the heater is turned on shortly before he arrives. All other persons who may arrive later, will not be considered because the appliance has already been turned on. This merged field's value is based on *Traffic delay in minutes for the first occupant*, *Traffic delay in minutes for the second occupant* and *Traffic delay in minutes for the third occupant*. Secondly, the *Traffic situation* shows the traffic situation for the first person arrives home.

6.3.1.4. Correlation Assessment

This step concentrates on assessing both correlation types: Feature-Wise, and Feature-Target. Features show high levels of feature-wise correlation will be eliminated, and high levels of feature-target correlation will be considered in the prediction analysis. This is calculated for several features using Spearman's method, as shown in table 6.4, where the column *Correlation Type* shows the type of the relation, either between a feature and the target (Feature-Target), or between 2 features (Feature-Wise), and the column *Feature 2* which is only relevant if the feature-wise correlation is examined and shows the correspondent's variable name. To avoid plotting big size figures, which may decrease the readability, similar variables are put together as follows:

1. **Temperatures** – The correlation assessed for the following fields: internal temperature (InTp), external temperature measured 5cm over the ground (XTp5c), external temperature measured 2 meters over the ground (XTp2m), external temperature measured by a sensor (XTpM), and incoming water temperature (WTp) against the target variable (to be predicted) the water flow, or the usage of the appliance (WFlow). Results are illustrated in Figure 6.19 that clearly emphasizes the idea that temperature has barely influenced the usage of the device, which is seen on the correlation values, on line one, that are around the zero between (WFlow) and all other variables. However, there is a strong

correlation among all external temperature measurements (for example between (XTpM) and (XTp5c) it reached 0.96)

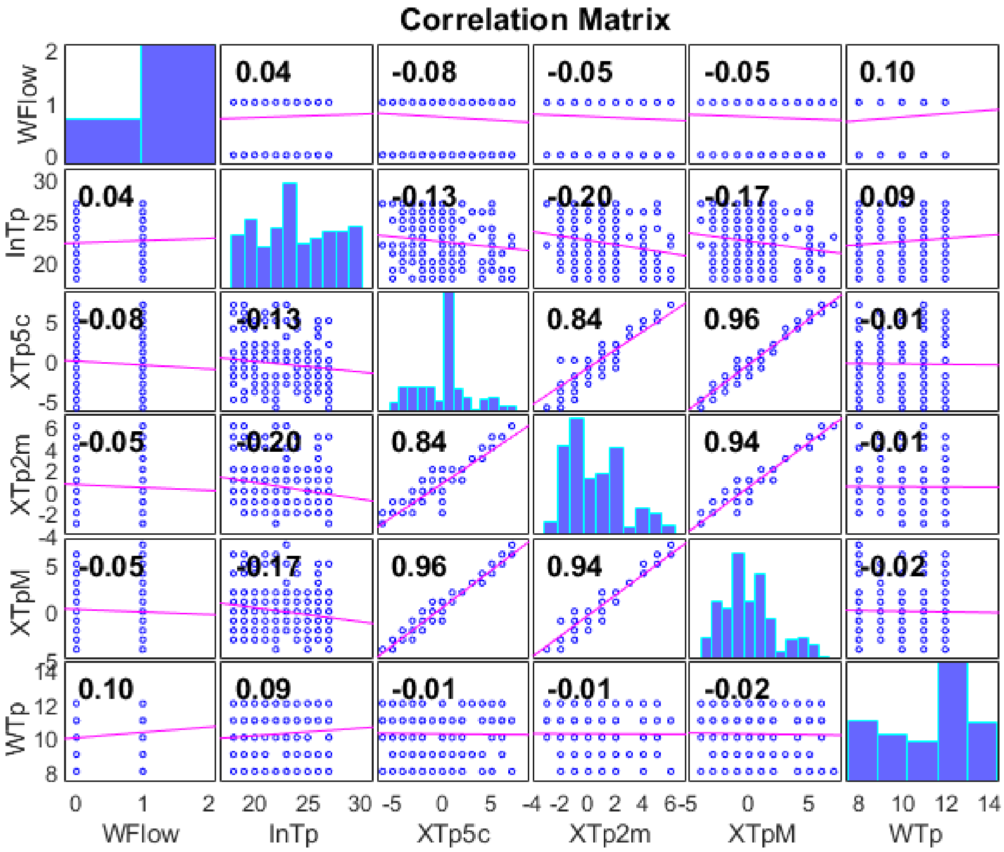


Figure 6.19: Correlation assessment of the target and all temperature-related variables, and among all temperature variables

2. **Humidity** – Calculating the correlation for humidity variables reveals a similar picture as done for temperatures as illustrated in Figure 6.20. Where there is almost no correlation between the target and the humidity, however, there is a strong correlation between humidity measured by a sensor and the humidity retrieved from the API which is 0.84

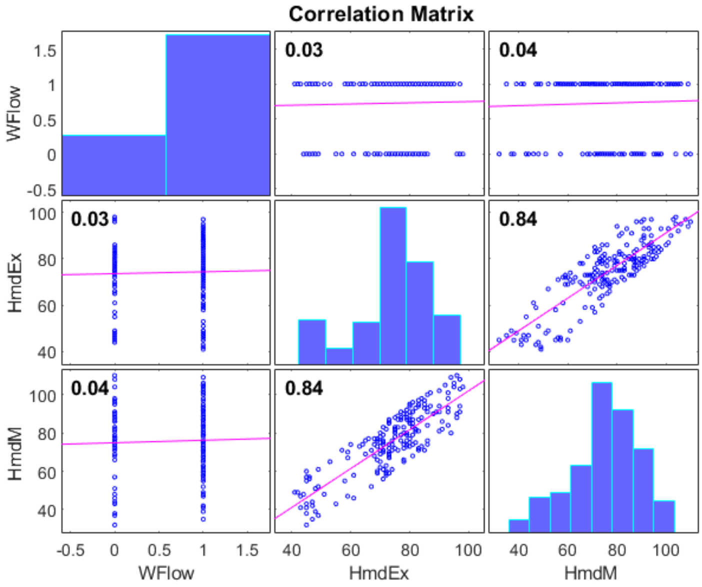


Figure 6.20: Correlation assessment for humidity related variables together with the target variable The usage of the appliance, and among themselves

3. **Heater Related Parameters** – In this part all variables measured directly from the heater will be assessed. This includes energy consumption (EngCo), appliance on/off (OnOff), and water consumption rates (WFlRt), together with the water flow on/off (WFlow). Except for the water consumption variable, it is not expected that all of these variables will have a strong correlation with the target variable *appliance usage*. Results are illustrated in Figure 6.21, which shows a relatively strong correlation with water consumption rates (0.55) and a weak relationship with the energy consumption quantity (0.25)

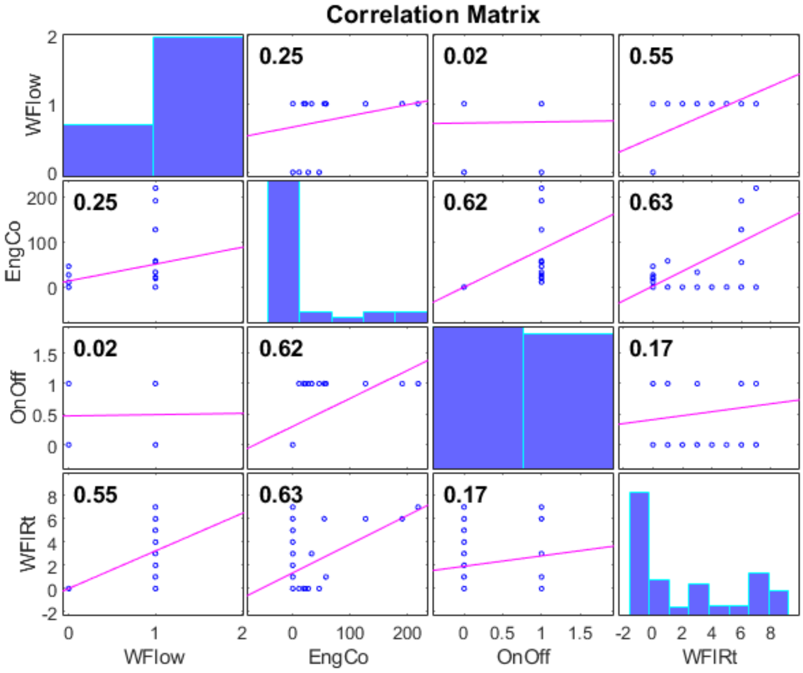


Figure 6.21: Illustration of heater features

4. **Others** – All other surrounding variables are examined here. These are occupants (Occpt), traffic delay in minutes (TrfDm), and weekend/weekdays (DyTyp). According to Figure 6.22, a weak correlation was noticed between the appliance usage target variable (WFlow) and both traffic delay and the type of the day (DyTyp), however there is a strong correlation with the number of occupants reached 0.96. This means when the number of occupants increases, the usage of the appliance increases as well.

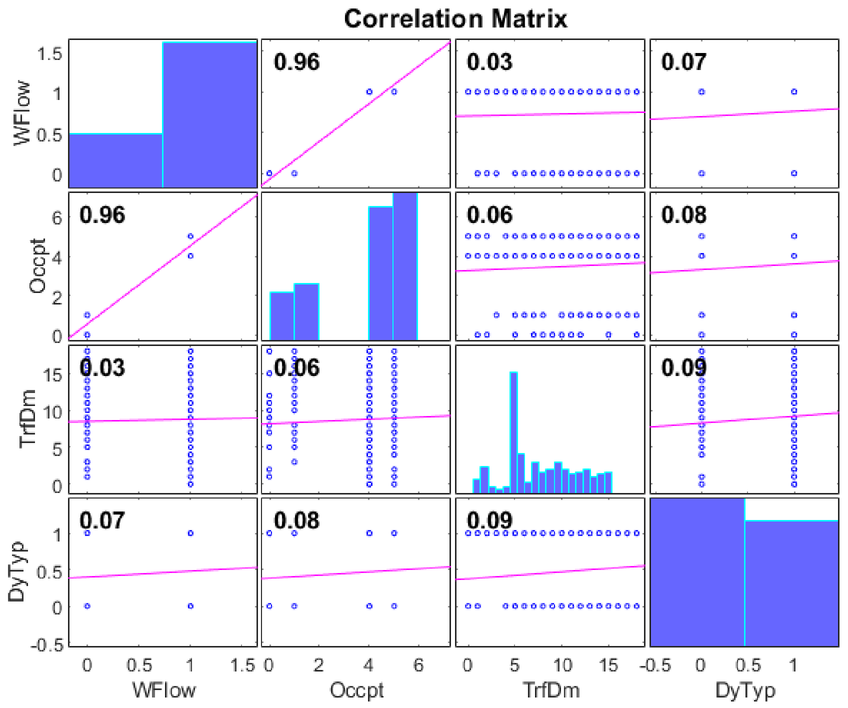


Figure 6.22: Plotting the correlation between surrounding variables and the target variable.

All calculated correlation factors are summarized in the following table 6.4.

Feature	Correlation Type (*)	Feature 2 (**)	Correlation rate (Result)
Internal Temperature	Feature-Target	Water flow (True/False)	0.04 (weak)
	Feature-Feature	All other temperature vars.	-0.2 - 0.09 (weak)
External Temperature 5cm	Feature-Target	Water flow (True/False)	-0.08 (weak)
	Feature-Feature	All other temperature vars.	-0.01 – 0.96
External Temperature 2m	Feature-Target	Water flow (True/False)	-0.05 (weak)
	Feature-Feature	All other temperature vars.	-0.01 – 0.94
External Temperature Measured	Feature-Target	Water flow (True/False)	-0.05 (weak)
	Feature-Feature	All other temperature vars.	-0.02 – 0.94

Incoming Water Temperature	Feature-Target	Water flow (True/False)	0.10 (weak)
	Feature-Feature	All other temperature vars.	-0.01–0.1 (weak)
External Relative Humidity	Feature-Target	Water flow (True/False)	0.03 (weak)
	Feature-Feature	Measured humidity	0.84 (strong)
External Relative Humidity Measured	Feature-Target	Water flow (True/False)	0.04 (weak)
	Feature-Feature	Retrieved humidity value	0.84 (strong)
Energy Consumption	Feature-Target	Water flow (True/False)	0.25 (weak)
	Feature-Feature	Heater related vars.	~0.6 (strong)
Appliance On/Off	Feature-Target	Water flow (True/False)	0.02 (weak)
	Feature-Feature	Heater related vars.	0.17 – 0.62
Water Consumption Rates	Feature-Target	Water flow (True/False)	0.55 (rel. strong)
	Feature-Feature	Heater related vars.	0.17 – 0.63
Day Type (weekends, holidays, annual leave, …)	Feature-Target	Water flow (True/False)	0.07 (weak)
	Feature-Feature	Additional variables	~0.08 (weak)
Occupants	Feature-Target	Water flow (True/False)	0.96 (strong)
	Feature-Feature	Additional variables	~0.07 (weak)
Traffic Delay in minutes	Feature-Target	Water flow (True/False)	0.03 (weak)
	Feature-Feature	Additional variables	~0.07 (weak)

Table 6.4: Results of the correlation assessment for all relevant features/predictors linked to the immersion heater experiment

6.3.1.5. Variances Analysis

Both *weather condition* and the *heater target temperature* were removed from the dataset because they have a very low variance score; 0.25 for the first, and zero for the second.

