A User-guided Personalization Methodology for New Smart Homes

S M Murad Ali

M00510260

Supervised by Prof. Juan Carlos Augusto

Co-supervised by Prof. David Windridge

Department of Computer Science

Faculty of Science

Middlesex University

October, 2022

A thesis submitted to Middlesex University in partial fulfilment of the requirements for the degree of PhD in Computer Science.

Acknowledgements

In the name of Allah, the most gracious and the merciful. First and foremost, I am thankful to Almighty ALLAH for giving me the strength, knowledge, ability and opportunity to undertake this study and complete it satisfactorily.

Secondly, my deep gratitude goes to my supervisor Professor Juan Carlos Augusto for his continuous patience, motivation, and support (both academically and non-academically). His vast knowledge of this research area has played a key role in my ability to successfully complete this thesis. I am also thankful to my co-supervisor Dr. David Windridge for his guidance, advise, patience, support and mentorship.

In addition, I want to thank Dr. Emma Ward, Department of Psychology, Middlesex University, for introducing me to several companies who assisted in evaluating the project. Thank you also to Dr. Emilio Serrano, Department of Artificial Intelligence, Technical University of Madrid, who taught me how to use the simulation for this research.

To the colleagues and lab-mates, I encountered throughout this process—especially Dr. Mario, Dr. Leo, Gines, and Nawa—with whom I had the utmost privilege of sharing thoughts, ideas, experiences: THANK YOU.

I am deeply grateful to my parents for their support, appreciation, encouragement and a keen interest in my academic achievements. Thank you to my beloved daughter, Ayat, for being a good girl when I wrote my thesis. Finally, really thank my wife for her patience, tolerance and moral support throughout the research work.

Abstract

Recently, smart homes have become a centre of attention due to their provision of an enhanced quality of life via automation services with-in the homes. Smart homes technology, however, must learn and adapt in accordance with the habits of their residents in order to provide the relevant services.

Human activity recognition is a well-known technique used to understand user behaviours and enables the smart home services to run automatically according to the human mind. Observing the pattern of resident's daily tasks is a useful technique used by the researchers to develop more user-centric personalised services for the occupant. There are several approaches to human activity recognition in case of smart home. Among the most popular ones is the data-driven approach, which provides promising results due to its advancement of machine learning. Regardless of this, several drawbacks such as limited availability of data in the initial phases can be a hurdle in providing smart home services. The purpose of this thesis is to introduce an approach known as useR-guided nEw smart home ADaptation sYstem (READY) for the development of a personalised automation system which provides users with smart home services when they move into their new house. The READY approach integrates several approaches, leverages user feedback, and builds a rich data set that helps the house recognise the user's daily activities and provide personalised smart home services, accordingly. The system development process was strongly user-centred, where the user was involved in every part of the development to receive fine-grained services from the outlet. Additionally, the research introduced a supplementary method along with READY called User-guided Transfer Learning (UTL) that leverages the existing smart home data set in order to enhance the overall automation functionality and effectiveness.

The approaches presented in this thesis have been tested and validated at Middlesex University Smart Lab by a group of internal and external participants. The results show that 100% of these participants believed that the READY method provides personalized services to the new smart home

from outset through user involvement. Moreover, the UTL approach detects new services, and this increases the acceptability of the new smart home. The results of the given approaches prove to be a significant advancement in the domain of smart home technology and become a positive step toward bridging the gap between the new smart home and incoming residents.

Contents

List of Figures

List of Tables

List of Abbreviations

UTL User-guided Transfer Learning iii
READY useR-guided nEw smart home ADaptation sYstem iii
WHO World Health Organization 3
AI Artificial Intelligence 3
HMM Hidden Markov Model 13
SVM Support Vector Machines 14
ANN Artificial Neural Networks 14
AR Activity Recognition 14
AD Activity Discovery 14
HHTL Home-to-Home Transfer Learning 23
FSR Feature-Space Remapping 24
GAFSR Genetic Algorithm for Feature-Space Remapping 24
GrFSR Greedy search for Feature- Space Remapping 24
AMM Activity Mapping Method 24
MHTL Multi-home Transfer Learning 24
ADL Daily Living Activities 1
COATI COntext-Aware systems Testing validation 8
U-CIEDP User-Centred Intelligent Environments Development Process . 29

1

Introduction

Smart homes [84, 21, 69] are houses that are typically equipped with a range of technological advancements that have the capacity to learn and adapt their services according to their inhabitant's behaviours. In turn, this results in customized services and provides their inhabitants a better quality of life. A smart home system comprises several interconnected sensors and household devices that utilize reciprocal interactions to automate daily housework and activities with or without resident intervention in a safer, cheaper, more convenient and efficient manner. Interest in smart homes has increased due to the potential they offer regarding reducing stress and increasing amusement in daily life, which can make a positive contribution to health and well-being.

However, one problem that has been encountered in smart homes by inhabitants is that it takes a considerable time for automation services to actually be delivered, despite the home being equipped with the necessary technology. This concept is explained in more detail in Chapter 1.1. Automating the smart home effectively requires knowledge of the user's habits. Human activity recognition [28] is a powerful method of detecting user habits from a Daily Living Activities (ADL) dataset (see for example: [74]). This has led to activity recognition in the smart home becoming a popular research area. Section 2.1 highlights state-of-the-art activity recognition. Researchers have introduced several types of research in order to solve the problem of activity recognition in smart homes (see Chapter 2). This research explores two main categories of approaches: data-driven and knowledge-driven [46, 70]. Data-driven [78] strategies require a large number of sensor datasets to learn user activities. By contrast, knowledge-driven approaches use prior domain knowledge to build the activity model.

It must be acknowledged that both approaches have advantages and disadvantages. For instance, the knowledge–driven approach does not have the capability of adapting to users' preferences. On the other hand, the data-driven approach does have this ability, as it uses machine learning techniques to model user activities. The machine learning model does this through a dynamic character that is automatically adapted by referencing the user's daily activity data. Presently, the superior adaptive capability and comparatively cheaper data collection and storage opportunities make the data–driven approach more popular. However, a major drawback is that a data-driven home cannot provide smart home functionalities immediately due to a lack of data to inform its model. This phenomenon is known as the "cold start" problem and is considered to be the main obstacle for data-driven smart homes being ready to function according to user preferences as soon as the user starts living in the house. Unfortunately, we have not seen an acceptable, efficient solution to this problem. Therefore, we wonder whether we can create a state-of-the-art approach that can help people start living in the new smart home to mitigate the "cold start" problem.

The remainder of this chapter is structured as follows: Section 1.1 explains the motivation for the research; Section 1.2 introduces the problem statement; Then, Section 1.4 formulates the aims and objectives of the work; and lastly, Section 1.5 provides an outline of the remainder of the dissertation.

1.1 | Context and Motivation

"I'd rather die than be a burden on my daughter - like many old people". Elderly people often embrace this sentiment [54] because they do not want to be viewed as a burden. Although many elderly people desire to live independently, their families may be hesitant and worried, regardless of how well-managed the home may be. Living alone can be dangerous for elderly people for many reasons. Smart homes and associated conveniences, however, could help improve their lives and mitigate such concerns.

Health and medical advancements have contributed to a rapid increase in the number of older people worldwide. According to a United Nation's (UN) report, one out of eight people world- wide were at or over the age of 60 in 2015 [3]. The same report also suggested that this percentage will double by 2050. In addition, Europe's elderly population is rapidly growing, with the World Health Organization (WHO) projecting that 30% of Europe's population will be 65 years-old by the year 2050 [49].

Even as people age, it is common to find that older people prefer to remain independent in their homes. This presents an increased demand for smart home environments that can provide these individuals with a sense of comfort, safety, and security. Lutolf [69] formalized the smart home concept, focusing on the integration of different services into the home using communication systems. According to Satpathy [89], integrating different devices can help users live independently and comfortably. Augusto and Nugent [21] then brought the smart home concept to the software-oriented Artificial Intelligence (AI) community, building a bridge between AI and smart homes by highlighting the need to build homes that can recognize user behaviors and assist users with daily tasks. Recently, Leitner [66] introduced a new paradigm—the wise home—offering an improved user experience by focusing more on the user's interaction experience (both explicit and implicit) than the technology that makes it possible. Currently, the extant research on smart homes suggests that integrating a variety of machine learning methods and artificial intelligence techniques can make smart homes more user-friendly.

As a thought experiment, imagine a scenario where a user, say Bob, is experiencing the early stages of dementia but is adamant about continuing to live independently. Despite concerns about safety and proper care, his family decides to move Bob to a new smart home where his daily living activities, such as personal hygiene and food preparation, will be facilitated by technology. The same home will also provide advanced functionality such as fall detection, as well as other safety and security measures.

In this scenario, a critical question arises: Will the chosen technology be able to provide Bob with the help he needs immediately after he moves into his new home? For smart homes that rely on large amounts of user data as inputs, the

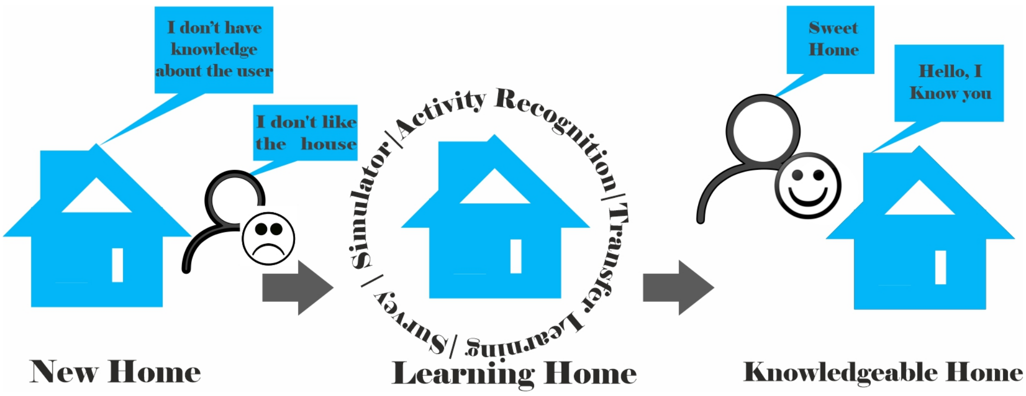


Figure 1.1: Conceptual illustration of the research.

answer to this question would be "no". A smart home typically needs copious amounts of data to recognize, understand and predict a user's behaviour and provide the required services [44]. This data dependency might give Bob's family cause for concern about his independent living, especially when he first moves in, leading them to abandon the idea despite the smart home's long-term capabilities.

This thought experiment illustrates one of the core challenges of readying new smart homes for new users—activity recognition. Activity recognition is the mechanism through which smart homes infer which services their users need by recognizing and classifying user behaviours, such as Bob's cooking, bathing, and sleeping. To do so effectively, smart homes need massive amounts of data that can be applied to the user. However, such data is not usually available when a user is just moving in, as the smart home has not had an opportunity to observe its occupant's behaviour. This paradox, otherwise known as the "cold start" problem, is what this thesis seeks to solve.

1.2 | Problem statement

In spite of the clear strengths of data- and knowledge-driven approaches, both methods suffer from key drawbacks that can make them unattractive to users and developers alike. For instance, the strength of the data-driven approach is its ability to draw on massive amounts of data from the user's own behaviour within the smart home. However, a newly developed smart home will not have access to such data at the outset. Therefore, as mentioned, the new smart home will face what is called the "cold start" problem. In essence, homes utilizing a data-driven approach require a considerable amount of time to collect enough data to model the activity of the user. During this intervening time, the occupant may abstain from the smart home, which may further exacerbate the need for user data.