6.3.1.6. Data Final Structure

Table 6.5 shows the immersion heater's energy consumption prediction's final variables list after running the preparation phase.

Variables	Source	Field Type	Example values	Action
Timestamp	System	Datetime	2021-03-10 13:59:45	Automatically generated
Heater Target Temperature	AI(*)	Integer	Varies between 1˚ - 85˚	Accepted
Occupants	AI(*)	Integer		Accepted
Internal Temperature	Sensor	Float	Ranges between +10 - +35 C˚	Accepted

External Temperature 5cm over ground	API/Sensor	Integer	Ranges between -20 and +45C˚	Removed
External Temperature 2m over ground	API/Sensor	Integer	Ranges between -20 and +45C˚	Removed
Wind speed	API	Integer		Removed
Incoming water temperature	Sensor	Integer	Ranges between +0 and +45C˚	Removed
Weather condition	API	Varchar	rainy, windy, snowy, sunny, …	Removed
External Humidity	API	Integer	0-100%	Removed
Internal Humidity	Sensor	Integer	0-100%	Removed. A high percentage of missing data. Explained in section 6.2.1.1.
Off day (weekends, holidays, annual leave, …)	System	Varchar	Yes, no	Accepted. High correlation
Energy consumption	Sensor	Float	kWh	Accepted. High correlation with other variables related to the heater
Appliance On/Off	DM(**)	Binary	1/0	Removed
Water Consumption	Sensor	Integer	Litre(s)/sec.	Accepted. Shows relatively high correlation with the target
Traffic situation for the first occupant	API	Varchar	*on-time, slightly-delayed* or *late*	Merged with *Traffic situation for 2. occupant* and *Traffic situation for 3. occupant*
Traffic delay in minutes for the second occupant	API/System	Integer	Minutes	Merged with *Traffic Delay in Minutes for 2. occupant* and *Traffic Delay in Minutes for 3. Occupant*
Traffic situation for the second occupant	API	Varchar	*on-time, slightly-delayed* or *late*	Merged. See *Traffic situation for 1. Occupant*
Traffic Delay in minutes for the second occupant	API/System	Integer	Minutes	Merged. See *Traffic Delay in Minutes for 1. Occupant*
Traffic situation for the third occupant	API	Varchar	*on-time, slightly-delayed* or *late*	Merged. See *Traffic situation for 1. occupant*

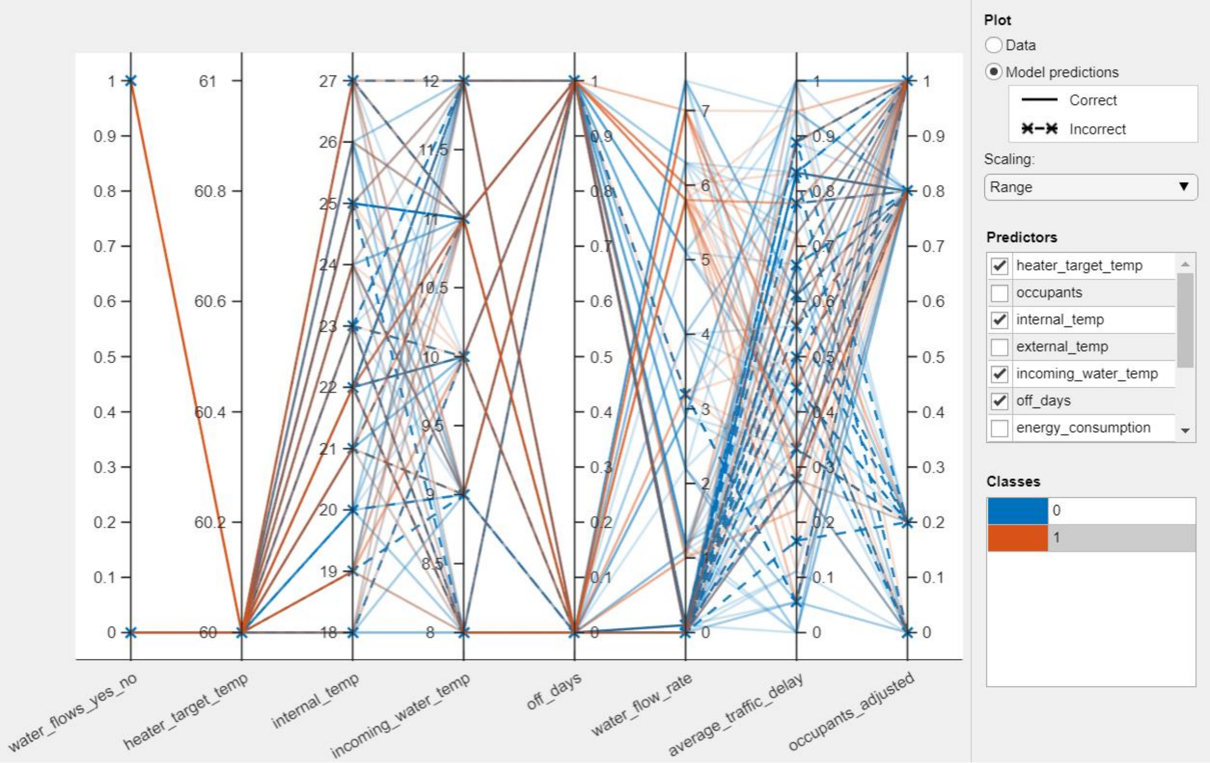
Traffic delay in minutes for the third occupant	API/System	Integer	Minutes	Merged. See *Traffic Delay in Minutes for 1. Occupant*

Table 6.5: Immersion heater's energy consumption final variables list. (*) AI: Artificial Intelligence, (**) Data Mining

6.3.2. Modelling & Evaluation

As mentioned before in section 5.4.4., it is required to predict the time when the immersion heater is used, this means -technically- predicting when the water flows out of the heater. Since the target variable is a label (true/false) that is combined with a time sequence, the sequence classification with recurrent neural network data mining techniques will be applied.

As shown in Figure 6.23, the MATLAB Parallel Coordinates Plot Tool is used to decide the most relevant predictors. It offers the possibility to investigate involved predictors and their influence on the overall prediction process, and determine the most suitable predictors, and eliminate the ones with less impact. In this figure dotted lines represent the mistaken prediction, where straight lines show the correct forecasting.



Plot
○ Data
● Model predictions

Scaling: Range

Predictors
☑ heater_target_temp
□ occupants
☑ internal_temp
□ external_temp
☑ incoming_water_temp
☑ off_days
□ energy_consumption

Figure 6.23: MATLAB Parallel Coordinates Plot Tool used to investigate predictors and their influence on the prediction

As shown in Figure 6.24, all begins with plotting the heaters running cycles, where the first 200 cycles are shown. The Y-Axes represents the appliance's ON and OFF status, by 0 and 1, where X-Axes shows the first 200 sequence or cycles.

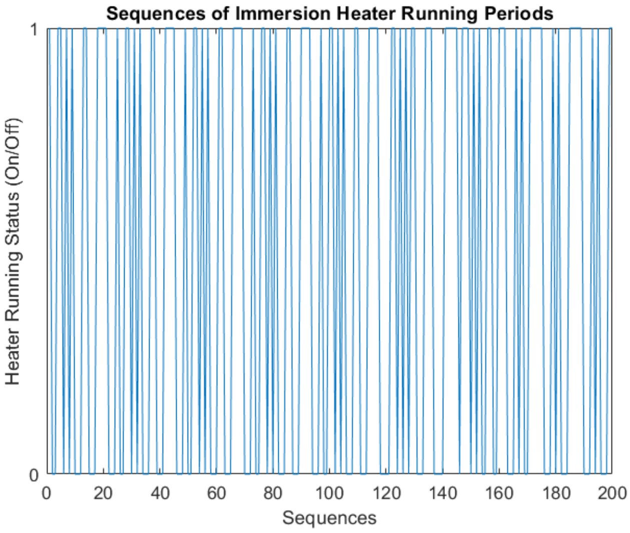


Figure 6.24: Plotting the sequence (in time) of the on/off cycles

The previously chosen predictors are used to forecast the heater's running cycles by feeding them to MATLAB. Figure 6.25 shows the training process for 250 iterations. Both RMSE and Loos values are calculated during the training, three peaks are noticed in the beginning, however, both values are getting closer to zero towards the end of the 250th iteration. This reveals that the training process was reliable and can be used for the forecasting illustrated in Figure 6.26.

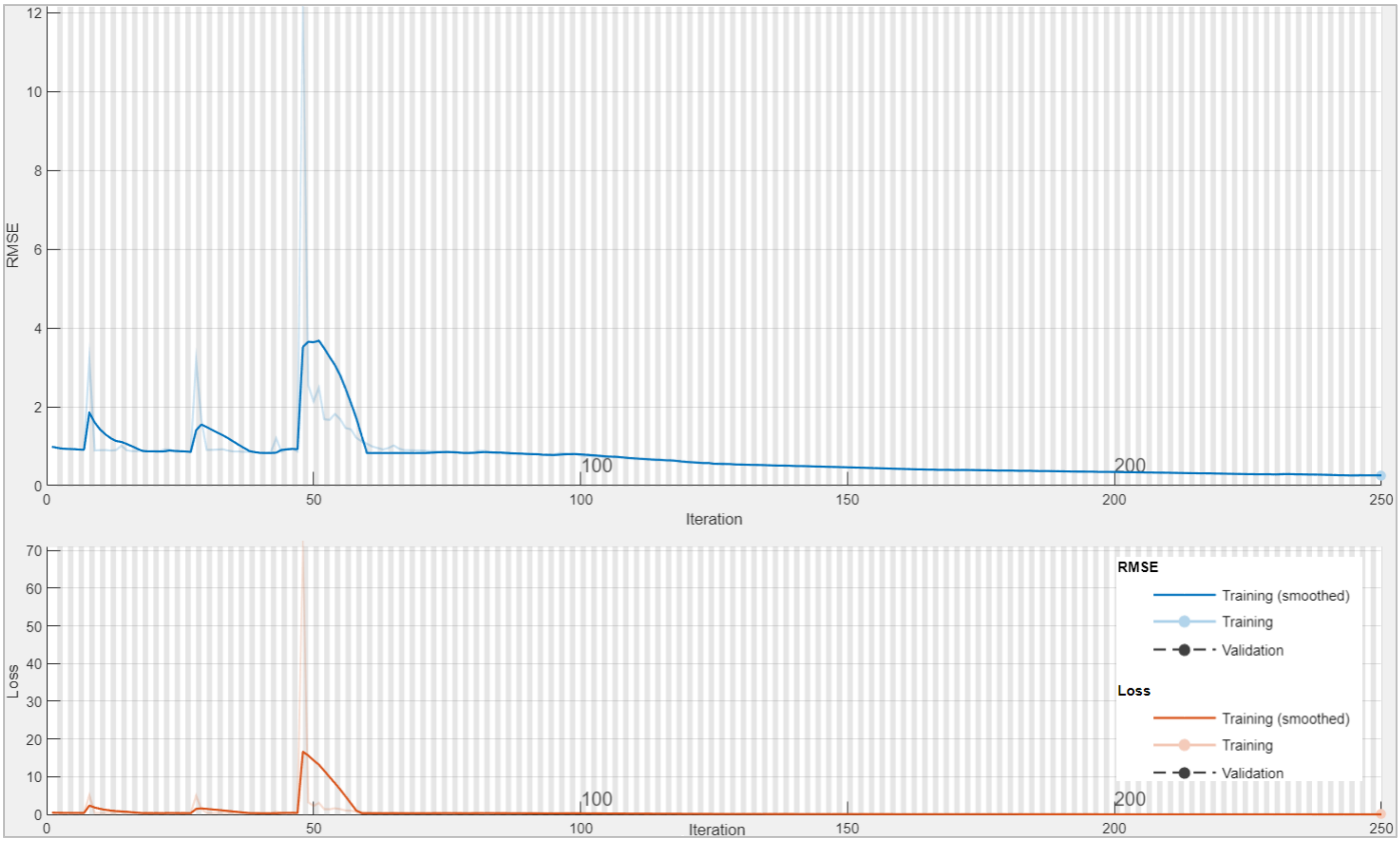


Figure 6.25: Training Process and the evolvement of both RMSE and Loss values

Results can be seen in Figure 6.26, where the forecast of the last 20 sequences is shown in red, and observed data is represented in blue. The forecast will be used to operate the immersion heater in the future.