On the other hand, knowledge-driven smart homes do not require user data but instead demand a contextual knowledge of daily activities usually acquired through standard knowledge engineering approaches [31]. For these types of smart homes, different approaches, such as logic-based approaches, logical formalisms [43], event calculus [43] and lattice theory [31], can be applied to represent the activity recognition models. Another approach called ontology activity modelling, which is closer to the logical approach, uses a description logic based on mark-up language [39]. However, unlike data-driven approaches, knowledge-driven methods cannot handle uncertainty. There are other reasoning techniques, such as fuzzy logic and probabilistic reasoning, but even these are not integrated with modelling techniques [55].

To the best of the author's knowledge, few scholarly works have addressed new smart home adaptation processes using data-driven approaches. One such adaptation process, for instance, the transfer learning method, which uses pre-existing smart home data to recognize user behaviour, is rarely used. However, even in such cases, the accuracy rate is very low if no data is available from the new smart home [41, 42]. Therefore, relying on transfer learning alone will fail to satisfactorily address the "cold start" problem.

As a result, data simulation tools that create synthetic data have become extremely popular, especially for assessing new models before they are implemented in smart homes [67, 64, 90]. In the context of solving the "cold start" problem, an approach proposed by Azkune et al. [22] is notable. The proposed hybrid methodology combines synthetic data with data from real user daily activity surveys as inputs for its simulation tools. The key idea is to distribute the survey among target users with the aim of learning how users perform their daily activities in the smart home. The survey data is then processed by synthetic data generator tools for an arbitrary number of days to generate a labelled activity dataset. A key step missing from this approach is an evaluation of the generated synthetic data. Thus, the created dataset is used only for modelling and recognizing user activity within a smart home. If the dataset turns out not to be applicable to the user, the process does not suggest any alternatives.

To address both the "cold start" problem and the lack of personalisation inherent in knowledge-driven approaches, the current research introduces a personalized data-driven smart home system that is rich in data from the moment a user moves into their new home. The process begins by collecting data from the user with a survey [10] on the user's daily activities. This data helps provide insight into how to perform several fixed activities (e.g., making tea) in terms of used objects (e.g., tea kettle) as well as required time. The simulation, designed based on these survey responses, then generates synthetic data. Because the data might not have sufficient knowledge to provide accurate home services to the user from the survey alone, the User-guided Transfer Learning (UTL) approach offers a complementary approach that validates the knowledge extracted by the simulation dataset.

1.3 | Research questions

In light of these existing problems and limitations, this thesis aims to answer the following research questions:

How can a data-driven smart home provide automation services to the occupant from the outset when the house does not have any data to understand the user's habits?

- Q1 How can the collected user's daily activities assist the system adaptation?
- Q2 How can it help developers understand the user's routines and preferences?
- Q3 How to process the acquired data and transfer it to the new home?
- Q4 How can make the current transfer learning techniques more user-tailored?

1.4 | Research aims and objectives

The primary goal of this research was to develop a smart home system that is ready to provide personalized smart home services as soon as the user moves into a new house, while leveraging existing smart home datasets to enhance overall automation functionality and effectiveness. The aims of the research project and the objectives are listed below:

A1 To identify how the user performs particular activities of daily living.

- **A1-O1** To design a questionnaire to collect the daily living information of the user.
- **A1-O2** To analysis survey responses to facilitate the developer in configuring the simulator.

A2 To identify a method for simulating the user's daily activities in the virtual house based on a user's survey responses and to use that method to generate synthetic data.

- **A2-O1** To create the virtual house and simulate user behaviour based on a user's survey responses.
- **A2-O2** To generate the synthetic dataset.

A3 To use the synthetic data to examine the system providing the required automation services, beginning on the day the user moves into the house.

- **A3-O1** To identify the automation system on which to test the synthetic dataset.
- **A3-O2** To determine through user feedback whether the system is able to provide automation facilities beginning on the day the user moves into the house.

A4 To determine the differences between rules generated by the synthetic and pre-existing smart home datasets and to use these differences to fine-tune the rules for the new smart home's user assistant.

A4-O1 To generate rules from the synthetic and pre-existing smart home datasets.

A4-O2 To fine-tune the rules for the new smart home's user assistant.

1.5 | Thesis structure

This section outlines the structure of the next chapters of the thesis.

Chapter 2 aims to find the related work that has been done for smart home adaptation. Literature review emphasizes user-centric, simulation, activity recognition, and transfer learning approaches for smart home adaptation. After identifying any gaps in these areas, we will then examine any previous work in this topic. This section certifies the novelty of the project.

The Methodology to be used for the project is described in Chapter 3. The user-centric approach considered for the project aims to keep the user at the centre of the development process, thereby ensuring a higher chance of system acceptance.

READY is a method to integrate three systems together and build a new system that provides a user with smart home services as soon as the user starts living in the home. The development process begins with gathering insights into user behaviour as they perform their daily activities by interview. Later, each of the components will be added based on the requirements and a tailored system will be developed as per the user's requirements. The evaluation of READY is an iterative process. Each component iteration added to the system is validated by the user before proceeding to the next one. Chapter 4 explains more details of the READY method.

The complete system testing appears in Chapter 5, where details of the testing house and participants are explained. As mentioned previously, each unit of the system is individually developed and validated. However, in Chapter 5, the system is tested as a whole using COntext-Aware systems Testing validation (COATI) [16] approach to gather insights from the different users. The purpose of this step is to identify potential issues and benefits that may not be evident when the entire system is tested by different users. Details of the testing process, results and the feedback from participants will be provided in this section. The research was validated by the professional participants are explained in Section 5.2. The outcomes of the project and the challenges faced are explained in Chapter 6. Finally, Chapter 7 describe the conclusions and suggest future directions to ensure continued research in this area.

2

Literature Overview

Smart home adaptation is a vast area of research that can be examined from a number of different perspectives. The aim of this chapter is to discuss the extant research in this area and describe in detail how the approaches presented in this thesis relate to those that current exist.

Section 2.1 provides an overview of data-driven activity recognition in smart homes. Then, Section 2.2 presents the most up-to-date approaches to developing a user-centric smart home. Recent advancements in transfer learning are highlighted in Section 2.3. Finally, Section 3.2 critically analysis the major insights gleaned from the available research as well as notable gaps that remain.

2.1 | Activity recognition in smart homes

The baseline infrastructure for a home to be considered a smart home includes several sensors and actuators, user interfaces (such as voice control and graphic displays), building services (ventilation, heating and lights), and appliance networks [93]. The external network (mobile phones, internet) can be combined with the in-house network. In this context, a smart home is focused on an automated building as well as on integrated communication services via existing building infrastructure. Researchers generally agree that a smart home system is comprised of three primary elements: the internal network, home automation, and intelligent control [79].

To enable activity recognition in a smart home, there are three main stages. It is necessary to 1. Collect low-level data from the sensors (acquiring sensor data), 2. Process the data collected (processing and data analysis), and 3. Apply learning or reasoning methods to make inferences about activities based on the processed data (activity recognition)(Figure 2.1) [68, 91, 74, 71, 5].

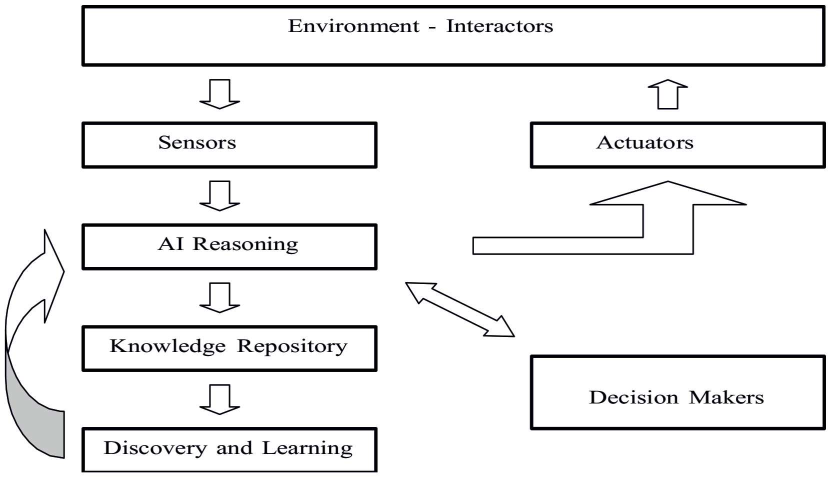


Figure 2.1: Information flow in smart home [20].

Step 1: Acquiring sensor data

The first stage of data collection requires the use of sensors and actuators, small and affordable devices around the household capable of perceiving, monitoring and logging human activities [28]. While a wide range of sensors is available, few of those on the market are specifically created for activity recognition [28]. Sensors can be categorized based on their type, purpose, output signals, and technical infrastructure.

Dense sensors can be split into two main categories: obtrusive, or vision-based sensors, and non-obtrusive sensors [34]. Vision-based sensors use video cameras for activity recognition; they are popular for this purpose because the sensing device does not require human intervention. Non-obtrusive sensors can be further divided into two classes: wearable sensors and dense-sensing environment sensors [94].

Wearable sensors are suitable for applications such as monitoring skin temperature, pulse, body position and movement. However, they have a number of limitations, notably short battery life due to their constant operation, and willingness of the user to wear them. To address these limitations, dense sensing has emerged, where the dense-sensing environment can collect all the necessary data without physical contact with the user.

Smart home applications continuously generate data, with the amount produced depending on the number of sensors, the number of occupants of the home, and the activities the occupants carry out. All sensors rely on wired or wireless communication and must have a Unique Identifier (ID), time stamps and status signals [28]. Sensors may fail from time to time, especially if the smart home environment is noisy . The data collected can be noisy and multidimensional; thus, temporal-ordered random data processing is used to isolate the necessary raw data for activity recognition.

Step 2: Processing and data analysis

Data analysis is a critical step for activity recognition in smart homes, mostly performed by machine learning algorithms and reasoning approaches [91, 5]. However, as future smart homes become more and more sophisticated, the data collected also becomes noisier, requiring additional processing before being subjected to next-level analysis. Data filtering becomes essential to smooth out the raw data by filtering out artefacts and removing outliers. Methods such as Bayes and particle filters, median filters, low-pass filters and Kalman filters can be used.

After filtering and handling missing values, the next step in data analysis is to put the data into a proper format depending on the algorithms used. Data segmentation is also necessary prior to classification, since smart home sensor data might be collected at requested or periodic intervals. The segmentation process divides collected data into smaller blocks before applying classification to improve classification performance. A smart home data segmentation process proposed by Ni et al. [74] divides data into three main segments: temporal-based, activity-based and sensor-based [74].

Step 3: Activity recognition

The last stage of the process outlined in (Figure 2.1) is recognition of an activity based on the acquired and filtered sensor data. There are two main approaches for activity recognition: knowledge-driven [46, 70] and data-driven [78]. Knowledge-driven methods use prior domain knowledge to model current activities [74, 28], and involve knowledge acquisition, formal modelling and knowledge presentation. Logical reasoning tasks such as deduction, induction and abduction are used for activity recognition or prediction in knowledge-driven models. Their design is semantically clear, logically elegant and easy for the user to apply immediately; it is well known that they provide a solution to the "cold-start" problem [22]. However, these models are a static method, weak in handling uncertainty and temporal information.

In contrast with knowledge-driven models, data-driven models learn from pre-existing datasets that contain user behaviours by utilizing data mining and machine learning techniques. The models involve the use of probabilistic or statistical methods for overcoming data uncertainty and temporal issues. Based on the categorization proposed by Jebara [62], data-driven approaches can be separated into two classes: generative and discriminative.

In the generative approaches, a probabilistic model is used to build a complete description of the input data [65]. For example, the naive Bayes classifier [88] provides adequate results for activity recognition since it incorporates probability concepts. Discriminative approaches use previous submissions for assembling correct and in-correct data. For instance, the nearest-neighbour algorithm utilizes a large number of training samples which grow exponentially with the anticipated accuracy. Another popular generative approach is the Hidden Markov Model (HMM), which handles temporal information well [72]. The model uses a probabilistic structure for efficient learning from the available data. Its main drawback is that a complete probabilistic representation requires adequate data.

There is also a need for discriminative approaches which solve the classification problem rather than the representation problem, like the generative approach. An example of the discriminative approach is the nearest-neighbour method [30] that compares the training dataset and allocates the most closely matched sequences together.

Decision trees are another example of a discriminative technique. Decision trees are used to learn logical descriptions of activities from complex sensor readings [92, 75]. Many available descriptive approaches classify activities based on decision boundaries, where the main challenge is to find the hard data points (those closest to the boundary). These data points, known as support vectors, are used in the well-known Support Vector Machines (SVM) machine learning technique [33]. SVMs are established and well-known classification methods that classify data in a non-probabilistic way. Other popular algorithm types include Artificial Neural Networks (ANN) which offer various advantages for both activity recognition and learning process in smart home applications [71]. Popular ANN applications in deep learning include recurrent neural networks, deep feed-forward networks and convolutional neural networks. These algorithms perform better than SVM, NB , and HMM [15, 53].

There are some other approaches that do not fall clearly into the categories of discriminative and generative. For instance, the Independent LifeStyle Assistant (ILSA) uses rule-based and statistical models [51]. The Learning Frequent Pattern of User Behaviour System (LFPUBS) also uses rules of association to find the most frequent patterns and distil and implement event condition action rules to detect patterns in real time [27, 24].

Bakar et al. [28] classified activity models into supervised Activity Recognition (AR) and unsupervised Activity Discovery (AD) based on the data instance. Supervised AR follows a supervised learning approach where labelled training data is available for activity classification. For example, decision trees, neural networks and support vector machine models are in the AR domain.

Unsupervised AD entails data flow analysis for discovering the most frequent patterns or knowledge through unsupervised approaches. The data can be represented using rules and, as mentioned above, LFPUBS is an example that falls within this category. There are also some approaches that could be classified as both AR and AD.

For instance, Bourobou and Yoo [32] proposed a method that first uses an unsupervised learning method, the K-pattern clustering algorithm, to detect discontinuous and interleaved user activity patterns and group them into appropriate clusters. In the next step, an ANN is used to recognize and predict user activity based on Hamblin's and Allen's interval-based temporal relations [18].

Activity Discovery (AD) entails data flow analysis for discovering the most frequent patterns or knowledge through unsupervised approaches. The data can be represented using rules and, as mentioned above, LFPUBS is the example of this category. There are also some approaches that could be classified as both AR and AD. For instance, Bourobou and Yoo [32] proposed a method that uses firstly an unsupervised learning method called the K-pattern clustering algorithm for detecting discontinuous and interleaved user activity patterns and groups them into appropriate clusters. In the next step, an ANN is used to recognize and predict user activity based on Hamblin's and Allen's interval-based temporal relations [18].

In summary, current smart home research focuses on learning algorithms and reasoning approaches [74, 28]. There are many studies devoted to user-centred approaches for data-driven smart home development, and it will be beneficial to compare user-centric smart homes against non-user-centred (Table 2.1).

2.2 | User-centred smart home

It is well-reported in the current literature that one of the main current drivers for smart home applications is to fulfil the desires of the elderly people who want to continue to live independently [74, 6]. Throughout the literature related to user-centred smart homes, there are a number of terminology ambiguities. Terms such as smart home (SH) and Ambient Assisted Living (AAL) are often found together, since smart homes have been a core instrument of AAL.

Table 2.1: Comparison between user-centred and non-user-centred smart homes.

User-centred smart home	Non-user-centred smart home
Users are more involved during system development. Hence, users feel a sense of ownership of the house.	Focuses more on the user generated data rather than direct user involvement.
The adaptation process starts from the development period.	The adaptation process starts when the user starts living in the house.
There are fewer chance to redesign the home because every design step only concludes after user acceptance.	May require redesign when the user starts living in the home.
User contribution is directly used in the validation process. Thus, there is only a small chance the user will refuse the house.	The design model is usually first validated by user data, and then directly by the user. There are more chances for the user to refuse the house.
User-centred design is time-consuming and costly because of intensive stakeholder involvement. Validation with the user requires more time.	Comparatively less time consuming and cheaper.
The final product is more effective and safe, especially for vulnerable users.	Product is less effective and safe.

The existing literature highlights that most approaches to creating user-centric smart homes focus on providing facilities for the elderly and disabled people. However, there are some user-centric approaches that focus on other areas, such as smart home power consumption [40]. Another recent example is the PersOnalised Smart Environment to increase Inclusion of people with DOwn's syNdrome (POSEIDON) system, in which a user-centric intelligent environment development process is implemented during the design of the system [20].

Based on the current scholarly work, a user-centric architecture implies two things. Firstly, it concentrates on the interface design between the IT application and the individual users [36, 77, 37]. Secondly, it entails development of scenarios and their evaluation concerning the prospects of IT usage by inviting prospective user participation and controls [13, 29]. The current smart home study highlights potentials and scenarios for IT application in residential buildings through smart home development.

A data-driven approach mainly concentrates on designing a system that works smoothly without user interruption [61]. These approaches may sometimes not work because of complex and irregular human behaviours [27]. For overcoming such cases, there is a need for a system design where users can incorporate their comments or opinions into the system. Such incorporation will allow the system to take more accurate decisions and provide maximum utility to the user.

Today, new smart home designs are mainly driven by technology rather than user needs. In very limited cases, the user is engaged in the development of the system. Since many smart homes are designed for elderly people, developers may find it difficult to find potential users for testing during systems development. There is a lack of knowledge from the developer's side with respect to engaging potential users during system development, which explains the lack of such involvement [13].

Time and stakeholders' constraints, along with financial limitations, are some of the reasons for neglecting to integrate user feedback during smart home design [50]. Knowledge that comes from user activity is used to improve the system during its development process. In a user-centred approach, the user is directly involved with the development process, and may feel the final product is more convenient and secure after taking part in the evaluation and validation process.

According to ISO [1], the user-centred design aims at building a system that meets all user requirements and is highly usable. During a user-centred design process, the users are fully involved in the process of designing the system, through interviews and other feedback, providing suggestions until all user requirements are satisfied.

In some cases where elderly users are involved, devices known to the users are preferable, because unfamiliar ones can cause anxiety for the occupants [13]. User involvement also improves the adaptation process, especially in a data-driven smart home where there is a great lack of user-centred smart home design [79, 83].

Amiribesheli and Bouchachia [13] proposed a user-centred scenario-based approach to develop smart homes for dementia patients. Their approach was created based on existing literature and collaboration with caregivers. They highlighted that stakeholders involved in user-centric systems should participate in every stage of the design process, and recommended collecting relevant information from stakeholders around the dementia patient as well as directly from the patient for better generalisation of the system [13].

Hussein et al. [58] proposed a self-adaptive smart home prototype for disabled people. Two types of neural networks were used in the prototype: feed-forward and recurrent. The proposed system uses a recurrent neural network for acquiring human behaviour patterns, and then the habits and activities are learned to predicting human activity and recommend actions on behalf of the user. A feed-forward architecture is then applied to integrate safety and security system applications within the smart home. The prototype also allows users to reduce power waste by evaluating and adapting consumer behaviour patterns [58].

Research has demonstrated that integrating continuous Ambient Intelligence (AmI) technologies, including sensor networks, pervasive computing and wearable devices in a user-centred design significantly improves the degree of user acceptance of intelligent systems. Casas et al. [35] argued for the importance of user modelling during user-centric smart home development since different users have different needs. However, developing a unique system for individual needs is costly and impractical; thus, Casas and colleagues proposed a user modelling technique for creating an accurate, parameterized profile for the individual user to enable the system to change its parameters for new users or adjust them for existing users upon request. The system profiles users, taking into account cognitive and sensorial disabilities [35].

Hwang and Hoey [59] presented the research gap found between current technology and end-users. To address this gap, a proposed person-specific knowledge base of user needs that connects the user with the medical professional, family member, product developer and all stakeholders is vital [59]. The work of Hwang and Hoey involved feedback from caregivers providing adult and elderly care, and information from complex smart homes that are able to sense surroundings and provide assistance. The main challenge was to develop an intervention (prompts) and sensing mechanism delivered at the appropriate time, since the system should understand the type of intervention necessary, as well as recognize changes in the user's ability and adapt the intervention accordingly.

Haines et al. [52] proposed that a user-centred approach improves the home system interface design and enables assistance to people with a wide range of characteristics and abilities. Additionally, the authors state that a prototype system can be designed and tested either in a laboratory setting or field trials to identify potential problems and solve them based on user feedback [52].

Ravishankar et al. [83] presented an approach for identifying technical and design issues during the designing, developing, and testing phase by using functional assessment systems. They conducted case studies to explore the deployment of the smart home systems interfaces and systems geared towards evaluating instrumental activities of daily living and activities of daily living. Several interesting challenges were identified, including connection failures between sensor and receiver. The male and female participants showed different responses to problems; where the female participants wanted to share problems with their families, the male participants denied any problems existed. After examining the pre- and post-interview results, the authors emphasized the importance of user experience related to independence and/or privacy needs and the necessity for adaptation or customization based on individual needs.

Other types of user-centric smart home approaches that are becoming popular are virtual smart home prototypes designed by user interaction through context-aware criteria [67, 64]. There are different approaches for designing such virtual smart home applications. 3D [67, 64] and 2D [64] virtual environments, for instance, can be created of full smart home facilities. The mouse pointer is dragged on an avatar (user) throughout the virtual environment to gather positioning data. In another virtual prototype design [63], the user performs some real-time interactions through visualization to develop a better understanding of the smart home. The user's feedback is used by the designers to improve the smart environment. Free smart home simulators such as UbikSim [90] and

OpenSHS [12] were used by researchers to import smart devices into the environment, allowing simulation of user-specific events and generation of synthetic data.

We aware that some smart home approaches focus on fulfilling the user's complex needs [86, 85] and embed sensors, audio, video or other technologies. However, this research focuses on diminishing the cold start problem of the data driven smart home with standard setup. A summary of user-centred approaches for smart home development is given in Table 2.2.

Table 2.2: Summary of the user-centred approach for smart home development.

References	User	Contributor	Interface	Design approaches
[13]	Dementia patient	User, Carer, Dementia specialist	Microphone, visual display	Scenario based approaches.
[59]	Older adult	Older adult, Carer	2D,3D inter-face	To design a system to reduce gap between the different type of stakeholder.
[83]	Older adult	Older adult, Interviewer	Face-to-face interview and feedback	User sanctification measurement prior and post activity feedback.
[58]	Disabled patient	User, responsible authority	Computer, tablet, microphone, TV	Integrating different types of neural networks to design self-adaptive smart homes.
[35]	Elderly People with disabilities	User, Carer	Haptic interface, voice controller	To design a system that understands the capability of disabled people.