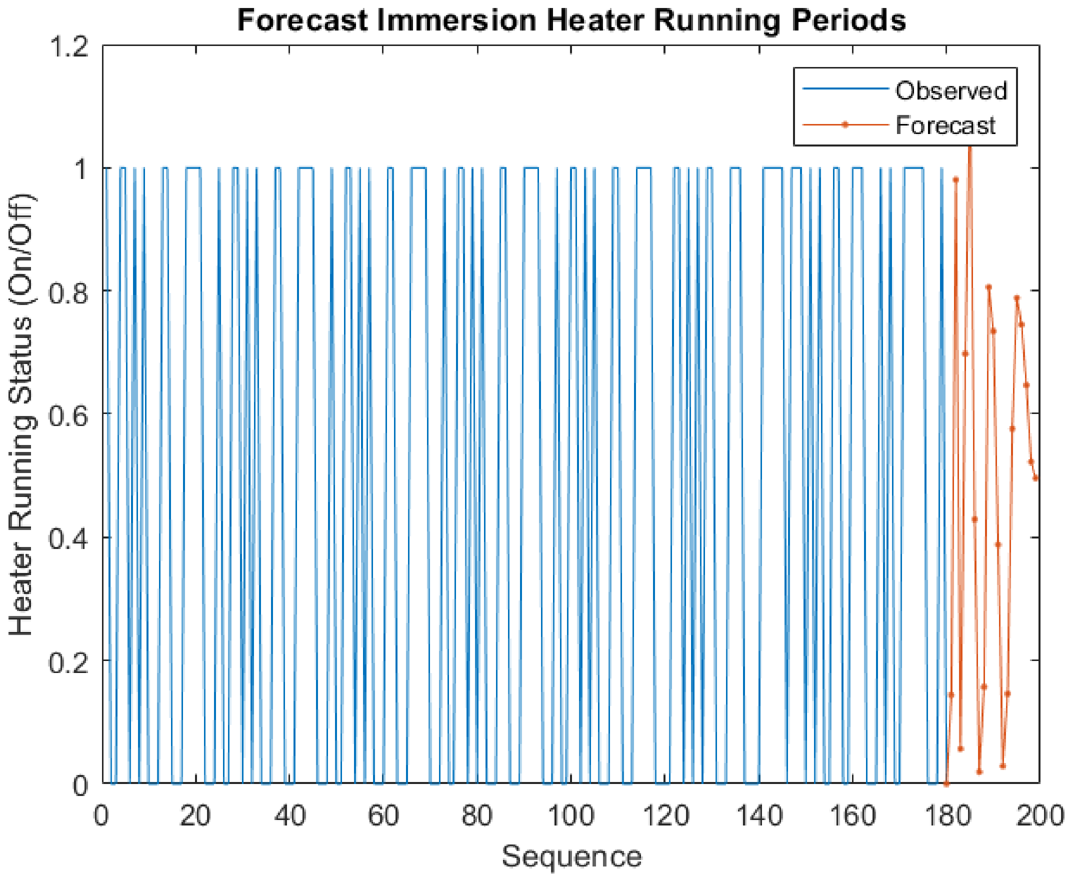


Figure 6.26: Combined plot shows the observed and forecast of the heater's running periods

The exact prediction of the last 20 time sequences and the correspondent RMSE error is shown in Figure 6.27, where both graphs indicate bad results for the first two cycles, then it improves starting from the third cycle till the 7th one where it shows incorrect prediction. This analysis is reflected in the bottom graph where a good prediction is seen when the error closes to zero, and the bad predictions are seen when the error is far from the bottom line.

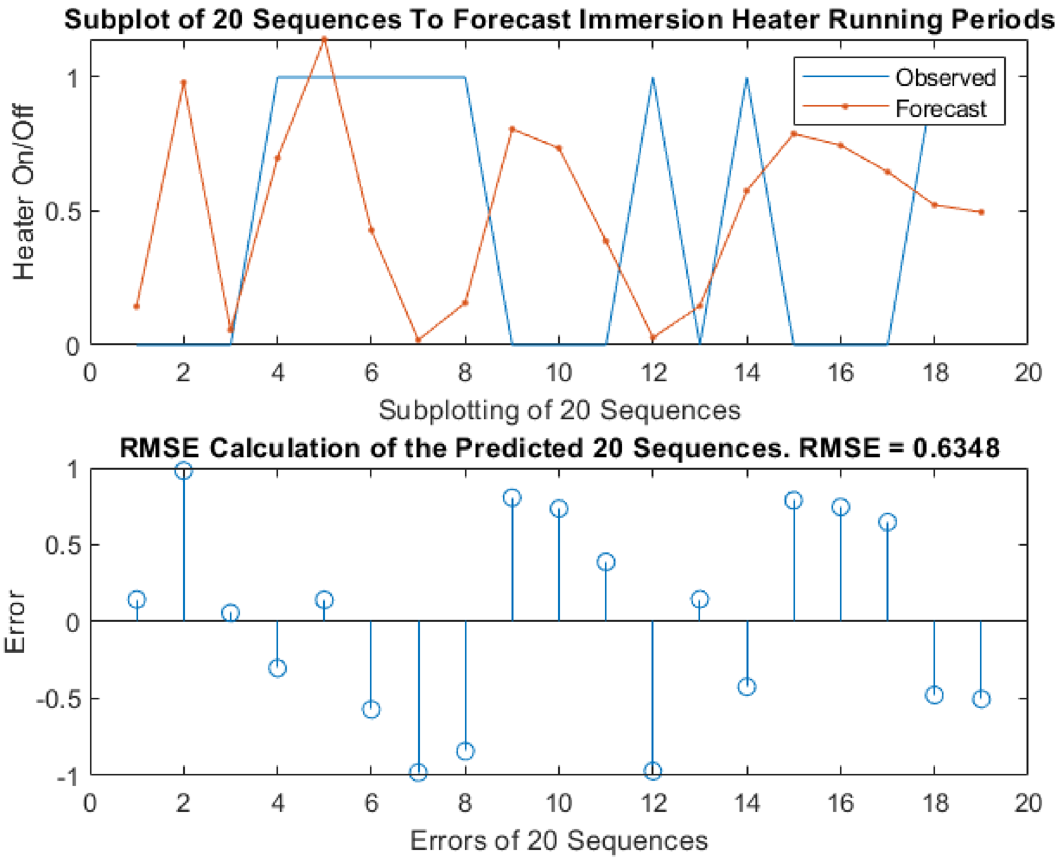


Figure 6.27: Sub-plotting the last 20 cycles and the correspondent RMSE error for each.

A similar trend can be seen in the overall detailed graph in Figure 6.28 where all types of training, validation and testing data are inspected.

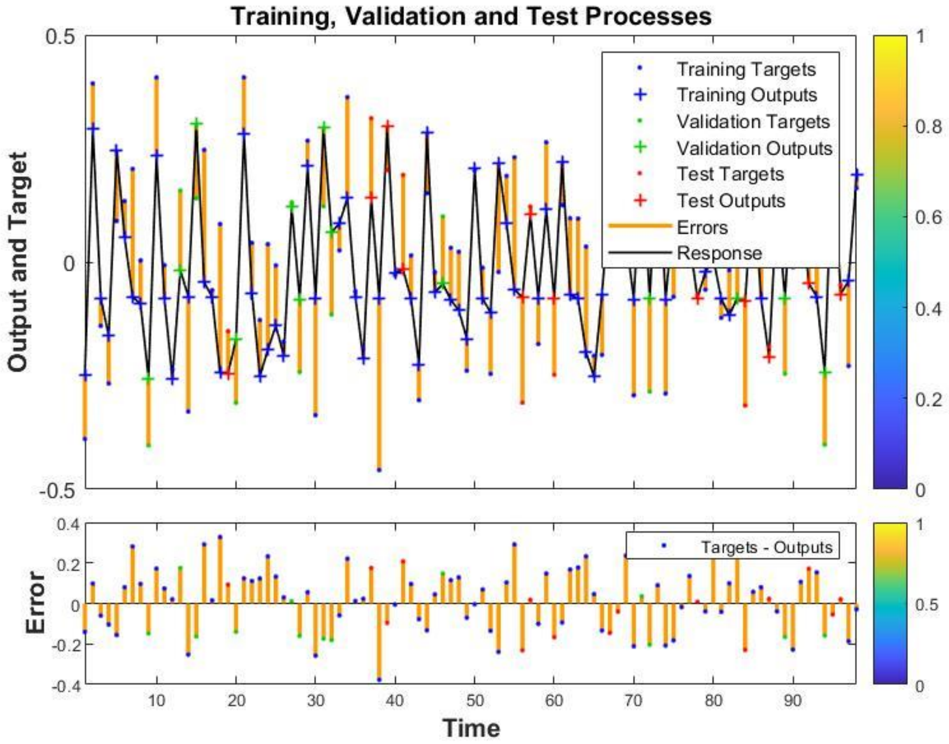


Figure 6.28: Sub-Plotting of all Data Types: Training, Validation and Testing.

After obtaining the prediction data, we can move to the next step to use it to apply the related ASRPP policy as explained in the next section.

6.3.3. The Application of The Automatic Scheduling of Running Periods Policy

As explained in section 5.2., the basic idea behind this policy is automatically controlling the running periods of the heater by switching it On/Off depending on the usage. This has been carried out using the previously mentioned RECHS application (section 5.5.) where the energy consumption of the immersion was measured before and after applying this policy.

Figure 6.17 has shown the indicator of both measured *usage* and energy consumption, where Figure 6.29 illustrates the same indicators after applying the ASRPP policy, where the system has turned off the appliance during the night (between 00:00 and 5:00 clock), which lead to saving the wasted energy.

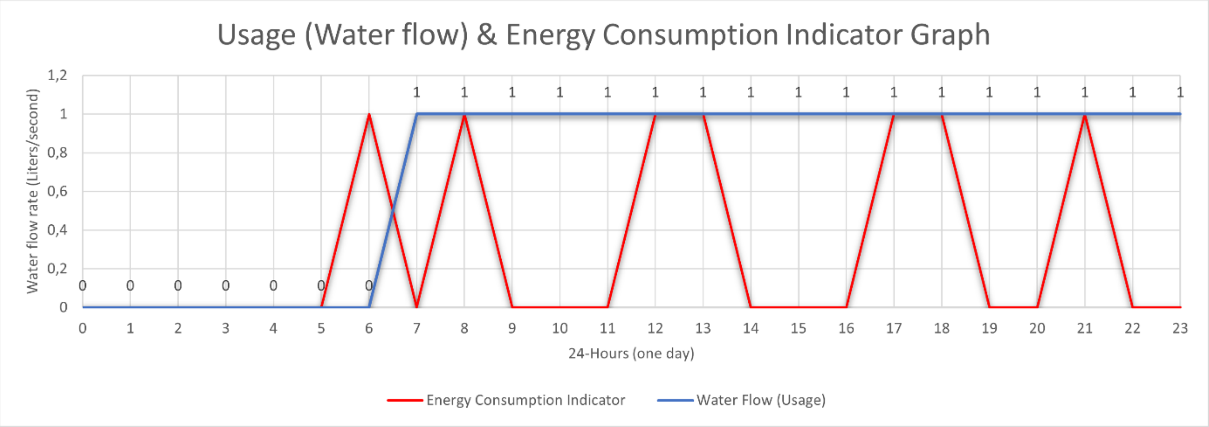


Figure 6.29: A comparison between the energy consumption (in kWh) and the Water Flow (usage) After applying the ASRPP Policy

The following Table 6.6 shows the water flow (appliance usage) and the running periods of the heater before and after applying the ASRPP policy, together with energy consumed in kWh. Using the figures in this table it is possible to calculate the amount of energy saved by calculating the average energy consumed *before* and *after* applying the policy. This can be achieved by creating a new column called *Possibly estimated saved energy (in kWh)* (highlighted in light green), which is calculated based on the flowchart from Figure 6.30.

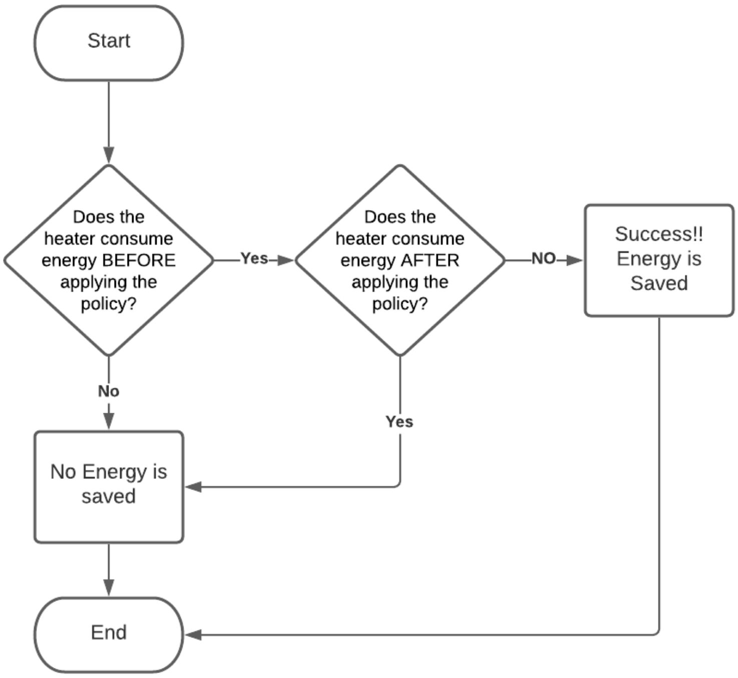

Start
Does the heater consume energy BEFORE applying the policy?
Yes
Does the heater consume energy AFTER applying the policy?
NO
Success!! Energy is Saved
No
Yes
No Energy is saved
End

Figure 6.30: Flowchart to decide when the appliance is saving energy after applying the ASRPP policy

24-Hours	Appliance Consuming Energy (BEFORE) applying the ASRPP Policy	Appliance Energy Consumption (AFTER) applying the ASRPP Policy	Appliance Is Being Used Yes/No	Energy Lost Yes/No	Energy Consumption (in kWh)	Possibly estimated Saved Energy (in kWh)
0	1	0	0	Yes	0.165	0.165
1	0	0	0	No	0	0
2	0	0	0	No	0	0
3	1	0	0	Yes	0.202	0.202
4	1	0	0	Yes	0.86	0.86
5	0	0	0	No	0	0
6	1	1	0	Yes	0.46	0
7	0	0	1	No	0	0
8	1	1	1	Yes	0.62	0
9	0	0	1	No	0	0
10	0	0	1	No	0	0
11	0	0	1	No	0	0
12	1	1	1	No	0.33	0
13	1	1	1	No	0.58	0
14	0	0	1	No	0	0
15	0	0	1	No	0	0
16	0	0	1	No	0	0
17	1	1	1	No	0.55	0
18	1	1	1	No	0.127	0
19	1	0	1	No	0.219	0
20	1	0	1	No	0.191	0
21	0	1	1	No	0	0
22	0	0	1	No	0	0
23	0	0	1	No	0	0

Table 6.6: Appliance running period (Energy consumption), appliance usage (water flow bigger than zero) and the estimated energy lost, the average kWh consumed every hour during the day, and the possibly estimated saved energy

The system has achieved a noticeable energy-saving rate, approximately 36% which is equivalent to 10.872 kWh from the average daily total energy consumption of 1.244 kWh, by switching off the appliance when it is not used and switching it on shortly before using it again.

6.4. Conclusion

In this chapter, two experiments were illustrated as an example of the possible implementation of the proposed I3SEM framework, where each of them was dedicated to demonstrate applying some components of the proposed framework on a sample appliance from the household and show how it did achieve a noticeable saving in the consumed energy reaching up to 36%, while keep offering almost the same level of comfort. Both appliances: the refrigerator and the immersion water heater were chosen based on several criteria: firstly, each one of them belongs to a different appliances' category; the refrigerator is considered a member of the uninterruptable devices, and the immersion heater is part of the schedulable appliances. Secondly, both are common devices that exist nearly in every household in the area where the implementation took place (in NRW state in the west part of Germany). Thirdly, the possibility to find and buy suitable and compatible hardware items such as sensors and actuators and attach them to these appliances was easy and cost-efficient. Fourthly, these devices fully belong to the household owner and are not shared with others, contrary to other potential appliances with high energy consumption, such as the heating system in the building which is shared by all other building inhabitants. Therefore, it is possible to switch those appliances on/off upon need to install sensors or try anything without interrupting others.

Both experiments began with the data preparation to decide which variable to consider and which one to discard. The decision was taken based on a number of data preparations steps started by ascertaining and eliminating the *measured internal humidity* which showed a high percentage of missing values. Then it continued by removing the date/time variable which is not relevant for this regression approach problem. Detecting outliers and anomalies was a considerable part of the job to ensure the data homogeneity and reliability. Finally, feature-wise and target-wise correlation assessments took place, to decide for the most relevant and effective features. This preparation phase leads to the first refrigerator experiment, to accept a total of 8 predictors, and remove or merge 8 others with other variables as shown in Table 6.2. The picture was almost similar in the second immersion heater experiment where 7 variables were accepted and 14 were merged or removed.

The prepared datasets were used to train different algorithms from regression and classification types using MATLAB and the Orange Data Mining Software. For the first refrigerator experiment, the regression approach was chosen because there is a need to predict continuous values (the energy consumption) and for the second immersion heater experiment the sequence classification with neural network data mining was applied, because there is a need to predict a discrete label bounded to time sequence. The resulted models were used to deliver forecasts of the future behaviour of appliances as a preparation for applying the correspondent policy; the *Energy Consumption based Appliance Substitution Policy (ECASP)* for the refrigerator experiment and *Automatic Scheduling of Running Periods Policy (ASRPP)* for the immersion heater. Which resulted in saving energy up to 22,2% for the refrigerator and 36% for the immersion heater.

The proposed framework attempts to deliver clear answers to the drawbacks of the reviewed frameworks by incorporating standards for exchanging, analysing and displaying energy data, and measuring the performance. Also, by supporting decision-taking mechanisms and organisation services to consider the amount of energy consumed by various assets, or by different processes, to enable energy optimization on both local and global levels. And meeting the requirements of compatibility, expandability and interoperability to support further future developments and extensions. Finally, offering a platform to run all together. Both presented case studies in this chapter demonstrated some of these points commenced by analysing and illustrating energy consumption data to gain an overall picture of some vital parameters of the system. This was followed by performing a number of data preparation processes to generate models and decide for the most suitable one in order to establish a proper base for predicting energy consumption in order to take decisions to enable energy optimization. The framework's integrability nature supports its ability to deal with various data structures and types resulting from different resources in this field, shaping them into processible datasets by offering a number of fundamental data integration and transformation components during the implementation. As discussed within the thesis, supporting scalability considered one of the basic pillars of the framework, therefore an examination test was carried on comparing performance parameters when running the application on initial hardware setup, and then run it on a scaled setup. Moreover, the illustrated case studies show how the whole system may look like when it deployed and operate as one unit.

Important to mention that both experiments were chosen to demonstrate the concept of the system, rather than energy savings. Both represent examples of the implementation of the proposed I3SEM Framework that can applied almost on any conventional or smart appliance inside any household with minimum hardware settings. In some cases, as done in the first experiment, the applied substation strategy based on the energy consumption may negatively affect the resulted footprint, because of the fact that producing new appliances may cost more energy than saved, however, this negative impact may be reduced due to offering some mitigating aspects represented in offering some recycling and spare parts opportunities, and apply this strategy exclusively in regions where energy prices are high. Moreover, applying the framework on the refrigerator in the first experiment, offer some additional benefits than a direct energy saving, shown in offering the real-life energy consumption data of the refrigerator while running under real life conditions, to the manufacturers to let them observe the possibilities to deal with eventual shortages and drawbacks.

7. Conclusion and Future Work

7.1. Conclusion

The main focus of this research is to address the evolving enormous energy consumption in the household sector within the last decades, and its disastrous consequences on the environment, planet resources, and householders' budgets. The key objective is proposing, implementing, and evaluating a contemporary integrated, scalable, smart energy management framework that assists in reducing the energy consumption in the household sector by applying a number of correspondent strategies and policies which utilise a set of observed and predicted system entities. This chapter converges the conducted research in corroboration of the literature in order to achieve the indicated research goals and objectives. The most important findings, which ultimately form the basis for a knowledge contribution in this area, are summarized.

The rapid development of smart techniques and the increasing maturity grade of machine learning technologies, besides the crucial need to overcome main drawbacks in the reviewed frameworks, were among the main pillars that support proposing the Integrated Scalable System for Smart Energy Management (I3SEM) framework utilised in this research. The framework's integrability nature supports its ability to deal with various data structures and types resulting from different resources in this field, shaping them into processible datasets by offering a number of fundamental data integration and transformation components. Moreover, there is an essential need to support the scalability attribute by offering a number of relevant modern software paradigms, due to the fact that the success of this framework is strictly bound to its ability to be rolled out to a huge number of households, and its ability to maintain an acceptable level of stability while being adapted to the rapidly evolving smart technological inventions in this field. Smartness is a pivotal and central characteristic of the framework that enables the integration of smart components to enhance the framework's overall capabilities and form its basis.

The initial stage of this research commenced with emphasizing the importance of energy management systems approaches, assessment of background and potential challenges, followed by a compiling of different statistics schemes related to the energy consumption datasets narrowed down to the household's sector in various geographical locations and application domains. Also, disclosing a description of current attempts aim to respond to the evolving challenge in the industry related to the production of clean energy, and providing statutory and legal rules and guidelines to enforce and govern actions in this area.

Establishing an in-depth understanding and enhancing the knowledge in this field was conducted by the literature review in the second chapter commenced with covering a number of Meta Operating Systems (MOS) which are established on top of an operating system allowing coordinating heterogeneous systems, devices, applications and processes allowing real-time communication. Energy management systems are considered as large and complex systems that operate in dynamic environments, therefore a clarification of the large systems' most relevant characteristics such as compatibility, expandability, interoperability, integrability, reliability and scalability, took part in this chapter. The discussion proceeds further to illustrate a number of data challenges related to the acquisition and storing of data. An analysis and comparison of the most relevant, existing conventional and intelligent energy management frameworks provided classification and insights into the relative merits and limitations of different approaches and techniques.

The literature review to intensify the knowledge of technologies and techniques which are considered the pillars to implement previously reviewed frameworks, and the proposed I3SEM framework, has continued in chapter 3 which commenced with recitation microservices, cloud computing and the Internet of Things (IoT) approach which meant to connect the ubiquitous appliances to the cloud. Then proceeds with unfolding a number of modern techniques and models pointed to safe pathways for constructing smart energy management systems such as big data, data mining, and machine learning approaches including supervised, unsupervised, classification, association, anomaly detection, clustering, regression and time series algorithms.

Energy management systems are required to offer an acceptable response to the challenging equation of achieving the most efficient energy consumption approaches without sacrificing the overall comfort level. The literature review revealed that these systems suffer from one or more drawbacks related to firstly, the deficiency of integration processes designed to deal with both conventional and smart appliances on one hand, and deal with data retrieved from different resources following different standards on the other hand. Secondly, the lack of integrated architecture that supports quality factors such as adaptability, expandability, and performance. Thirdly, the lack of standardisation and unified data architectures as the majority of them are following their own vendor's standards and data architecture; this resulted in complicating the overall collaboration and interoperability in the industry and prevent small vendors from contributing to the industry by developing specific aggregable units by following a certain standard and unified data architecture. Fourthly, the restricted applicability on legacy and modern and smart environments, covering almost every single appliance within the regular household. Fifthly, lack of mobility management and the shortage of engaging stakeholders in the whole process. And finally, the absence of mesh network nodes management in terms of security.

The proposed I3SEM, in chapter 4, provides a comprehensive and solid architecture to bypass these downsides by offering a unique structure that divides the framework into three main zones, and presenting a number of relevant generic components, moreover, utilising appropriate state-of-art and modern paradigms, besides offering essential approaches related to the context-sensitive analysis, detection and probability generation, predictive analysis and the alerting messaging. The division of the framework into two main zones combined with a gate zone improves several quality-driven aspects related to scalability, enhanced encapsulation, performance, and interoperability. The client zone is implemented and physically resides in the household site, and external APIs providers. This approach shifts a remarkable part of processing power from the central processing units in the cloud to the client and offer more privacy and enhanced security to deal with sensitive data within the correspondent household without the need to transfer it to the cloud, where only filtered, anonymous constraints are processed centrally. The cloud gate stands in front of the cloud zone to facilitate a number of relevant characteristics related to security and performance, by authorising, load-balancing incoming requests and caching responses. The cloud zone is the core and central unit of the architecture comprised of all necessary components to process and analyse the gathered data and deliver responses to all upcoming requests. Shifting shared units from the client zone to the cloud has a direct impact on enhancing scalability, modularity and performance.

In this research, household appliances are classified according to their operational nature into three main categories: uninterruptible, schedulable, and instant or run-on-demand appliances. In order to achieve the main goal of reducing energy consumption, three main strategies and policies were designed and introduced: Energy Consumption-based Appliance Substitution Policy (ECASP), Usage Percentage-based Appliance Substitution Policy (UPASP) and Automatic Scheduling of Running Periods Policy (ASRPP). Each policy applies to one or more appliances' groups, taking into consideration the capabilities and limitations of appliances from each category. Chapter 5 covered the novel aspects of the I3SEM framework capabilities by demonstrating them in two experiments applied to two different household appliances that belong to two categories: Refrigerator from the uninterruptable appliances category, and the immersion water heater from the schedulable appliances category. Both appliances were chosen because they are commonly used in households, and are suitable to be equipped with affordable additional sensors and actuators. Moreover, they can be switched on/off without interrupting anybody.

The final dataset entities used while applying the mentioned strategies are comprised of observed and predicted data. The observed data were retrieved from 5 specially designed and assembled Arduino-based systems: Z-Wave-based Energy Consumption Recorder (ZW-ECR), Arduino Uno-based Unique Occupant Detector (AU-UOD), Arduino Uno-based Refrigerator Fullness Detector (AU-RFD), Arduino Uno-based Refrigerator Settings Panel Reader (AU-RSPR) and Arduino Uno-based Immersion Heater Inspector (AU-IHI). However, the predicted data resulted from using the data mining techniques which follow the CRISP-DM methodology that includes understanding business, understanding data, preparing data, modelling and evaluation. The resulted models were used to deliver forecasts of the future behaviour of appliances as a preparation for applying the correspondent policy; the ECASP for the refrigerator experiment and ASRPP for the immersion heater. Which have delivered promising results by saving energy up to 22,2% for the refrigerator and 36% for the immersion heater.