Based on the above discussion, it seems that the user-centred approach is complicated and time-consuming due to the iterative refinement process. In practise, though, the approach has a lot of benefits. We also have seen there are some approaches that allow users to indicate their preferences, like the LFPUBS system [27]. Rashidi and Cook [79] designed a user centric interface, CASU-U, that accepts user input to identify and modify automated events and their times. These user-centric approaches are important at some level in improving the adaptation process; however, LFPUBS and CASU-U do not offer user adaptation during development, but only after the system has been launched. Consider the case of Bob, mentioned in the Section 1.1. Bob and his family can contribute to the smart home adaptation process. Bob's family or his carers can explain his behaviour and preferences to the smart home developer. Based on the information provided, the developer can install sensors and configure the system to suit Bob's needs. Likewise, the family does not immediately leave Bob alone in the smart house. In the user-centric approach, technology is adjusted to human activities; the family's concern about Bob living is gradually reduced, as trust in the smart home system increases.

2.3 | Transfer learning for activity recognition in smart home

In this section we provide an overview of recent research about transfer learning for activity recognition in sensor-based smart homes and address potential issues along their solutions. A generic and comprehensive review on transfer learning can be found in the survey works of Cook et al. [45] and Pan and Yang [76].

As mentioned previously, current research on activity recognition in smart homes focuses on the introduction of new machine learning algorithms [61]. Even machine algorithms, though, cannot provide immediate results where there is a lack of training data. With transfer learning, a system can leverage experience from previous tasks to improve the performance of new tasks, solving the challenge of missing training data [45].

Transfer learning has been categorized into several types depending on the applications, source and target domains, including differences in feature-space representation, marginal probability distribution, and conditional probability distribution among others [45]. Pan and Yang [76] proposed four classifications based on the transferred type; instance-transfer, feature representation transfer, parameter re- escalation transfer and relational-knowledge transfer. For the specific domain of activity recognition, the difference between source and target is more prominent due to the inclusion of time, people, sensors, and space. In the smart home context, it is presumed that enough data is available in the source domain while the main observation is the target domain [61].

	Supervised	**Unsupervised**
Informed	Labelled data available both source and target domain	Labelled data available only target domain
Uninformed	Labelled data available only source domain	Labelled data available neither source or target

Figure 2.2: Different types of machine learning

Traditionally, machine learning is categorized into two classes, namely supervised and unsupervised learning. The classification depends on the availability of labelled data [45]. Based on the source and the target domain, labelled data availability has been divided into four types: informed supervised, uninformed supervised, informed unsupervised and uninformed unsupervised [76] Figure 2.2. However, based on the label and data availability, the target domain could be classified into three categories: labelled, unlabelled and no data, as shown in Figure 2.3.

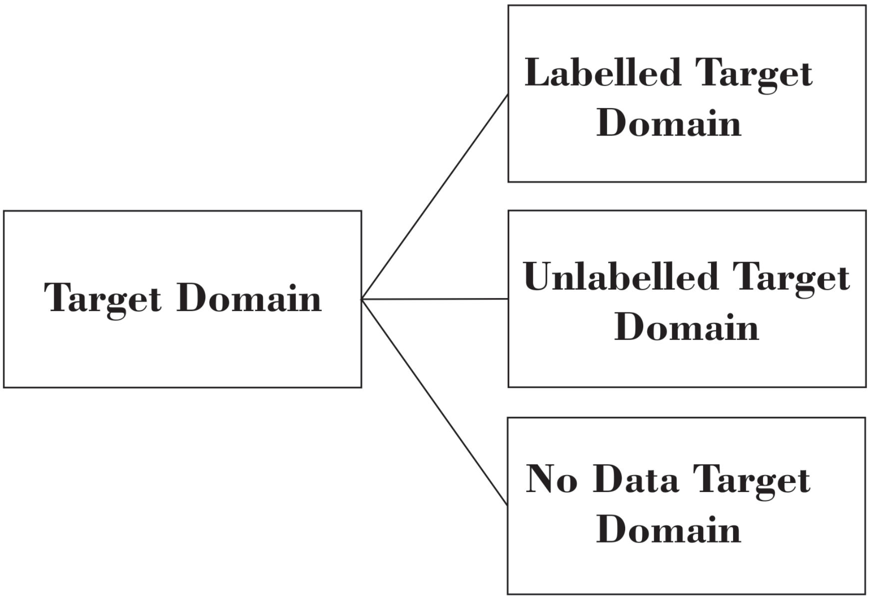


Figure 2.3: Different types of target domain.

2.3.1 | Labelled target domain

Labelled data can be available in the target domain regardless of the source domain defining the inductive learning [76]. Supervised transfer learning techniques can thus be considered for activity recognition, and in fact, this approach is popular for activity recognition in smart homes. If data availability in the target domain is adequate, the traditional supervised machine learning methods perform adequately in activity recognition. However, if there is a limited amount of labelled data, the activity recognition is inefficient [47]. In such cases, the transfer learning process can be elaborated at this point to improve activity recognition performance.

More precisely, in activity recognition, the primary challenge is to collect and annotate huge amounts of data for every single new physical setting to facilitate the customary activity discovery as well as the recognition algorithms.Rashidi and Cook [81] proposed a Home-to-Home Transfer Learning (HHTL) method to improve the performance of the target home in cases of limited datasets. The proposed method transfers learnt activity knowledge into new physical spaces. The method includes sensor grouping based on location and function and mapping similar types of sensors in the target domain. By using the proposed technique, several insights from prior spaces can be obtained that allow a better adaptation process.

While universal computer applications normally need information concerning the activities being undertaken, activity recognition models usually need a considerable amount of labelled training data for every setting [28]. Reusing the available labelled data in some new settings has been proposed as a solution in some cases. Dillon Feuz and J. Cook [47] proposed three different ways to achieve transfer learning, namely Feature-Space Remapping (FSR), the genetic algorithm for Genetic Algorithm for Feature-Space Remapping (GAFSR), and the Greedy search for Feature- Space Remapping (GrFSR). These techniques facilitate feature-based mapping, and a single day's labelled data are used for target domain validation and 30 days of labelled data for source domain validation.

2.3.2 | Unlabelled target domain

Several studies have been conducted on activity recognition based on unlabelled data in the target domain. Like in other domains, smart home data is not well-annotated, since labelling is one of the most time-consuming tasks involved in activity recognition. For that reason, source domain data is utilised for labelling the target domain dataset. Hu and Yang [57] introduced an approach that uses web knowledge as a bridge to build a map between the two domains. Transfer learning occurs in the area where the two domains have different sets of sensors and different activity labels, with the source domain having labelled sensor readings and the target domain unlabelled ones [57].

The transfer learning approach proposed by Rashidi and Cook [80], the Discontinuous Misplaced Sequential Method (DMSM), discovers variations of the required pattern from the target domain. Then, the Activity Mapping Method (AMM) is utilised and the results applied for mapping the source to the target. Another approach proposed by the same authors [82], Multi-home Transfer Learning (MHTL) helps recognize human activities from multiple physical smart environments and exploit this knowledge for a new or target home. This is achieved by a data mining method for finding the target activities from the target dataset by representing the source and target space in the same form. Then, a semi-expectation maximization (EM) framework is used to map each source to the target domain, finally fusing the multiple source dataset and labelling the target [82].

2.3.3 | No data target domain

There are no data available in the target domain and is considered a new approach in the transfer learning area. To date, published surveys [76, 45] do not discuss this approach. All other approaches are categorized based on data labelling; the lack of data labelling in this approach makes it difficult to categorize. The lack of available data in the target domain is a common issue for activity recognition in a smart home, because when a brand-new home is launched, no data or information is available for the occupant.

To tackle this problem, Chiang and Hsu [41] introduced a solution to build an intelligent smart home system in a laboratory, collect the required data, and then transfer it to a new home. During this process, the authors used sensor profiling methods for both the source and target domains, with additional background knowledge from the sensor network. The weakness of the method is that sensor profiles, like sensors, need manual profiling, making them appropriate only for certain datasets [41].

Table 2.3 indicates three types of smart homes (target domain). Any smart home used as a target domain can fall into one of the categories. For example, the no-data category could be the perfect match for Bob's scenario mentioned in Section 1.1. To improve the adaptation process, Bob's home can be considered a target home, and another, similar home type can be used as the source domain.

2.4 | Critical Analysis

It is clear that data-driven activity recognition is a rapidly growing area of research, driven by increased interest in smart homes. Concurrently, recent ad-

Table 2.3: Summary of transfer learning for activity recognition in a data-driven smart home.

Paper	Target Domain Data	Multiple Sources	Differences	Type of Knowledge Transfer
[47]	Labelled	No	Location, Layout	Instance-based and feature-representation
[81]	Labelled and Unlabelled	No	Layout, sensor network	Feature-representation
[57]	Unlabelled	No	Lab Space, Location	Instance-based and feature-representation
[80]	Unlabelled	No	People	Feature-representation
[82]	Labelled and Unlabelled	No	Layout, sensor network	Feature-representation
[41]	No data	No	Lab Space, Location	Feature-representation

vances in transfer learning methods have opened new doors for smart home research. This is because sensor-based smart homes can generate an enormous amount of data that can then be used to develop other smart homes. This application of transfer learning, however, demands further investigation.

A review of the literature reveals that very little scholarly work has focused on the new smart home adaptation process. Only one approach, proposed by Chiang and Hsu Chiang and Hsu [41], offers a potential new smart home adaptation process. The method involves collecting data from a user in a laboratory environment that simulates the smart home experience. After data is collected, a transfer learning approach is used to pass the data to the new smart home [41]. Chiang, Lu, and Hsu Chiang et al. [42] demonstrated that without any target data (i.e., no data) the amount of transferred knowledge is insufficient, but it can be increased using only a small amount of labelled data.

Data-driven approaches, while powerful, still present challenges for new smart home adaptation. By contrast, knowledge-driven smart homes do not require data but instead need a contextual knowledge, usually acquired by standard knowledge engineering approaches [31]. Based on this acquired knowledge, different approaches can be applied for representing activity recognition models, such as logic-based approaches, logical formalisms [17], event calculus [43] and lattice theory [31]. Another class of approaches, ontology activity modelling, is similar to the logical approach and uses a description logic based on mark-up language [39].

Knowledge-driven approaches are advantageous in the context of smart home adaptation, as they do not suffer from the cold start problem. However, in contrast to data-driven approaches, these methods cannot handle uncertainty. This is because they use inference and reasoning based on generic knowledge rather than uncertain sensor data.

Data-driven approaches are, in many ways, preferable to knowledge-driven approaches for smart home adaptation. However, to the best of our knowledge, there is no available method for tackling the cold start problem in a data-driven activity recognition smart home. As a result, the main aim of this thesis is to show how transfer learning can be used to improve the accuracy of activity recognition in new smart homes. The transfer learning method is rarely used when old smart home data is used to train the new home to recognize user activity. However, even in such cases, the accuracy rate tends to be very low if no data is available to the new smart home [42]. Pure transfer learning processes will therefore struggle to satisfactorily address the cold start problem without human input. In the context of the thought experiment posed earlier, it would be necessary to input Bob's daily activity information to the system before he begins living in the house.

To address the limitations of pure transfer learning, researchers often use data simulation tools that can evaluate newly designed models with synthetic data before implementing them in a smart home [67, 64, 90]. A particularly notable approach in this area comes from Azkune et al. [22]. The proposed hybrid methodology uses data from real user daily activity surveys as inputs to the simulation. The key idea is to gain insights into how the actual users will perform their daily activities in the smart home. The survey data is then processed by synthetic data generator tools for an arbitrary number of days to generate a labelled activity dataset. A key shortcoming is that this approach does not provide any synthetic data evaluation; that is, the created dataset is used only for modelling and recognizing user activity in a smart home. If the dataset is not applicable to the user, the process will not suggest any alternatives.