Due to the applied methods, collected measurements, performed strategies and predictions, it was possible to make a clear comparison between the overall energy consumption 'before' and 'after', and show how percentual energy consumption can be reduced. Due to the fact that this research is moving among two relatively new areas; the reduction of energy consumption and various smart technologies, it was a hard and challenging task to go through a long and sensitive selecting process to decide on the most reliable, accurate and most-fit-for-purpose software and hardware. The decision was sometimes easy and straightforward, however, sometimes it must go through a tough, challenging, time-consuming, precise checking and testing process.

7.2. Contributions

The key finding of this research is proposing a robust, integrated, scalable smart energy management architecture that contributes to the knowledge of managing energy by overcoming shortcomings and drawbacks from existing approaches and addressing further benefits. The I3SEM represents a novel contribution by proposing a generic, integrated, scalable architecture that employs smart techniques. The proposed framework's architecture is divided into two main zones separated by a gate zone to enhance a number of quality-driven aspects related to privacy and security, by processing the sensitive and person-related data within the local mesh network. Moreover, increasing performance by integrating a distributed-computing-like approach, and improving the modularity by introducing modules with standard interfaces.

The I3SEM framework provides practical and effective ways to handle different data structures retrieved from different internal and external resources by utilising data integration and data transformation components. It does support scalability horizontally and vertically. Horizontally, by employing appropriate paradigms, to offer possibilities to expand the framework's capabilities. And vertically, by having autonomous control units that observe the system's load and scale it up or down accordingly. The I3SEM framework provides real-time evaluation and visualization of results, predictions, charts and plain figures to reflect the outcomes, or can be passed to third parties such as local energy providers, appliances' manufacturers and involved governmental agencies.

Supporting data mining techniques is an essential and core part of the framework, in order to enhance the overall dataset required for the decision-making component. This was achieved by introducing the Data Analytical Engine (DAE), and defining a set of evaluation metrics to enhance the predicted models' reliability and correctness level. Also supports the decision-making processes during the application phase of various strategies and policies. Moreover, additional holistic security assessments are considered in the I3SEM framework to address a number of key deficits observed in the reviewed literature, to offer additional safeguarding for the mesh networks. This framework further extends the panel of household appliances to be part of the mesh network by applying IoT concepts to add native support for both smart and conventional appliances. As a response to the absence of a mature and agreed-upon standardisation in this evolving field, the framework contributes the Data Collection and Integration (DCI) component which includes several conversion, transformation and translation modules to support the inter-components communication.

Based on this thesis the best presentation award was obtained in the Smart Energy Management and Energy Efficiency Conference, in August 2020 in Paris by the Program Committee as per the Conference Awards Scheme, for the paper titled "Reduction of Energy Consumption Using Smart Home Techniques in the Household Sector" [193], and same paper was chosen to be published in the International Journal of Energy and Environmental Engineering. Another journal paper titled "Predictive Analytics based Smart Energy Management Framework for Household Appliances" is submitted to the Environmental Progress & Sustainable Energy, published by John Wiley and Sons Inc., on 12th of May 2021 which is still under review.

7.3. Future Work

The proposed framework should be considered as an initial step towards building a comprehensive smart energy management framework that suggests a wide range of layers, zones and components to deal with the evolving nature of this field, where continuous and rapid development steps are achieved. The proposed future work can be categorized into three main directions: introducing new layers and zones, augmenting the current zones with additional components, and finally enhancing the implementation phase. These suggested future work aim to bring the framework to the next level, and deal with the framework's limitations which can be summarised in: missing a concept to deal with clean-energy production and storage from reasonable sources, having an overhead administration and increased costs, spending a considerable portion of time and resources in data preparation and analysis, having limited reporting capabilities, and missing comprehensive policies to deal with footprint resulted from applying various strategies related to appliances' substitution. However, as follows, each point of these limitations is discussed, and potential layers, components or modules are suggested to deal with each issue.

The proposed framework might be extended by introducing a new layer that considers the utilisation of clean energy production from reasonable sources such as solar panels, combined with storage capabilities to reduce the dependence on local energy suppliers and to contribute to increase the generation of clean energy and reduce the energy consumption during peaks. Also, introducing a new zone called *Community Zone* that includes collective households residing together such as several departments in the same building. This will minimize the overhead administration efforts and reduce costs. Moreover, a number of components can be introduced such as an A*utomatic Data Validator* which is responsible to validate and discover anomalies, deficiencies in the data during the collecting and processing phase. This leads to saving storage space and reducing the processing time. Also, the generating of the standard reports can be enhanced by adding the C*ustomised Reports Creator* component which allows third parties to generate specially dedicated reports which are not included in the standard report templates. For example, the amount of saved energy after applying the framework. Moreover, the framework's interoperability can be extended by adding a new component *Automatic Smart Appliance Installer*, which is used by admins to add a new smart appliance to the mesh network by defining the data nodes interface by admins, so data can flow from the new appliance into the system effortlessly. And finally, inventing the component *Appliances' Recycler* to add a possibility to recycle old, substituted appliances by selling them as spare parts, or recycling them as a whole. Important to ensure that they are not sold as a whole to another customer, unless it detaches another higher-energy-consuming appliance.

Furthermore, the future work might be concentrating on applying the framework on a bigger scale study, covering more appliances within the household, or integrating a large number of households to aggregate the data. Moreover, increasing the sensing accuracy by utilising better hardware to enhance the quality of the retrieved data. Moreover, implementing a case study to apply the third strategy which is based on substituting appliances based on their usage. Appliances are observed to find out whether they have bigger capacity than needed by the occupants, then making recommendations to replace them with smaller ones, to reduce energy.

References

[1] S. Van Dam, C. Bakker and J. Buiter, “Do home energy management systems make sense? Assessing their overall lifecycle impact,” *Energy Policy,* vol. 63, pp. 398-407, 2013.

[2] S. Marksteiner, V. J. Exposito Jimenez, H. Valiant and H. Zeiner, “An overview of wireless IoT protocol security in the smart home domain,” in *2017 Internet of Things Business Models, Users, and Networks*, 2017, pp. 1-8.

[3] A. Al-Ali, I. Zualkernan, M. Rashid, R. Gupta and M. Alikarar, “A smart home energy management system using IoT and big data analytics approach,” *IEEE Transactions on Consumer Electronics,* vol. 63, no. 4, pp. 426-434, 2017.

[4] M. Su, C. Chen and Z. Yang, “Urban energy structure optimization at the sector scale: considering environmental impact based on life cycle assessment,” *Journal of Cleaner Production,* vol. 112, no. 2, pp. 1464-1474, 2016.

[5] I. Pineda and P. Tardieu, “Windeurope.org - Wind in power 2017 Annual combined onshore and offshore wind energy statistics.,” Feb. 2018. [Online]. Available: https://windeurope.org/wp-content/uploads/files/about-wind/statistics/WindEurope-Annual-Statistics-2017.pdf. [Accessed 19 June 2019].

[6] H. Ritchie, “How have the world’s energy sources changed over the last two centuries?,” 1 December 2021. [Online]. Available: https://ourworldindata.org/global-energy-200-years?country=. [Accessed 24 January 2022].

[7] EUROSTAT, “Eurostat,” European Commission, 02 Feb. 2020. [Online]. Available: https://ec.europa.eu/eurostat/web/main/home. [Accessed 02 Feb. 2020].

[8] International Energy Agency (iea), “World Energy Outlook 2021,” October 2021. [Online]. Available: https://www.iea.org/reports/world-energy-outlook-2021. [Accessed 24 January 2022].

[9] W. Shepherd and L. Zhang, Electricity Generation Using Wind Power, Second Edition ed., Singapore: World Scientific Publishing Company., 2017.

[10] L. Wilson, “Average household electricity use around the world,” 24 January 2022. [Online]. Available: http://shrinkthatfootprint.com/average-household-electricity-consumption. [Accessed 24 January 2022].

[11] L. Wilson, “What are the major uses of electricity?,” 2021. [Online]. Available: https://shrinkthatfootprint.com/how-do-we-use-electricity. [Accessed 17 10 2021].

[12] U.S. Energy Information Administration, “U.S. energy facts explained,” April 2021. [Online]. Available: https://www.eia.gov/energyexplained/us-energy-facts/. [Accessed 17 October 2021].

[13] J. Dillinger, “WorldAtlas - Cost Of Electricity By Country,” 2019. [Online]. Available: https://www.worldatlas.com/articles/electricity-rates-around-the-world.html. [Accessed 26 June 2019].

[14] The World Bank, “Data | The World Bank,” 2019. [Online]. Available: https://data.worldbank.org/share/widget?indicators=EN.CO2.ETOT.ZS&view=map. [Accessed 26 June 2019].

[15] H. Ritchie and M. Roser, “Renewable Energy. [online] Our World in Data,” 2019. [Online]. Available: https://ourworldindata.org/renewable-energy. [Accessed 26 June 2019].

[16] Our World in Data, “Renewable energy generation, World,” 24 January 2022. [Online]. Available: https://ourworldindata.org/grapher/modern-renewable-energy-consumption?country=~OWID_WRL. [Accessed 24 January 2022].

[17] ISO, “ISO. (2019). ISO 50001 Energy management.,” 2019. [Online]. Available: https://www.iso.org/iso-50001-energy-management.html. [Accessed 23 May 2019].

[18] www.euenergylabels.com, “How to create correct European Union Energy Labels,” [Online]. Available: http://www.euenergylabels.com/Creating-EU-Energy-Labels-(pitfalls-thereof).htm. [Accessed 29 Jan. 2020].

[19] ENERGY STAR, “ENERGY STAR | The Simple Choice for Energy Efficiency,” 2019. [Online]. Available: https://www.energystar.gov/. [Accessed 23 May 2019].

[20] Organisation for Economic Co-operation and Development, “Oecd.org,” 17 Feb. 2015. [Online]. Available: http://www.oecd.org/sti/ind/DSTI-SU-SC(2014)14-FINAL-ENG.pdf. [Accessed 21 May 2019].

[21] International Energy Agency, “Iea.org - World Energy Outlook 2017 China : Key Findings.,” 2019. [Online]. Available: https://www.iea.org/weo/china/. [Accessed 26 June 2019].

[22] A. Averian, “A Survey on Context Aware Computing in Digital Ecosystems,” *Ann. Univ. Spiru Haret,* vol. 13, no. 1, pp. 21-40, 2017.

[23] G. Paroux, L. Martin, J. Nowalczyk and I. Demeure, “Transhumance: A powersensitive middleware for data sharing on mobile ad hoc networks,” in *Proceedings of the 7th international Workshop on Applications and Services in Wireless*, Spain, 2007.

[24] G. Paroux, l. Demeure and L. Reynaud, “A Power-aware Middleware for Mobile Ad-hoc Networks,” in *Proceedings of the 8th International Conference on New Technologies in Distributed Systems (NOTERE’08)*, ACM, 2008.

[25] I. Demeure, G. Paroux, J. Hernando-ureta, A. Khakpour and J. Nowalczyk, “An energy-aware middleware for collaboration on small scale manets,” in *Proceedings of the Autonomous and Spontaneous Networks Symposium (ASN’08)*, Paris, 2008.

[26] V. Vardhan, W. Yuan, A. III, S. Adve, R. Kravets, K. Nahrstedt, D. Sachs and D. Jones, “Grace-2: integrating finegrained application adaptation with global adaptation for saving energy.,” *International Journal of Engineering Science,* vol. 4, no. 2, pp. 152-169, 2009.

[27] Y. Xiao, P. Hui, P. Savolainen and a. Yla-J, “Cascap: cloud-assisted context-aware power management for mobile devices,” in *Proceedings of the 2nd international workshop on Mobile cloud computing and services, MCS ’11, ACM*, New York, 2011.

[28] S. Mohapatra, N. Dutt, A. Nicolau and N. Venkatasubramanian, “Dynamo: A cross-layer framework for end-to-end qos and energy optimization in mobile handheld devices.,” *Selected Areas in Communications, IEEE Journal,* vol. 25, no. 4, pp. 722-737, 2007.

[29] S. Mohapatra and N. Venkatasubramanian, “PARM : power aware reconfigurable middleware,” in *23rd International Conference on Distributed Computing Systems, 2003. Proceedings.*, 2003.

[30] Z. Heng, C. Schlatter Ellis, A. R. Lebeck and V. Amin, "ECOSystem: managing energy as a first class operating system resource," in *ASPLOS X*, 2002.

[31] ISO/IEC JTC 1/SC 7 Software and systems engineering, "ISO/IEC 25010:2011 Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — System and software quality models," ISO/IEC JTC 1/SC 7 Software and systems engineering, 2017. [Online]. Available: https://www.iso.org/standard/35733.html. [Accessed 01 06 2020].

[32] CORDIS EU Research results , "EE-12-2017 - Integration of Demand Response in Energy Management Systems while ensuring interoperability through Public Private Partnership (EeB PPP)," EUROPEAN COMISSION, 31 October 2017. [Online]. Available: https://cordis.europa.eu/programme/id/H2020_EE-12-2017. [Accessed 01 06 2020].

[33] K. Vikhorev, R. Greenough and N. Brown, "An advanced energy management framework to promote energy awareness," *Journal of Cleaner Production,* vol. 43, pp. 103-112, 2013.

[34] Statista, Inc., "Amount of data created, consumed, and stored 2010-2025," 7 June 2021. [Online]. Available: https://www.statista.com/statistics/871513/worldwide-data-created/. [Accessed 16 June 2021].

[35] A. Heiss, "Big Data Challenges in Big Science," *Computing and Software for Big Science,* vol. 3, no. 1, p. 15, 2019.

[36] J. van Wezel, A. Streit, C. Jung, R. Stotzka, S. Halstenberg, F. Rigoll, A. Garcia, A. Heiss, K. Schwarz, M. Gasthuber and A. Giesler, "Data Life Cycle Labs, A New Concept to Support Data-Intensive Science.," 2020. [Online]. Available: https://arxiv.org/abs/1212.5596. [Accessed 21 Jan. 2020].

[37] Helmholtz.de, "Helmholtz Data Federation (HDF)," 2020. [Online]. Available: https://www.helmholtz.de/en/research/information-data-science/helmholtz-data-federation-hdf/. [Accessed 21 Jan. 2020].

[38] P. Misra, A. Pal, B. Purushothaman, C. Bhaumik, D. Swamy, V. Subrahmanian, D. Kar, S. Naskar, S. Ghosh and S. Adak, "Computer Platform for Development and Deployment of Sensor Data Based Applications and Services". United States Patent US20140359552A1, 2019.

[39] A. Meier and M. Kaufmann, "SQL & NoSQL Databases: Models, Languages, Consistency Options and Architectures for Big Data Management," Wiesbaden, Springer Fachmedien Wiesbaden, 2019, pp. 201-218.

[40] Info.couchbase.com, "Explore the only data platform built on the world's most powerful NoSQL technology," 2020. [Online]. Available: https://info.couchbase.com/big-data.html. [Accessed 22 Jan. 2020].

[41] Couchbase.com, "Solutions – IoT Data Management | Couchbase," 2020. [Online]. Available: https://www.couchbase.com/solutions/iot-data-management. [Accessed 22 01 2020].

[42] A. Zaina, A. Reinhardt and J. Huchtkoetter, "Relational or Non-Relational? A Comparative Evaluation of Database Solutions for Energy Consumption Data," in *In Proceedings of the Ninth International Conference on Future Energy Systems*, 2018.

[43] Y. Onda, K. Taniguchi, K. Yoshimura, H. Kato, J. Takahashi, Y. Wakiyama, F. Coppin and H. Smith, "Radionuclides from the Fukushima Daiichi nuclear

power plant in terrestrial systems," *Nature Reviews Earth & Environment,* vol. 1, no. 12, pp. 644-660, 2020.

[44] N. Loganathan, P. S. Mayurappriyan and K. Lakshmi, "Smart energy management systems: a literature review.," *In MATEC Web of Conferences,* vol. 225, p. 01016, 2018.

[45] B. Asare-Bediako, W. Kling and P. Ribeiro, "September. Home energy management systems: Evolution, trends and frameworks," *In 2012 47th International Universities Power Engineering Conference (UPEC) IEEE,* pp. 1-5, 2012.

[46] K. P. Natarajan and S. Bhagavath Singh, "Wireless sensor network based remote monitoring system in smart grids," *International Journal of Control Theory and Applications,* vol. 9, no. 14, p. 6639–6646, 2016.

[47] D. Jelic, D. Gordic, M. Babic, D. Koncalovic and V. Sustersi, "Review of existing energy management standards and possibilities for its introduction in Serbia," *Thermal Science,* vol. 14, no. 3, pp. 613-623, 2010.

[48] A. Prashar, "Adopting PDCA (Plan-Do-Check-Act) cycle for energy optimization in energy-intensive SMEs,," *Journal of Cleaner Production,* vol. 145, pp. 277-293, 2017.

[49] P. Heusinger and M. Wagner, "Open source framework for energy management - Software and hardware aspects of an energy management gateway," *2013 IEEE 18th International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD),* p. 129–133, 2013.

[50] G. &. B. Vikhorev, "An advanced energy management framework to promote energy awareness," *Journal of Cleaner Production,* vol. 43, no. C, pp. 103-112, 2013.

[51] O. Foundation., "What is OPC? - OPC Foundation," 2019. [Online]. Available: https://opcfoundation.org/about/what-is-opc/. [Accessed 25 June 2019].

[52] S. M. E. B. R. &. S. Y.-J. Kucuksari, "Energy management system using policy based hierarchical simulation framework," *62nd IIE Annual Conference and Expo 2012, Orlando, FL, United States, 5/19/12. Institute of Industrial Engineers,* pp. 1545-1554, 2012.

[53] Y. Kwak, J. Huh and C. Jang, "Development of a model predictive control framework through real-time building energy management system data," *Applied Energy,* vol. 155, pp. 1-13, 2015.

[54] B. Zhou, W. Li, K. W. Chan, Y. Cao, Y. Kuang, X. Liu and X. Wang, "Smart home energy management systems: Concept, configurations, and scheduling strategies.," *Renewable and Sustainable Energy Reviews,* vol. 61, pp. 30-40, 2016.

[55] K. I. Katsigarakis, G. D. Kontes, G. I. Giannakis and D. V. Rovas, "Sense-Think-Act Framework for Intelligent Building Energy Management," *Computer-Aided Civil and Infrastructure Engineering,* vol. 31, no. 1, pp. 50-64, 2016.

[56] D. Minoli, K. Sohraby and B. Occhiogrosso, "IoT Considerations, Requirements, and Architectures for Smart Buildings—Energy Optimization and Next-Generation Building Management Systems.," *IEEE Internet of Things Journal,,* vol. 4, no. 1, pp. 269-283, 2017.

[57] L. Salman, S. Salman, S. Jahangirian, M. Abraham, F. German, C. Blair and P. Krenz, "Energy Efficient IoT-Based Smart Home," *2016 IEEE 3rd World Forum on Internet of Things (WF-IoT),* pp. 526 - 529, 2016.

[58] M. Collotta and G. Pau., "An Innovative Approach for Forecasting of Energy Requirements to Improve a Smart Home Management System Based on BLE," *IEEE Transactions on Green Communications and Networking Journal,* vol. 1, no. 1, pp. 112-120, 2017.

[59] I. F. Akyildiz, X. Wang and W. Wang, "Wireless mesh networks: a survey," *Computer networks,* vol. 47, no. 4, pp. 445-487, 2005.

[60] B. Butzin, F. Golatowski and D. Timmermann, "Microservices approach for the internet of things," *Emerging Technologies and Factory Automation (ETFA), 2016 IEEE 21st International Conference,* pp. 1-6, 2016.

[61] M. A. Jarwar, A. Sajjad and C. Ilyoung, "Microservices Model to Enhance the Availability of Data for Buildings Energy Efficiency Management Services," *Energies,* vol. 12, no. 3, p. 360, 2019.

[62] D. Zhang, S. Li, M. Sun and Z. O'Neill, "An Optimal and Learning-Based Demand Response and Home Energy Management System," *IEEE Transactions on Smart Grid,* vol. 7, no. 4, pp. 1790-1801, 2016.

[63] N. Prakash and D. P. Vadana, "Machine Learning Based Residential Energy Management System," in *2017 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC)*, Coimbatore, 2017.

[64] A. Amazon, "IoT Applications &amp; Solutions | What is the Internet of Things (IoT)? | AWS," 2018. [Online]. Available: https://aws.amazon.com/iot/. [Accessed 10 October 2018].

[65] D. T. Nguyen, C. Song, Z. Qian, S. V. Krishnamurthy, E. J. M. Colbert and P. McDaniel, "IotSan: fortifying the safety of IoT systems," *CoNEXT '18: Proceedings of the 14th International Conference on emerging Networking EXperiments and Technologies,* pp. 191-203, 2018.

[66] N. Dragoni, S. Giallorenzo, A. L. Lafuente, M. Mazzara, F. Montesi, R. Mustafin and L. Safina, "Microservices: Yesterday, Today, and Tomorrow," in *Present and Ulterior Software Engineering*, Cham, Springer, 2017, pp. 195-216.

[67] J. Thönes, "Microservices," *IEEE software,* vol. 32, no. 1, pp. 116-116, 2015.

[68] K. Khanda, D. Salikhov, K. Gusmanov, M. Mazzara and N. Mavridis, "Microservice-Based IoT for Smart Buildings," in *2017 31st International Conference on Advanced Information Networking and Applications Workshops (WAINA)*, Aina, 2017.

[69] The Jolie Team, "Jolie Programming Language - Official Website," The Jolie project is open source., 6 November 2017. [Online]. Available: https://www.jolie-lang.org/. [Accessed 03 March 2020].

[70] M. Yassein, W. Mardini and Khalil, "A. Smart homes automation using Z-wave protocol," *Engineering & MIS (ICEMIS), International Conference,* pp. 1-6, 2016, September.

[71] L. Eldén, Matrix methods in data mining and pattern recognition, Second Edition ed., vol. 15, Linköping: SIAM, 2019, p. 3.

[72] R. O. Duda, P. E. Hart and D. G. Storck, Pattern Classification, 2nd edition, New York: Wiley-Interscience, 2001.

[73] P. Zikopoulos, Understanding big data : analytics for enterprise class Hadoop and streaming data, New York: McGraw-Hill, 2012.

[74] Tech Entice, “The data veracity - Big Data,” Tech Entice, [Online]. Available: https://www.techentice.com/the-data-veracity-big-data/. [Accessed 2020 Feb, 02].

[75] D. R. Schrider and A. D. Kern, “Supervised Machine Learning for Population Genetics: A New Paradigm,” *Trends in Genetics,* vol. 34, no. 4, pp. 301-312, 2018.

[76] J. Berral, Í. Goiri, R. Nou, F. Julià, J. Guitart, R. Gavaldà and J. Torres, “Towards energy-aware scheduling in data centers using machine learning.,” in *Proceedings of the 1st International Conference on Energy-Efficient Computing and Networking*, New York, NY, USA, Association for Computing Machinery, 2010, pp. 215-224.

[77] H. Huo, Z. Rong, O. Kononova, W. Sun, T. Botari, T. He, V. Tshitoyan and G. Ceder, “Semi-supervised machine-learning classification of materials synthesis procedures,” *npj Computational Materials* , vol. 5, no. 62, 2019.

[78] Kognitio, “https://kognitio.com/,” 2020. [Online]. Available: Big Data Analytics Platform | Kognitiohttps://kognitio.com/. [Accessed 24 Jan. 2020].

[79] EDUCBA, “Data mining vs Machine learning - 10 Best Thing You Need To Know.,” 2019. [Online]. Available: https://www.educba.com/data-mining-vs-machine-learning/. [Accessed 24 Oct. 2019].

[80] J. Brownlee, “Supervised and Unsupervised Machine Learning Algorithms,” 16 March 2016. [Online]. Available: https://machinelearningmastery.com/supervised-and-unsupervised-machine-learning-algorithms/. [Accessed 02 March 2020].

[81] Y. Gao, E. Tumwesigye, B. Cahill and K. Menzel, “Using data mining in optimisation of building energy consumption and thermal comfort management,” in *The 2nd International Conference on Software Engineering and Data Mining*, 2010.

[82] S. Althaher, P. Mancarella and J. Mutale, “Automated demand response from home energy management system under dynamic pricing and power and comfort constraints,” *IEEE Transactions on Smart Grid,* vol. 6, pp. 1874-1883, 2015.

[83] K. Dittawit and F. A. Aagesen, “Home energy management system for electricity cost savings and comfort preservation,” in *2014 IEEE Fourth International Conference on Consumer Electronics Berlin (ICCE-Berlin),* Berlin, 2014.

[84] R. Varun Arvind, R. Rohith Raj, R. Ranjithh Raj and N. Krishna Prakash, “Industrial automation using wireless sensor networks,” *Indian Journal of Science and Technology,* vol. 9, no. 11, p. 1–8, 2016.

[85] C. H. Antunes, A. Soares and Á. Gomes, “An energy management system for residential demand response based on multiobjective optimization,” *IEEE Smart Energy Grid Engineering,* pp. 90-94, 2016.

[86] P. Dongbaare, S. O. Osuri and S. P. Daniel Chowdhury, “A smart energy management system for residential use,” *2017 IEEE PES PowerAfrica,* pp. 612-616, 2017.

[87] R. Rajasekaran, S. Manikandaraj and R. Kamaleshwar, “Implementation of Machine Learning Algorithm for predicting user behavior and smart energy management,” in *2017 International Conference on Data Management, Analytics and Innovation (ICDMAI),* Pune, 2017.

[88] K. Aurangzeb, "Short Term Power Load Forecasting using Machine Learning Models for energy management in a smart community," *2019 International Conference on Computer and Information Sciences (ICCIS),* pp. 1-6, 2019.