This chapter provides an in-depth review of user-centric approaches to activity recognition in the smart environment. As mentioned earlier, transfer learning is an advanced branch of machine learning used for activity recognition that can be effectively applied to the new smart home adaptation problem. Thus, this chapter also discusses the transfer learning approach in the context of the smart home domain. Finally, the method proposed in this thesis is introduced and justified against those that currently exist.

3

Methodology

This chapter describes the methodology that guided the research project. The User-Centred Intelligent Environments Development Process (U-CIEDP) is highlighted as the core of the methodology. The activities that led to obtaining the outcomes of the research project are explained under the framework of the U-CIEDP. Section 3.1 provides an explanation of the U-CIEDP, and how it guided the methodology of the research project. Section 3.2 explains the nature of user involvement in the initial programming of the system, as well as user's roles in refinements made further along in the process.

3.1 | The User-Centred Intelligent Environment Development Process

This section describes the methodology used for this project. The User-Centred Intelligent Environments Development Process (U-CIEDP) is relevant for this project as it situates the user at the centre of the development process. U-CIEDP has some unique features which convince us to consider this framework as system development methodology for READY approach. For example, frequent stake-holder involvement throughout the project and the life cycle of the sensing system is emphasised during the developed and installation. Several significant developments successfully applied the U-CIEDP framework. For example, recently, Quinde et al. [35] used U-CIEDP to develop context-aware solutions

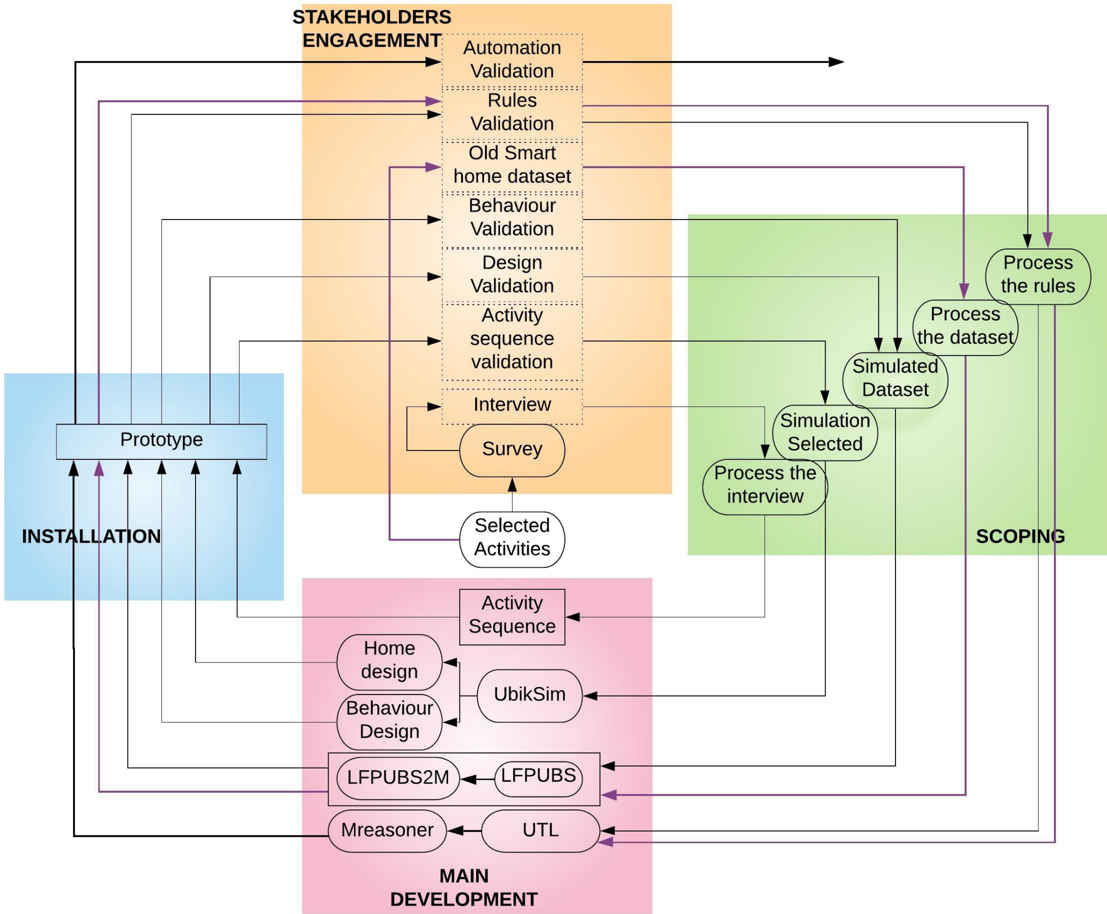


Figure 3.1: Methodology process.

to support the personalisation of asthma management plans. Augusto et al. [9] applied U-CIEDP for POSEIDON (PersOnalised Smart Environment to increase Inclusion of people with DOwn's syNdrome) project which aimed at helping people with Down's syndrome in smart environment. In POSEIDON, U-CIEDP was used as an iterative co-design methodology that involved all the stakeholders.

Figure 3.1 shows how the research project fits with the U-CIEDP framework and represents the incremental development of a new house adaptation system where every iteration was accomplished by user observation. The method consists of four squares: Stakeholders Engagement, Scoping, Main Development, and Installation. The square *Stakeholder's engagements* show the interactions that took place with the stakeholders. *Scoping* captured and conceptualized all the ideas gathered through stakeholder interaction. The *Main development* is the section where multiple systems were integrated, and READY was developed based on the requirements explained in the next section. Finally, the components developed for main development will be installed to ensure that they meet the needs of stakeholders in the *Installation* section.

It is important to explain each of the iterations in-depth to enable the reader to understand the project comprehensively. The project began with the *selected activities*. Activities were selected based on the smart home services the user expects. *Survey* questions were prepared for an interview based on the selected activities. In the interview, the user was asked to describe how they perform the daily living activities. The details of the activities and questionnaires will be explained further in section 4.1. The user's answers were processed, and *activity sequences* were created as the first step in the development of a READY prototype. Afterwards, sequences were validated with each user to ensure that they meet the users' needs.

The technical team then determined the appropriate tools required to translate the activity sequences into data. To convert sequences into data, several simulation tools are available, but none meet the project requirements since most of the simulation [56] focuses on generating datasets to test machine learning algorithms. On the other side, the proposed approach uses real data for the simulation. After some conversion, the data will be implemented in the system, which provides services to the user. Therefore, it is vital to carefully choose the simulation so that the significance of each user's needs is not lost. In the initial analysis, UbikSim [90] was selected, although it had not been developed specifically for smart homes. Despite its shortcomings, the software's core was close to our requirements, making it easy to use. In section 4.2, we describe how READY was customized to make UbikSim work for the project. We can create a virtual house and an avatar (which represents a virtual user) once the prototype is ready. Furthermore, the prototype is capable of generating simulated data.

Assuming the simulation dataset generation has solved the "cold-start" problem, we needed a system to verify if the simulation dataset reflects accurate information about user behavior. The LFPUBS learning system by Aztiria et al. [27] was used for the READY approach. It elicited knowledge from the underlying dataset and represented the discovered knowledge as a list of patterns. The representation of patterns helps the developer to form an in-depth understanding of the surfaced knowledge, which is vital to the success of the overall system. Section 4.3 explains the features of the LFPUBS system in detail.

Usually, the learning system catch up with the reasoning system. A framework of the reasoning system is available [19]. The project achieves its eventual goal of ensuring that the user receives automation services. Ibarra et al. [60] introduced MReasoner that is influenced by Galton and Augusto [48]. M reasoning system is capable of handling causality within context-aware systems, such as a modern smart home. A benefit of using the MReasoner is that both MReasoner and LFPUBS were developed based on the ECA (event-condition-action) paradigm; that is why they are pretty close to each other in syntax. Furthermore, with the support of an LFPUBS2M translator [14], LFPUBS data can not passed directly into MReasoner. We, therefore, added LFPUBS2M translator to the prototype.

The User-guided Transfer Learning (UTL) approach helps to increase the acceptance of the current rules generated from the *simulation dataset* and allows the developer to create an updated version of rules with the user guide. The old smart home dataset is the main component of this approach, where, similarly, the old smart home dataset (old dataset) is used to generate the rules, which can be called old rules. The old rules are used to modify the current rules (simulated rules), and also the user guide is provided to the developer for this modification process. A description of how the rules were generated and modified is given in section 4.4.

This section showed the READY approach incrementally developed with stakeholders (i.e., User) involvement following the U-CIEDP methodology. However, READY is slightly more user-centric because every iteration is repeated until the prototype satisfies the user. All of the components mentioned in Figure 3.1 have been described above. Five users were involved in system testing in Middlesex University Smart Lab. Next, twelve participants from a wide range of fields join a live online event to test smart home technology. During the project, all the University procedures were followed for ethical clearance, and all testing and validation activities were formally assessed and approved by the ethics committee. The following section explains the useR-guided nEw smart home ADaptation sYstem (READY), the paper's main contribution.

3.2 | READY in U-CIEDP framework

The aim of this project is to provide personalised smart home facilities as soon as a user moves into the new house. To achieve this goal, this project proposes the READY approach, which builds on the U-CIEDP framework. As mentioned, the U-CIEDP provides a framework for developed in which the user is able to provide feedback at each stage of system development. The complete READY methodology is explained in Chapter 4. This section explains the nature of user involvement in the initial programming of the system, as well as user's roles in refinements made further along in the process. Prototyping occurred in four stages, each explained in more detailed below.

3.2.1 | Stage one

System development began by selecting a set of activities for which the smart home would provide automated services. These activities should represent a combination of those the user is likely to engage in within the smart home, as well as those specifically requested by the user. After this initial scoping was completed, the developer interviewed the user to gain further insight into the types of technology (i.e., hardware and software) needed to fulfil user requirements. As is characteristic of the U-CIEDP framework, this initial of the first prototype was user-focused in nature. A particular advantage of this approach is that by involving users at outset, developers reduce the risk that there will be substantial modifications after the full prototype has been developed. With user requirements in hand, the developer then created a mock-up of the system.

After the set of activities had been chosen, system development proceeded with the developer creates a sequence of activities for the prototype to test. This sequence of activities was created based on answers from the user interview, in combination with the developer's knowledge of software and hardware constraints. Finally, the user was invited back to review the sequence of activities and provide feedback or revisions as necessary. This first stage of prototyping was only considered complete with the user's confirmation of both the selected activities and their sequence of presentation.

3.2.2 | Stage two

The next stage of prototyping involved the selection of an appropriate simulator. Because the U-CIEDP framework is so flexible—that is, it does not impose hard rules or regulations on the developer during system development—the developer is able to spend as much time as needed to choose a suitable simulator. For the purposes of smart home automation, however, and this project in particular, simulators had to, at minimum, allow for the following:

- Display of a user interface (UI) so that the developer does not need any technical knowledge to use the simulator.
- Creation of a 3D floor plan, along with the ability to place sensors with an adjustable sensing radius throughout the simulated environment.
- Configuration of smart home automation services that take into account user interactions within the environment (e.g., changing room lighting when a user lies down in a bed).
- Provision of a form-based event scheduler to facilitate the grouping of events in a particular scenario (e.g., the specification of a resident profile for movement speed, as well as start and end times).
- Demonstration of how activities are performed by showing how avatars might move within the smart home or interact with virtual sensors or objects (e.g., a PIR sensor detecting inhabitants when they enter or leave the kitchen).

- Adjustments to the layout of the environment (e.g., moving a table from one side of the room to another) to account for expected avatar movement paths.

Upon consideration of the necessary requirements, UbikSim was deemed the best choice for system development. UbikSim represents the first software element of system development. Importantly, it has two features: first, it allows for any intelligent environment to be developed; second, it enables the developer to simulate behaviour within that environment. Altogether, UbikSim provides the means for the developer to crate the smart home environment from which the script archived from the first prototype can be run. When it was initially selected, UbikSim did not allow for all the features outlined; however, it was determined to be the most adaptable for the purposes of this project. UbikSim was eventually modified to meet additional system requirements, and its current version now provides the necessary components described above.

After the developer finished development the smart home, simulations of avatar activity were presented to the user. These simulations served two key purposes: first, generating synthetic data; second, giving the user another opportunity to provide feedback on the fidelity of the system.

3.2.3 | Stage three

The third stage of prototyping involved the search for and selection of a tool to accurately recognize and classify human activity. In this project, the activity recognition tool had to meet several requirements, including:

- The ability to recognize the user's expected activities.
- Possession of a knowledge representation feature to extract patterns from the user interview of expected daily activities.
- An easily compressible system to accommodate the copious amounts of data needed for prototyping.

To meet these requirements, the LFPUBS [27] learning system was chosen. The main advantage of LFPUBS is that the output produces a set of rules that are much easier to understand than those of other forms of representation, such as Artificial Neural Networks, Markov Models and Bayesian Networks. LFPUBS elicits knowledge from the underlying dataset and represents the discovered knowledge as a list of patterns. In this project, LFPUBS identified patterns from the synthetic data generated in stage two of prototyping, and an associated system, LFPUBS2M, then extracted a set of rules from the most frequent patterns. Finally, users were consulted for feedback on whether the rules converged with the ways in which they would expect to interact with the smart home. If, for instance, the system produced a rule that showed the user waking up between 5 and 7 am, but the user said they typically arose between 6 and 8 am, the developer would incorporate that feedback into the programming and correct the rule.

Again, the aforementioned tools were chosen, in part, because they are fairly intuitive and easy to understand. Such affordances were considered benefits not only in this project, but also for future developers of varying levels of knowledge and experience hoping to adopt the READY method.

3.2.4 | Stage four

In the fourth and final stage of prototyping, the rules extracted by LFPUBS2M were applied to MReasoner. As described in previous sections, MReasoner is a computational tool for modelling reasoning processes based on the ECA paradigm. In this project, MReasoner was used to aid in the last part of the automation process.

The final test of the system involved setting up the smart home with all the required hardware. Then, MReasoner was run with the rules it received during stage three. As with previous stages, this stage considered user feedback and made any changes as necessary.

The UTL method was applied during this stage as well. As explained, the UTL method offered a complementary approach to that of the READY method by taking data produced by users in the smart home laboratory, comparing it to the rules generated from the synthetic data set, and modifying those rules based on any discrepancies discovered. A more detailed explanation of UTL approach is provided in Section 4.4.

3.3 | READY in brief

As mentioned before, READY was developed using the U-CIEDP system development approach. The next chapter explains this method further. READY is briefly discussed in this section. READY was created to provide smart home services from the outset. The main obstacle in providing these services is the cold-start problem. READY aggregates four approaches to tackle the cold-start problem: survey, simulation, activity recognition and transfer learning. Each of the approaches individually has some contribution to the smart home domain. To the best of the author's knowledge, these methods solve together the data-driven smart home problem in this project.

3.3.1 | Survey to collect the user daily activity dataset

The data-driven activity recognition system development started with the data collection process. However, as the research focuses on adapting a new home for a new user, no data has been collected for activity recognition. Therefore, this section proposes a method to collect daily user activities through surveys for generating the activity dataset.

3.3.1.1 | Approach

The sensor-based smart home monitors that track the user activities based on the status (activation) of the sensors. Therefore, the smart home database mainly collects three elements: sensor status, the sensor's name, and the activation time. Smart home sensors are connected with the object, which means if any of the sensors is activated, the belonging object is being used. Based on the order of the sensor, an activation activity model is created. Likewise, it can be determined that the activity model is also created based on the user's usage, which is

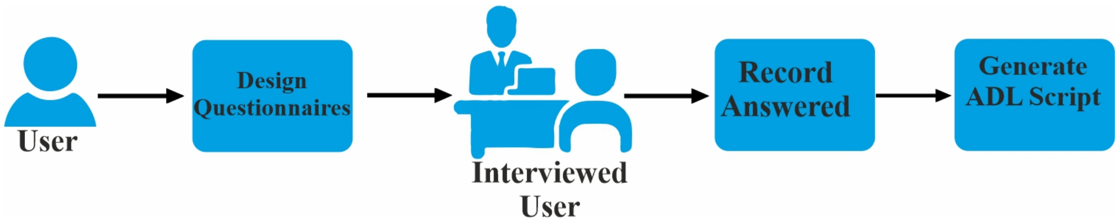


Figure 3.2: Data collection approach for the new home

the critical concept of generating user daily activity dataset in the smart home. Figure 3.2 shows all the steps are taken for the survey.

The user activity model is defined based on the sequence of the usage of the object by the user. For example, making tea activity objects uses sequence such as turning on kettle, getting the mug from the cupboard, and opening the fridge for milk. So, the making tea activity can be drawn as follows:

Making tea » Kettle on →Cupboard on →Fridge on

On the other hand, the user behaviour model is created based on the sequence of activities by users associated with time slots and time-lapses. For example, the user's weekday morning activity pattern is to wake up at 7:00, use the wash room at 7:15, use the shower room at 7:30, make breakfast at 8:00 and go outside at 8:30. So the sequence of the activity is as follows:

Morning activity » 7:00-8:30 wake up →Use washroom →Showerroom → making breakfast →going outside

3.3.1.2 | Summary

As mentioned before, the data is collected in face-to-face interviews because of the detailed understanding of the user activity. Moreover, the lack of domain knowledge could make the questions less understandable for the user. The survey questionnaires, data collection techniques and data processing are explained in detail in section 4.1. In conclusion, the section result is generated by the activity sequences script, which the simulator would execute.

3.3.2 | Simulation to generate synthetic dataset

Although several simulations are available to generate the human activity dataset, the research is based on a simulation with particular features mentioned in section 3.2.2. Also, all the simulation development generated by the dataset is to test and evaluate the algorithm. So, traditionally simulation is used for testing purposes. However, on the other hand, this research simulation is critical because the data generated by the simulator is thoroughly used for the system development, which finally provides services to the user.

3.3.2.1 | Approach

According to the recent literature mainly two categories of approaches for data simulation: model-based and interactive.

Model-based approaches generate the simulation data based on the order of the events, the probability of the events occurring, and the time taken for each event. The accuracy of the data heavily relies on the quality of model description and related parameters. The model-based approach does not focus on the IE environment development and context awareness. Following the U-CIEDP approach, one of the objectives of this research is to provide a sense of the system to users during development, an objective not fulfilled by the model-based approach.

Interactive approaches create an IE environment and provide heavy control over the activities and generated dataset. This approach mainly focuses on developing a virtual environment rather than an activity model. Thus, the interactive approaches slightly fulfil the requirements of our research expectation. However, the approach does not allow the user activities model, which is the main element.

There are a few hybrid approaches available. OpenSHS [12], recently published simulation software that allowed the developer to generate a database using both approaches. However, the simulation does not allow fine-grain control of the PIR sensor area created. For example, the Smart Lab corridor movement sensor detects the user movement around one meter inside the bedroom

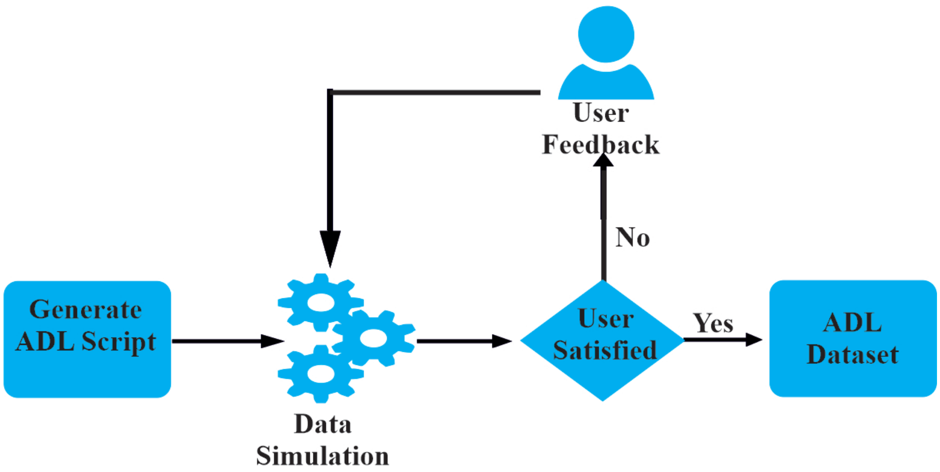


Figure 3.3: The method to generate synthetic dataset using simulator

door, which is difficult to create by the simulation. Figure 1.2 illustrated the user involvement of simulation process.

Section 3.2.2 explained the feature needed to choose the simulator for this research. It is recommended to create a new simulation for this project. However, UbikSim is chosen for the following reasons.

- It allows the developer to create smart home simulations without heavy labour.
- It allows added objects from sweet home libraries, which helps the developer take the virtual smart home closer to the actual home, which is essential for our research. As mentioned before, one of this research's expectations is to introduce the user to the home before moving to the home.

For the project, the following new features were added to the UbikSim:

- The developer had fine-grain control over the sensors and floorplan.
- Database was added to the simulator. So, the user (avatar) activities are recorded in the database.
- It allows the developer to insert a script, and the avatar act according to the script.

3.3.2.2 | Summary

The research modified and developed subtle simulation despite several simulation ready to use because this project tackles two key features; it introduces the developer to the new home, so the virtual house should be expected. The approach relies only upon the survey data. Therefore, the simulator does not lose the ground truth of the data. However, the research does not address some other real-world problems that are more complex such as inactive or missed fire sensors, which is beyond the scope of this research.

3.3.3 | Examine the smart home services for simulation dataset:

This research aims to provide smart home services to the user when the user starts living in the new house. First, the user habit needs to be detected to provide the user's expected services. Human activity recognition is a well-known technique in the smart home domain to detect user activities in the smart home. In this section, the data is inserted into the activity recognition system to learn the user habit from the dataset. Later, the data output is inserted into the reasoning system to see how accurately the system automation reflects user expectations.

3.3.3.1 | Approach

Some essential factors are needed to be considered before finalising the activity recognition system. First, the quality and amount of data are needed for good accuracy from the system. There is an interesting debate about the amount of training data required for classification [73]. Zhu et al. [95] claim that the classifier accuracy increases with respect to the size of the data, and the little noisy data will not impact the accuracy. However, Chapelle et al. [38] argues that the model selection is more vital than the amount of data as different model training data expectations differ. So, there is no rule of thumb as to how much data is needed for training. In this project, we will take the imperial approach, study

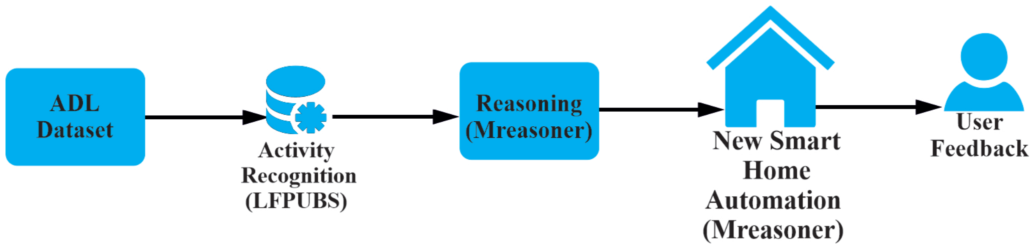


Figure 3.4: ADL dataset using for activity recognition

the relationship between the amount of training data and performance for the individual models.