[89] M. Hossain, S. Mekhilef, M. Danesh, L. Olatomiwa and S. Shamshirband, "Application of extreme learning machine for short term output power forecasting of three grid-connected PV systems," *Journal of Cleaner Production,* vol. 167, pp. 395-405, 2017.

[90] H. Bendu, B. B. V. L. Deepak and S. Murugan, "Multi-objective optimization of ethanol fuelled HCCI engine performance using hybrid GRNN–PSO," *Applied Energy,* vol. 187, pp. 601-611, 2017.

[91] R. Deo, X. Wen and F. QiA, "wavelet-coupled support vector machine model for forecasting global incident solar radiation using limited meteorological dataset," *Applied Energy,* vol. 168, pp. 568-593, 2016.

[92] T. Shireen, C. Shao, H. Wang, J. Li, X. Zhang and M. Li, "Iterative multi-task learning for time-series modeling of solar panel PV outputs," *Applied Energy,* vol. 212, 2018.

[93] F. Wang, Z. Zhen, Z. Mi, H. Sun, S. Su and G. Yang, "Solar irradiance feature extraction and support vector machines based weather status pattern recognition model for short-term photovoltaic power forecasting," *Energy Build,* vol. 86, pp. 427-438, 2015.

[94] P. Ramsami and V. Oree, "A hybrid method for forecasting the energy output of photovoltaic systems," *Energy Convers Manage,* vol. 95, pp. 406-413, 2015.

[95] J. Xiea, H. Lib, Z. Maa, Q. Sunc, F. Wallinb, Z. Sid and J. Gu, "Analysis of key factors in heat demand prediction with neural networks," *Energy Procedia,* vol. 105, pp. 2965-2970, 2017.

[96] Z. Ma, J. H. Xue, A. Leijon, Z. H. Tan and J. Guo, "Decorrelation of neutral vector variables: theory and applications," *IEEE Trans Neural Netw Learn Syst,* pp. 1-15, 2016.

[97] D. Koolen, N. Sadat-Razavi and W. Ketter, "Machine Learning for Identifying Demand Patterns of Home Energy Management Systems with Dynamic Electricity Pricing," *Applied Sciences,* vol. 7, no. 11, p. 1160, 2017.

[98] M. Vahedipour-Dahraie, H. Najafi, A. Anvari-Moghaddam and J. Guerrero, "Study of the Effect of Time-Based Rate Demand Response Programs on Stochastic Day-Ahead Energy and Reserve Scheduling in Islanded Residential Microgrids," *Applied Science,* vol. 7, p. 378, 2017.

[99] B. Yener, A. Taşcıkaraoğlu, O. Erdinç, M. Baysal and J. Catalão, "Design and implementation of an interactive interface for demand response and home energy management applications," *Applied Sciences,* vol. 7, no. 6, p. 641, 2017.

[100] C. Fan, F. Xiao, C. Yan, C. Liu, Z. Li and J. Wang, "A novel methodology to explain and evaluate data-driven building energy performance models based on interpretable machine learning," *Applied Energy,* vol. 235, pp. 1551-1560, 2019.

[101] C. Miller, Z. Nagy and A. Schlueter, "A review of unsupervised statistical learning and visual analytics techniques applied to performance analysis of non-residential buildings," *Renewable and Sustainable Energy Reviews,* vol. 81, pp. 1365-1377, 2018.

[102] C. Fan, F. Xiao, Z. Li and J. Wang, "Unsupervised data analytics in mining big building operational data for energy efficiency enhancement: A review.," *Energy and Buildings,* vol. 159, pp. 296-308, 2018.

[103] M. Molina-Solana, M. Ros, M. Ruiz, J. Gómez-Romero and M. Martin-Bautista, "Data science for building energy management: A review," *Renewable and Sustainable Energy Reviews,* vol. 70, pp. 598-609, 2017.

[104] K. Amasyali and N. El-Gohary, "A review of data-driven building energy consumption prediction studies," *Renewable and Sustainable Energy Reviews,* vol. 81, pp. 1192-1205, 2018.

[105] Z. Yu, F. Haghighat and B. Fung, "Advances and challenges in building engineering and data mining applications for energy-efficient communities," *Sustainable Cities and Society,* vol. 25, pp. 33-38, 2016.

[106] Y. Ding, Q. Zhang and T. Yuan, "Research on short-term and ultra-short-term cooling load prediction models for office buildings," *Energy and Buildings,* vol. 154, pp. 254-267, 2017.

[107] C. Fan, F. Xiao and Y. Zhao, "A short-term building cooling load prediction method using deep learning algorithms," *Appl Energy,* vol. 195, pp. 222-233, 2017.

[108] A. Rahman, V. Srikumar and A. Smith, "Predicting electricity consumption for commercial and residential buildings using deep recurrent neural networks," *Appl Energy,* vol. 212, pp. 372-385, 2018.

[109] Y. Chen and H. Tan, "Short-term prediction of electric demand in building sector via hybrid support vector regression," *Appl Energy,* vol. 204, pp. 1363-1374, 2017.

[110] Z. Afroz, G. Shafiullah, T. Urmee and G. Higgins, "Prediction of Indoor Temperature in an Institutional Building," *Energy Procedia,* vol. 142, pp. 1860-1866, 2017.

[111] A. Geronazzo, G. Brager and S. Manu, "Making sense of building data: New analysis methods for understanding indoor climate," *Building and Environment,* vol. 128, pp. 260-271, 2018.

[112] M. Biswas, M. Robinson and N. Fumo, "Prediction of residential building energy consumption: A neural network approach," *Energy,* vol. 117, pp. 84-92, 2016.

[113] C. Deb, L. Eang, J. Yang and M. Santamouris, "Forecasting diurnal cooling energy load for institutional buildings using Artificial Neural Networks," *Energy and Buildings,* vol. 121, pp. 284-297, 2016.

[114] C. Fan, F. Xiao and S. Wang, "Development of prediction models for next-day building energy consumption and peak power demand using data mining techniques," *Applied Energy,* vol. 127, pp. 1-10, 2014.

[115] Z. Wang, Y. Wang and R. Srinivasan, "A novel ensemble learning approach to support building energy use prediction," *Energy and Buildings,* vol. 159, pp. 109-122, 2018.

[116] S. Touzani, J. Granderson and S. Fernandes, "Gradient boosting machine for modeling the energy consumption of commercial buildings," *Energy Build,* vol. 158, pp. 1533-1543, 2018.

[117] Y. Guo, J. Wang, H. Chen, G. Li, J. Liu and C. Xu, "Machine learning-based thermal response time ahead energy demand prediction for building heating systems," *Appl Energy,* vol. 221, pp. 16-27, 2018.

[118] Y. Wei, X. Zhang, Y. Shi, L. Xia, S. Pan and J. Wu, "A review of data-driven approaches for prediction and classification of building energy consumption," *Renew Sustain Energy Rev,* vol. 82, pp. 1027-1047, 2018.

[119] C. Molnar, Interpretable Machine Learning: A Guide for Making Black Box Models Explainable, Lulu, 1st edition, March 24, 2019; eBook (GitHub, 2019-06-19), 2018.

[120] R. Team and R. d. c. team, R: A language and environment for statistical computing, Vienna, Austria: R foundation for statistical computing, 2014.

[121] L. Alton, "The 7 Most Important Data Mining Techniques," 2017. [Online]. Available: https://www.datasciencecentral.com/profiles/blogs/the-7-most-important-data-mining-techniques. [Accessed 22 October 2019].

[122] M. Aggarwal and A. Bhatia, "Pattern Discovery Techniques in Online Data Mining," *International Journal of Engineering and Technical Research,* vol. III, p. 28, 2015.

[123] G.-F. Fan, Y.-H. Guo, J.-M. Zheng and W.-C. Hong, "Application of the Weighted K-Nearest Neighbor Algorithm for Short-Term Load Forecasting," *Energies,* vol. 12, no. 5, p. 916, 2019.

[124] N. Altman, "An introduction to kernel and nearest-neighbor nonparametric regression," *Am. Stat.,* vol. 46, p. 175–185, 1992.

[125] G. Bhattacharya, K. Ghosh and A. Chowdhury, "Granger causality driven AHP for feature weighted kNN," *Pattern Recognit,* vol. 66, pp. 425-436, 2017.

[126] C.-X. Nie and F.-T. Song, "Analyzing the stock market based on the structure of kNN network," *Chaos Solitons Fractals,* vol. 113, pp. 148-159, 2018.

[127] S. Madeti and S. Singh, "Modeling of PV system based on experimental data for fault detection using kNN method," *Sol. Energy,* vol. 173, pp. 139-151, 2018.

[128] S. Wazarkar, B. Keshavamurthy and A. Hussain, "Region-based segmentation of social images using soft KNN algorithm," *Procedia Comput. Sci.,* vol. 125, pp. 93-98, 2018.

[129] S. Zhang, D. Cheng, Z. Deng, M. Zong and X. Deng, "A novel kNN algorithm with data-driven k parameter," *Pattern Recognit. Lett.,* vol. 109, pp. 44-54, 2018.

[130] A. Troncoso, J. M. R. Santos and A. G. Expósito, "Electricity market price forecasting based on weighted nearest neighbors techniques.," *IEEE Trans. Power Syst.,* vol. 22, p. 1294–1301, 2007.

[131] Y. Chen and Y. Hao, "A feature weighted support vector machine and K-nearest neighbor algorithm for stock market indices prediction," *Expert Syst. Appl.,* vol. 80, pp. 340-355, 2017.

[132] C. C. Aggarwal, Neural Networks and Deep Learning, NY, USA: Springer International Publishing, 2018.

[133] S. Hayou, A. Doucet and J. Rousseau, "On the Impact of the Activation Function on Deep Neural Networks Training," *arXiv preprint arXiv:1902.06853,* 2019.

[134] A. A. Heidari, H. Faris, S. Mirjalili, I. Aljarah and M. Mafarja, "Ant Lion Optimizer: Theory, Literature Review, and Application in Multi-layer Perceptron Neural Networks," in *Nature-Inspired Optimizers*, Cham, Springer, 2020, pp. 23-49.

[135] V. K. Ojha, A. Abraham and V. Snášel, "Metaheuristic design of feedforward neural networks: A review of two decades of research," *Engineering Applications of Artificial Intelligence,* vol. 60, p. 97–116, 2017.

[136] H. Faris, I. Aljarah, N. Al-Madi and S. Mirjalili, “Optimizing the learning process of feedforward neural networks using lightning search algorithm.,” *International Journal on Artificial Intelligence Tools,* vol. 25, no. 06, p. 1650033, 2016.

[137] L. Athanasiou, D. Fotiadis and L. Michalis, “6. Plaque Characterization Methods Using Computed Tomography,” in *Atherosclerotic Plaque Characterization Methods Based on Coronary Imaging*, London, Academic Press, 2017, pp. 115-129.

[138] M. Hestenes and E. Stiefel, “Methods of conjugate gradients for solving linear systems,” *Journal of research of the National Bureau of Standards,* vol. 49, no. 6, pp. 409-436, 1952.

[139] M. Brilliant and D. Handoko, “Implementation of Data Mining Using Association Rules for Transactional Data Analysis.,” in *In Prosiding International conference on Information Technology and Business (ICITB)*, 2018.

[140] Y. Luo, “Research on Food Consumer Price Index Based on Association Rules and Clustering,” in *2019 5th International Conference on Education Technology, Management and Humanities Science (ETMHS 2019)*, Taiyuan, China, 2019.

[141] X. Yuan, “An improved Apriori algorithm for mining association rules,” in *AIP conference proceedings (Vol. 1820, No. 1, p. 080005)*, Shanghai, 2017.

[142] R. Agrawal and R. Srikant, “Fast algorithms for mining association rules,” in *Proc. 20th int. conf. very large data bases, VLDB (Vol. 1215, pp. 487-499).*, Santiago, Chile, 1994.

[143] L. Szathmary, “Finding frequent closed itemsets with an extended version of the Eclat algorithm,” *Annales Mathematicae et Informaticae,* vol. 48, pp. 75-82, 2018.

[144] J. Han, J. P. and Y. Y., “Mining Frequent Patterns without Candidate Generation,” *ACM sigmod record,* vol. 29, no. 2, pp. 1-12, 2000.

[145] D. Savage, X. Zhang, X. Yu, P. Chou and Q. Wang, “Anomaly detection in online social networks,” *Soc Networks,* vol. 39, pp. 62-70, 2014.

[146] V. Chandola, A. Banerjee and V. Kumar, “Anomaly detection: a survey,” *ACM Comput. Surv.,* vol. 41, no. 3, p. 15, 2009.

[147] C. Aggarwal and P. Yu, “Outlier detection for high dimensional data,” *ACM Sigmod Rec,* vol. 30, no. 2, pp. 37-46, 2001.

[148] Freepik, “Icon made by Freepik from www.flaticon.com,” [Online]. Available: https://www.flaticon.com/free-icon/fish_394730?term=fish&page=1&position=17. [Accessed 24 02 2020].