This research chooses LFPUBS for activity recognition because LFPUBS approaches the problem holistically, i.e., in comparison to the other systems, it discovers all the aspects of user behaviours (frequent actions, order (topology), time relations and conditions). In contrast, the other systems focus only on one single aspect of user behaviour. section 4.3.1 shows how the LFPUBS system helped process, monitor data and extract knowledge from those datasets. Figure 3.4 is drawn the conceptual steps involved in this section.

After translating the output by the LFPUBS2M, the LFPUBS inserted reasoning system [60], which provides the framework to control the sensing environment, automate the devices, and provide the services according to the M rules. As mentioned previously, after finishing the steps, following the U-CIEDP research, it goes back to the user to take feedback.

3.3.3.2 | Summary

Activity recognition in smart homes is a vast research area that is full of machine learning algorithms where researchers compete to generate more accurate algorithms. So, the area has become more techno-centric rather than user-centric. However, this research is fully user-centric; every step of the system development involves the user.

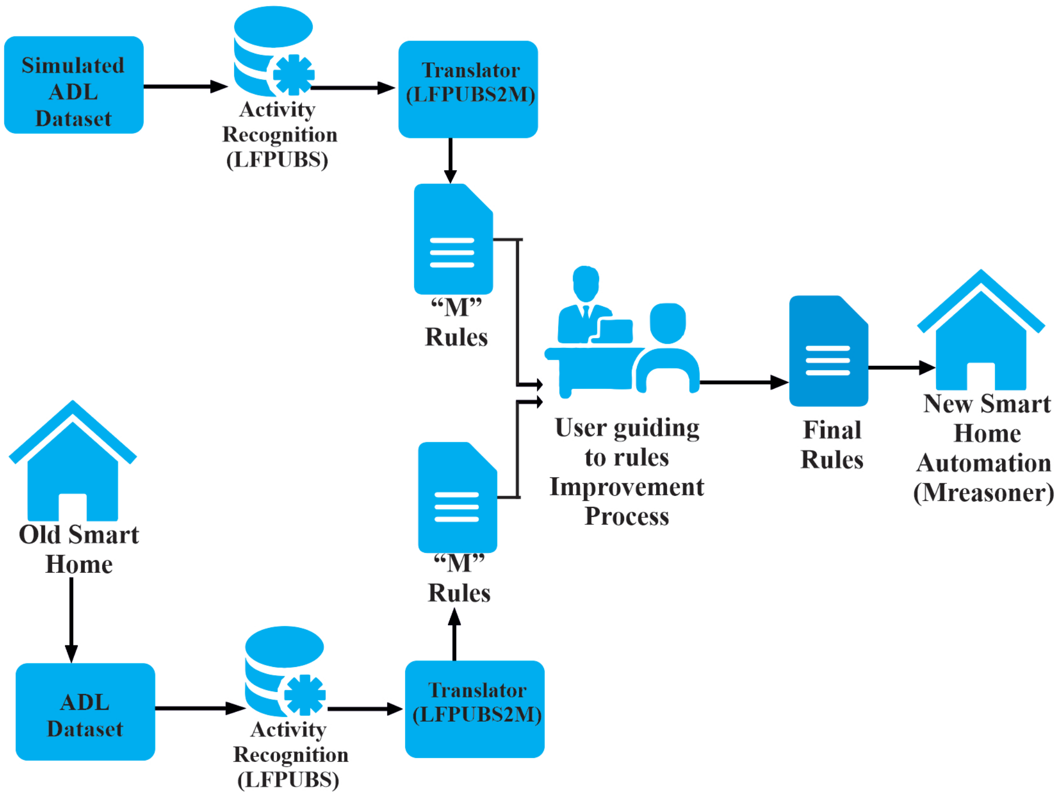


Figure 3.5: UTL approach in brief

3.3.4 | Assure READY by transfer learning

The goal of activity recognition is to classify the current activities based on the previously collected data. To define the activity, recognition algorithm needs sufficiently labelled training data. The researcher tries to create a connection between different domains activity recognition dataset through transfer learning, which helps to reduce the training time and effort to initialise the new activity recognition system. Transfer learning can apply in many forms to activity recognition [45]. This section described the user-guided transfer learning approach (UTL), where the old smart home knowledge is extracted from data and used for the new smart home.

3.3.4.1 | Approach

The sections above described how the READY approach provides smart home services to a new smart home user from the outset, which is the crucial aim of this research. On the other hand, the UTL approach is supplementary to the READY method, where old smart home data is processed accordingly. The generated M rules are compared with the pre-existing rules using the user's assistance and the most appropriate one is finalised. Figure 3.5 exhibit the process of UTL.

3.3.4.2 | Summary

Every step of the READY approach is monitored by the developer and also feedback is taken from the user. However, it still could have some concerns because of the data generated by the simulator. UTL revaluated the READY result to overcome the concerns.

4

useR-guided nEw smart home ADaptation sYstem (READY)

This section describes the READY method. Figure 4.1 shows the conceptual architecture of the system (number will be used further down to explain the process). READY aims to provide a user with smart home services as soon as they start living in a house. READY is an integrated system that brings together four separate approaches: survey, simulation, activity recognition and transfer learning. As mentioned previously, adopting a U-CIEDP approach allowed the system to evolve as a natural consequence through several iterations before building the final system.

READY is the critical element of this project. The first version of the READY is important because the data collected from the users is very raw. Data is processed by developers and entered into the system, but errors made at this point will affect the final version of READY. Therefore, the initial interview should be conducted face-to-face with the user in order to ensure that the interviewer is able to retrieve the necessary knowledge without misunderstanding. The U-CIEDP approach ensures that the next iteration of READY development will only occur if the user has agreed with the current one. User responses are processed and converted in sequence to facilitate the next step. The data processing technique utilised in this study is explained in section 4.1.

A simulation is a tool well suited to transform an interviewed answer to a daily activity dataset. Hence, simulation was added to the READY method. A

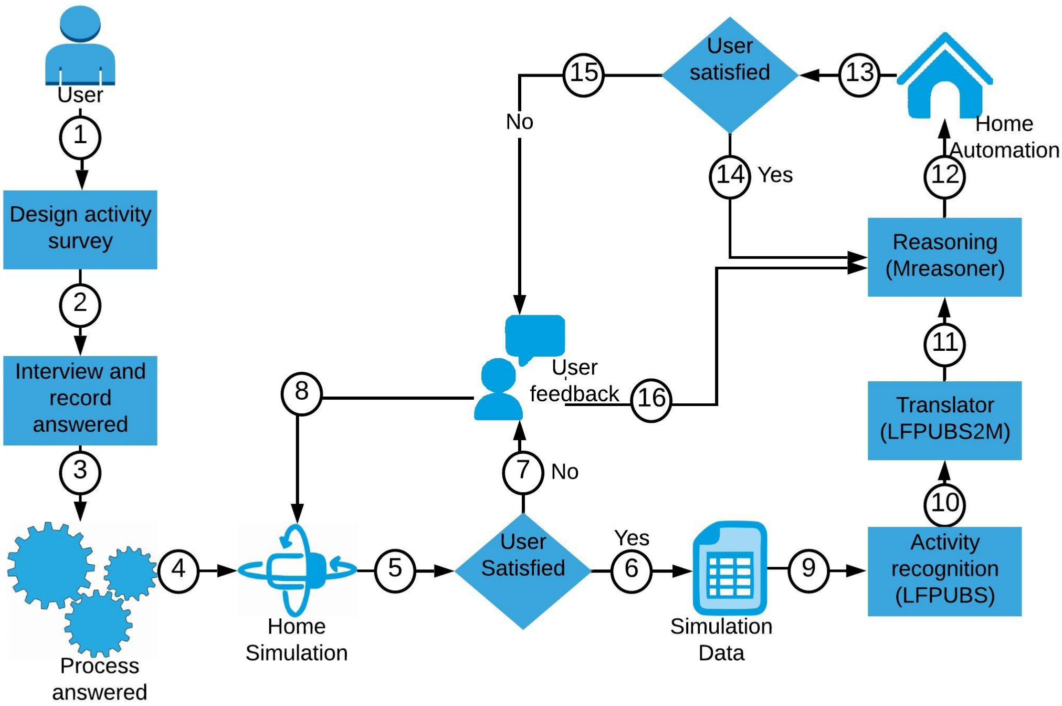


Figure 4.1: READY System Architecture.

simulation mimics a real smart home where data is generated by an avatar when it moves within the smart home and either actively or passively interacts with the virtual sensors. Section 4.2 contains details of the simulation.

Activity recognition is a well-known approach for the extraction of human behaviour from a dataset. The final aim of this project is to provide the home automation service to the user. So, it is essential to understand user behaviour before automation. It is for this reason that READY includes activity recognition and automation tools. Section 4.3 contains further details of these tools.

Transfer learning is the method utilised to transfer old smart home knowledge to a new smart home while taking guidance from the user. Working with the old smart home dataset helps to measure the accuracy of the simulated dataset, and at the same time, it gives an opportunity to improve the current knowledge if it is not sufficient to provide the user with the required services.

Section 4.4 explains how transfer learning integrates with READY.

This section explains and illustrates how READY uses each of the components in order to provide smart home services to the user. Two users participated voluntarily in this testing process. We have named them user A and user B.

Scenario: *The user wakes up, uses the bathroom, and then goes to the kitchen to make their breakfast. They then eat breakfast, go back to their bedroom, get ready, and go outside.*

The user would expect lights in the bedroom, corridor, bathroom, kitchen, shower, and on the table to come on automatically as needed, as well as a kettle and radio in the kitchen. The user will also expect the switching off all automated devices if they forget to switch them off before leaving the house.

4.1 | Survey to understand the daily user activities

Data-driven activity recognition systems predict human behaviour via analysis of the user's past daily living activity dataset. Unfortunately, a new smart home does not have any records about the user activities or preferences. At this point, we require a technique to enrich the data available to the system. For this reason, we offered a questionnaire [10] to collect daily living information of the user (Figure 4.1, step 2). Once the user activity and behavioural data are collected in details, the information will be ready to enter into the system.

4.1.1 | Collection of activity data

In order to select target activities, a questionnaire must be created based on the services the user desires. For simplifying, target activities were divided into two categories; namely *simple activity* and *complex activity*. The number and sequence of actions in the *simple activity* category are the same for all users. For example, during the activity of "wake up", three separate sensors, bed pressure, bedroom motion, and bedroom light detect activity irrespective of the time each activity occurs.

Conversely, the number of actions and their sequence differs for *complex activities*. As an example, making a cup of tea is considered a *complex activity*. There are two ways to prepare tea: some people use milk, some do not, and the sequence of actions in the process can also differ.

In the questionnaire, each user specifies the days that a particular activity occurs, the activity time slots, and the time relation between any two consecutive activities. For example, between 5:00 and 6:00 p.m., the user may enter the house and then go to the kitchen ten minutes later to make a coffee.

4.1.2 | Collection of behaviour data

In this part, our objective is to understand each activity in greater detail to identify factors specific to each user. Data collected will include when the activity occurred, the sequence of activities, their duration and location. Objects used to complete each activity are also important because this information suggests what sensor will be necessary to detect the activity.

4.1.3 | Survey evaluation with a real scenario

The users (user A and user B) were invited individually for the interview. As previously described, the interview was held face-to-face with a pre-created set of questions posed to each user. Users were encouraged to explain how they perform various activities naturally.