[149] R. Kaur and S. Singh, “A survey of data mining and social network analysis based anomaly detection techniques,” *Egyptian Informatics Journal,* vol. 17, no. 2, pp. 199-216, 2016.

[150] C. Aggarwal and C. Reddy, in *Data Clustering Algorithms and Applications*, New York, Chapman and Hall/CRC, 2018.

[151] R. Jothi, S. Mohanty and A. Ojha, “DK-means: a deterministic K-means clustering algorithm for gene expression analysis,” *Pattern Anal Applic,* vol. 22, pp. 649-667, 2019.

[152] A. Trevino, "Introduction to K-means Clustering," Oracle, 6 June 2016. [Online]. Available: https://blogs.oracle.com/datascience/introduction-to-k-means-clustering. [Accessed 04 02 2020].

[153] Google Developers, "k-Means Advantages and Disadvantages," Google Developers, 04 Feb. 2020. [Online]. Available: https://developers.google.com/machine-learning/clustering/algorithm/advantages-disadvantages. [Accessed 06 Feb. 2020].

[154] V. Cohen-addad, V. Kanade, F. Mallmann-trenn and C. Mathieu, "Hierarchical Clustering: Objective Functions and Algorithms," *J. ACM,* vol. 66, no. 4, p. 42, 2019.

[155] A. Kassambara, "HIERARCHICAL CLUSTERING IN R: THE ESSENTIALS," 20 Oct. 2018. [Online]. Available: https://www.datanovia.com/en/lessons/agglomerative-hierarchical-clustering/. [Accessed 11 Feb. 2020].

[156] A. Krishna Menon, A. Rajagopalan, B. Sumengen, G. Citovsky, Q. Cao and S. Kumar, "Online Hierarchical Clustering Approximations.," *arXiv preprint arXiv:1909.09667.,* 2019.

[157] Oracle Database Online Documentation 12c Release 1 (12.1), "Data Mining Concepts - Regression," [Online]. Available: https://docs.oracle.com/database/121/DMCON/GUID-51A08CFC-1487-4887-AB47-794C50D67358.htm#DMCON176. [Accessed 26 Feb. 2020].

[158] K. Joshi, Foundations of Discrete Mathematics, Bombay: New Age International Limited, 1989.

[159] L. Flom and D. Cassell, "Stopping stepwise: Why stepwise and similar selection methods are bad, and what you should use," *NorthEast SAS Users Group Inc 20th Annual Conference: 11-14th November 2007; Baltimore, Maryland,* 2007.

[160] E. B. Roecker, "Prediction Error and Its Estimation for Subset-Selected Models," *Technometrics,* vol. 33, no. 4, pp. 459-468, 1991.

[161] A. G. Assaf, M. Tsionas and A. Tasiopoulos, "Diagnosing and correcting the effects of multicollinearity: Bayesian implications of ridge regression," *Tourism Management,* vol. 71, pp. 1-8, 2019.

[162] M. Torabi, S. Hashemi, M. R. Saybani, S. Shamshirband and A. Mosavi, "A Hybrid clustering and classification technique for forecasting short-term energy consumption," *Environmental Progress & Sustainable Energy,* vol. 38, no. 1, pp. 66-76, 2019.

[163] J. Chen, K. Li, H. Rong, K. Bilal, K. Li and P. S.Yu, "A periodicity-based parallel time series prediction algorithm in cloud computing environments," *Information Sciences,* vol. 496, pp. 506-537, 2019.

[164] Microsoft Azure, "What is cloud computin?," Microsoft Azure, 03 03 2020. [Online]. Available: https://azure.microsoft.com/en-in/overview/what-is-cloud-computing/. [Accessed 03 03 2020].

[165] C. Stergiou, K. E. Psannis, B.-G. Kim and B. Gupta, "Secure integration of IoT and Cloud Computing," *Future Generation Computer Systems,* vol. 78, no. 3, pp. 964-975, 2018.

[166] N. Lloret Romero, ""Cloud computing" in library automation: benefits and drawbacks," *The Bottom Line,* vol. 25, no. 3, pp. 110-114, 2012.

[167] D. Müller, “Cloud Computing,” *Datenschutz und Datensicherheit - DuD,* vol. 41, no. 6, p. 371–376, 2017.

[168] A. Gerber, “IBM Developer | 7 key concepts and skills for getting started with IoT,” 2018. [Online]. Available: https://developer.ibm.com/dwblog/2017/just-getting-started-iot-consider-7-key-iot-concepts-skills/. [Accessed 9 October 2018].

[169] L. Atzori and e. al., “The Internet of things: A survey,” *Comput. Netw.,* vol. 54, pp. 2787-2805, 2010.

[170] S. Roy, R. Bose and D. Sarddar, “A fog-based DSS model for driving rule violation monitoring framework on the Internet of things,” *International Journal of Advanced Science and Technology,* vol. 82, pp. 23-32, 2015.

[171] S. Vashi, J. Ram, J. Modi, S. Verma and C. Prakash, “Internet of Things (IoT): A vision, architectural elements, and security issues,” *2017 International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC),* pp. 492-496, 2017.

[172] M. Khan and K. Salah, “IoT security: Review, blockchain solutions, and open challenges,” *Future Generation Computer Systems,* vol. 82, pp. 395-411, 2018.

[173] D. Hardt, The OAuth 2.0 authorization framework (No. RFC 6749)., Microsoft, 2012.

[174] D. Acharjya and N. Ahmed, “Recognizing Attacks in Wireless Sensor Network in View of Internet of Things,” in *Internet of Things: Novel Advances and Envisioned Applications*, Springer, 2017, pp. 149-172.

[175] Daser David, “Medium - A lazy man’s introduction to Multi-Party encryption and decryption.,” 2019. [Online]. Available: https://medium.com/@daser/a-lazy-mans-introduction-to-multi-party-encryption-and-decryption-59f62b8616d8. [Accessed 14 July 2019].

[176] S. Nakrani and C. Tovey, “On Honey Bees and Dynamic Server Allocation in Internet Hosting Centers,” *Adaptive Behavior,* vol. 12, no. 3-4, pp. 223-240, 2004.

[177] X. D. Xue, B. Xu, H. L. Wang and C. P. Jiang, “The basic principle and application of ant colony optimization algorithm,” *Proc. IEEE Conf. Artificial Intelligence and Education (ICAIE),* pp. 358-360, 2010.

[178] A. Jain and R. Kumar, “Scalable load balancing approach for cloud environment,” *International Journal of Engineering and Technology Innovation,* vol. 7, no. 4, pp. 292-307, 2017.

[179] S. J. Fowler, Production-Ready Microservices, O'Reilly Media, Inc., 2016.

[180] H. Paik, M. Barukh, B. Benatallah and A. Natarajan, Web Service Implementation and Composition Techniques, 1 ed., Sydney - Australia: Springer, 2017, pp. 67-91.

[181] AEOTEC GROUP, “Z-Stick Gen5+,” Aeon Labs LLC, 29 10 2020. [Online]. Available: https://aeotec.com/z-wave-usb-stick/. [Accessed 05 11 2020].

[182] AEOTEC GROUP, “Smart Switch 7,” Aeon Labs LLC, 30 08 2109. [Online]. Available: https://aeotec.com/z-wave-plug-in-switch/. [Accessed 05 11 2020].

[183] Arduino LLC, “What is Arduino?,” Arduino LLC, 18 February 2018. [Online]. Available: https://www.arduino.cc/en/Guide/Introduction. [Accessed 06 11 2020].

[184] © ELECTRONICS-LAB.COM, “VIDEO STREAMING SERVER ON ESP32-CAM,” © ELECTRONICS-LAB.COM, 24 01 2020. [Online]. Available:

https://www.electronics-lab.com/project/video-streaming-server-esp32-cam/. [Accessed 06 11 2020].

[185] Arduino, “Sketch,” Arduino, [Online]. Available: https://www.arduino.cc/en/Tutorial/Sketch. [Accessed 07 11 2020].

[186] Arduino, “Arduino IDE 1.8.13,” Arduino, 07 10 2020. [Online]. Available: https://www.arduino.cc/en/software. [Accessed 07 11 2020].

[187] Martínez-Plumed, F. a. Contreras-Ochando, L. a. Ferri, C. a. H. Orallo, J. a. Kull, M. a. Lachiche, N. a. R. Quintana, M. J. a. Flach and P. A., “CRISP-DM Twenty Years Later: From Data Mining Processes to Data Science Trajectories,” *IEEE Transactions on Knowledge and Data Engineering,* pp. 1-1, 2019.

[188] J. Demsar, T. Curk, A. Erjavec , C. Gorup , T. Hocevar , M. Milutinovic , M. Mozina , M. Polajnar , M. Toplak , A. Staric , M. Stajdohar , L. Umek , L. Zagar , J. Zbontar , M. Zitnik and B. Zupan , “Orange: Data Mining Toolbox in Python,” *Journal of Machine Learning Research,* vol. 14, pp. 2349-2353, 2013.

[189] M. S. Brown, Data Mining for Dummies, New Jersey: John Wiley & Sons, 2014.

[190] M. K. M. Shapi, N. A. Ramli and L. J. Awalin, “Energy consumption prediction by using machine learning for smart building: Case study in Malaysia,” *Developments in the Built Environment,* vol. 5, p. 100037, 2021.

[191] T.-Y. Kim and S.-B. Cho, “Predicting residential energy consumption using CNN-LSTM neural networks,” *Energy,* vol. 182, pp. 72-81, 2019.

[192] M. A. Sulaiman, “Evaluating Data Mining Classification Methods Performance in Internet of Things Applications,” *Journal of Soft Computing and Data Mining,* vol. 1, no. 2, pp. 11-25, 2020.

[193] A. Al-Adaileh and S. Khaddaj, “Reduction of Energy Consumption Using Smart Home Techniques in the Household Sector,” *International Journal of Energy and Environmental Engineering,* vol. 14, no. 12, 2020.

[194] PAT RESEARCH, “predictiveanalyticstoday.com - What is Predictive Analytics?,” 2019. [Online]. Available: https://www.predictiveanalyticstoday.com/what-is-predictive-analytics/. [Accessed 17 July 2019].

[195] Department for Environment Food & Rural Affairs, “Department for Environment, Food & Rural Affairs,” GOV.UK, 30 Jan. 2020. [Online]. Available: https://www.gov.uk/government/organisations/department-for-environment-food-rural-affairs. [Accessed 02 Feb. 2020].

[196] TG, “IBM marketing experts predict the 10 key marketing trends for 2017,” 3 Feb. 2017. [Online]. Available: https://totallygaming.com/eventblog/ice-live/ibm-marketing-experts-predict-10-key-marketing-trends-2017. [Accessed 18 Feb. 2020].

[197] S. Huh, S. Cho and S. Kim, “"Managing IoT devices using blockchain platform,” *19th International Conference on Advanced Communication Technology (ICACT),* pp. 464-467, 2017.

[198] A. Noureddine, R. Rouvoy and L. Seinturier, “A Review of Middleware Approaches for Energy,” *Software: Practice and Experience,* vol. 43, no. 9, pp. 1071-1100, 2013.

[199] T. Xiao, S. Li, B. Wang, L. Lin and X. Wang, “Joint Detection and Identification Feature Learning for Person Search,” *arXiv:1604.01850v3,* vol. Volume, no. Issue, p. Pages, 2017.

[200] Cambridge University, “Cambridge Dictionary,” Cambridge university press, 31 01 2021. [Online]. Available: https://dictionary.cambridge.org/dictionary/english/tumble-dryer. [Accessed 31 01 2021].

[201] G. Arul Freeda Vinodhini and J. Anitha, “General Service Time Distribution of Fuzzy Queue Using Parametric Non - Linear Programming,” *Annals of the Romanian Society for Cell Biology,* vol. 25, no. 4, p. 2818, 2021.

[202] J. Dem\v{s}ar, T. Curk, A. Erjavec, \. Gorup, T. Ho\v{c}evar, M. Milutinovi\v{c}, M. Mo\v{z}ina, M. Polajnar, M. Toplak, A. Stari\v{c}, M. \v{S}tajdohar, L. Umek, L. \v{Z}agar, J. \v{Z}bontar, M. \v{Z}itnik and Bla, “Orange: Data Mining Toolbox in Python,” *Journal of Machine Learning Research,* vol. 14, pp. 2349-2353, 2013.

[203] J. Demšar, T. Curk, A. Erjavec, Č. Gorup, T. Hočevar, M. Milutinovič, M. Možina, M. Polajnar, M. Toplak, A. Starič, M. Štajdohar, L. Umek, L. Žagar, J. Žbontar, M. Žitnik and B. Zupan, “Orange: Data Mining Toolbox in Python,” *Journal of Machine Learning Research,* vol. 14, pp. 2349-2353, 2013.

[204] STIEBEL ELTRON, “SNU 10 SL,” 24 January 2022. [Online]. Available: https://www.stiebel-eltron.de/de/home/produkte-loesungen/warmwasser/klein-_wand-_undstandspeicher/kleinspeicher_5_bis15l/snu-sl/snu-10-sl.html. [Accessed 24 January 2022].