We only considered monitoring those activities necessary to provide a particular automation facility. According to the above scenario on section 4, the monitoring activities were- wake up, use the bathroom, use shower, make tea, and go outside.

The time slots and the activity sequences performed within those time slots for both users A and B are in Table 4.1. Simple and complex activities (Table 4.2) have been separated and given unique names (labels). For example, we discovered that "Make tea" is the single most complex activity because making tea

Table 4.1: user's weekday activity sequences.

User	Time range	Activity and sequences
user A	06:00-07:00AM	Wake up → Use bathroom →Make tea →Go outside
user B	06:30-08:00AM	Wake up → Use bathroom →Use shower →Make tea →Go outside

Table 4.2: Example of simple and complex activities.

Scenario	Simple Activity	Complex Activity
Morning	Wake up, Use bathroom, Use shower, Go outside	Make tea
Evening	Enter home, Use bathroom, Sleeping	Make tea, Relaxing

could be different for different users; some people use milk, some do not, and as a result, the sequences of the action could also be different. We have described the action steps involved in the performance of each of the activities in Table 4.3.

4.2 | User behaviours and smart home simulation

A simulation (Figure 4.1, step 4) is created to establish and acquire user knowledge of a new house, model user behaviour based on user responses, and generate an initial dataset. In case of READY, UbikSim [88] is used to draw the simulation. Although, any simulator can be used for the READY, specifications mentioned in the sections 3.2.2 represent the reasons behind the author's choice of UbikSim.

UBikSim is an open source, Java-based program with a rich library. These features make it easy to integrate with the other components of the proposed system. Originally developed to study complex Multi-Agent Systems (MAS), UbikSim is modified to include new features for this project.

For this project, we utilised UbikSim in two phases. Phase 1: Virtual house developed and phase 2: Simulate the user's behaviour.

Table 4.3: Users' action sequences for particular activities.

Activity name	User	Action involves	New name
Wake up	user A	BedPressure ON →BedroomMotion ON →BedroomLight ON	N/A
	user B	BedPressure ON →BedroomMotion ON →BedroomLight ON	N/A
Use bathroom	user A	CorridorMotion ON → Bathroom-Door OFF →BathroomMotion ON →BathroomLight ON	N/A
	user B	CorridorMotion ON → Bathroom-Door OFF →BathroomMotion ON →BathroomLight ON	N/A
Use shower	user A	CorridorMotion ON → ShowerDoor ON →ShowerMotion ON →ShowerLight ON	N/A
	user B	CorridorMotion ON → ShowerDoor ON →ShowerMotion ON →ShowerLight ON	N/A
Make tea	user A	KitchenDoor ON → KitchenMotion ON →Kettle ON →Cupboard ON →Fridge ON	Milk Tea
	user B	KitchenDoor ON → KitchenMotion ON →Kettle ON →Cupboard ON	Black Tea
Go outside	user A	CorridorMotion ON → EntranceMotion ON →CorridorLight OFF →EntranceDoor OFF	N/A
	user B	CorridorMotion ON → EntranceMotion ON →CorridorLight OFF →EntranceDoor OFF	N/A

4.2.1 | Virtual house developed

In this phase, the developer needs the original floor plans and furniture layout of the new home in order to develop the virtual home. UbikSim editor is then used to prepare the virtual floor plan and control different aspects, such as room dimensions and available square footage. Next, furniture, appliances and other home items are added from either UbikSim library or Sweet Smart Home library [2]. UbikSim supports the Sweet Smart Home library, which has an extensive collection of home furniture and appliances. Finally, import required sensors to the smart home from the sensor's library. All the required sensors are available in the current version of UbikSim, such as motion sensor, door sensor, light sensor, object sensor and pressure sensor. An advantage of UbikSim is that it allows developers to add smart features to any home furniture or appliances easily.

4.2.2 | Simulate the user's behaviour

After creating the virtual smart home, the developer needs to decide the context to developing the simulation. Here, context means the specific time frame to be simulated such as morning, evening, or afternoon.

In the previous section, the developer allocated a name (Table 4.3) to each particular activity. So, in this stage, the activity name allocated is assigned as an activity label. Now the virtual home is ready to perform. Before running the simulation, the time and location of the avatar (virtual user) also need to be assigned.

Within UbikSim, an avatar is an interactive object that can move within the virtual smart home and passively or actively interacts with the virtual sensors to represent the behaviour of a real inhabitant. A server records the interaction between the Avatar and the virtual sensors.

This example illustrates how READY engages the user in the simulation process. In the first interview, the developer gathers the required answers needed to simulate user behaviour in the new house. The user becomes familiar with the new home in a virtual environment and examines the simulated behaviour. If the user has agreed that the simulation reflects his or her daily living activity, then the next step in the development process is initiated. Otherwise, the step is repeated.

Figure 4.2: Simulation of the Middlesex University smart space lab.

4.2.3 | Simulation evaluation with a real scenario

The Smart Spaces Lab of Middlesex University was used to evaluate the scenario in section 4. This Lab contains a living room, a bedroom, a kitchen, a bathroom, a shower room and a corridor space.

To facilitate the simulation, we created a virtualised replica of the physical Lab environment using UbikSim editor. We paid particular attention to ensure that the virtual environment looked precisely like the Lab environment and that none of the independent sensors and house appliances embedded with a sensor should be excluded. The results of this simulation are shown in Figure 4.2.

In a second interview, we examined and validated the simulation of expected user behaviour based on the information provided in Tables 4.1, 4.2 and 4.3. In addition, the simulation should be modified to include any newly discovered information and any information that had been overlooked or inadvertently omitted from prior consideration.

Table 4.4 shows feedback received from users A and B. This feedback contains vital information for the further development process. For example, user A corrected that he usually sits on the bed 5-10 min, and he also does not take a

Table 4.4: The feedback received from the user.

Activity name	user	feedback	Sequence #
Wake up	user A	After waking up user wait 5-10 min on the bed	1
	user B	Accepted, no feedback	1
Use bathroom	user A	Accepted, no feedback	2
	user B	The user goes to the kitchen before the bathroom to put the kettle on	3
Use shower	user A	N/A	N/A
	user B	Accepted, no feedback	4
Make tea	user A	Milk tea	3
	user B	Black tea	2
Go outside	user A	Accepted, no feedback	4
	user B	Accepted, no feedback	5

shower in the morning. According to user B, his first activity after waking up is turning on the kettle in the kitchen, creating a change in the sequence of activities. Hence, we shall modify our simulation accordingly and, finally, generate the dataset.

4.3 | User activity recognition

This section describes how we use activity recognition, translator, and reasoner tools to provide the user with smart home services. Notably, the READY approach is flexible to use any activity recognition and reasoning tools, however, the tool must comply with the minimum requirements mentioned in section 3.2.3. LFPUBS is considered here because it approaches the problem holistically; for example, it discovers all aspects of user behaviours (frequent actions, order (topology), time relations, and conditions), whereas the other systems only focus on one single aspect. Furthermore, LFPUBS2M and Mreasoner complement LFPUBS, ultimately helping us to check whether the produced data can provide the user smart home services.We offer a thorough explanation of LFPUBS, LFPUBS2M and MReasoner in the following sections.

4.3.1 | LFPUBS

The purpose of smart home automation services is to make the user's life more comfortable, safer and efficient. The habits of human beings determine their behaviour. As a result, the past and present behaviour of the user is also indicative of future behaviour of the user. This section provides a brief description of the LFPUBS system, which played a vital role for this project. The details of the LFPUBS system is available in [27]. The project used LFPUBS to identify behaviour patterns from within the user's daily activity dataset. LFPUBS consisted of a three-layer architecture, each of which played a vital role in representing data so that the developer could visualise the sequential transformation of the data. In addition, this feature of LFPUBS convinced us it was the most suitable for this project. Furthermore, LFPUBS provided a user-centric development aligned with the overall goal of this project, which involved collecting user data and acting intelligently on behalf of the user. LFPUBS layers are explained as follows:

4.3.1.1 | Transform layer

In the first layer, raw data from sensors is transformed into information for use in the learning layer. As acquired data various with the environment, different transformations must be performed before introducing the data to the learning layer. Data is represented as a string of actions with a temporal order but without any structure or organisation. All the introduced data is collected sequentially to determine the meaning of the data by dividing it into sequences taking into account the duration from the beginning of the sequence to the end of it.

4.3.1.2 | Learning layer

The learning layer is the core of the system. By transforming the meaningful information passed to it from the transformation layer into knowledge, the learning layer keeps itself independent of external influences. Action Map is one of the two approaches used by the learning layer to generate knowledge from the information passed to it. A pattern-based approach identifies user behaviour patterns and represents them in a comprehensible manner, while pairwise approaches focus on discovering pairwise relationships between the actions of the user. The Learning layer consists of two separate but fully integrated modules: a representation module and a discovery module.

Representation module This module uses a language called L_{LFPUBS} that represents patterns based on Event-Condition-Action (ECA) rules. Besides providing a standard way of describing patterns, it also allows other technologies to verify the integrity of the patterns. The frequency behaviour defines an Action Map (Figure 4.3.3.1), containing all relevant patterns of frequency behaviour. In LFPUBS language, two actions (ON and THEN clauses) and one condition (IF clause) are tied together, similar to the ECA rules. In addition, it describes the time relationship between the two actions. Figure 4.4 shows how L_{LFPUBS} language represent turning the kettle on.

Discovery module The A_{LFPUBS} algorithm is used to discover the frequent behaviours of the user in four phases. Firstly, the algorithm identifies the frequent sets of actions followed by the topology and quantitative time relations and conditions. A_{LFPUBS} algorithm represents the most frequent behaviours of the user using the Action Maps. This project uses LFPUBS because it has the power to represent data at every step, as mentioned before. All four phases can be represented in GUI. The four phases are explained below.

Identifying frequent set action In this step, the developer determines the most frequent set of actions that occur together, allowing them to specify parameters of the minimum confidence level. A minimum confidence level considers only the set of actions occurring above the minimum time interval, which is called Frequent sets. Apriori [4] algorithm is used to find out Frequency set of action. According to the activity recognition domain user behaviours, the most frequent activity is the user behaviour or habit.

Identifying topology After finding the most frequent action, it is important to determine the order of the actions. Aztiria et al. [27] introduced several algorithms to manage the ordered and unordered subsets of actions along with the repeated actions. For a more detailed understanding of these algorithms, refer to [26, 25, 23]. The user's behaviour was represented through the Action Maps in our research context; thus, the response to the question about the user's daily activities in section 4.1.1 was reflected in this pattern, providing a key advantage of using the LFPUBS in our study.

Identifying time relation Each action pattern is shown in the topology as a time-relationship. Qualitative relationships allow us to understand the logical order of actions. The quantitative time relation is considered in order to understand higher quality information. Two algorithms, basic algorithm and EM algorithm [87], are included to determine the time distances of all occurrences in each pattern.

Identify conditions This section identifies and describes two types of conditions, specific and general.

Specific condition is all the relations displayed in the Action Map based on the number of occurrences such that one situation is followed by two or more different actions. These situations are easily identified on an Action Map since they are represented as splitting points where more than one relationship is formed. For those cases, it is necessary to identify under what conditions each of those relations is true.

The *general condition* refers to the calendar and the context information that provide the user with an understanding of the circumstances under which the action map occurs.

4.3.1.3 | Application layer

The application layer displays or uses the knowledge generated by the learning layer to develop various Human Computer Interfaces (HCIs) for different learning processes.

4.3.1.4 | Summary of LFPUBS

A key objective of LFPUBS is to discover and represent the entire behaviour of the user, without limiting how many actions are involved in the pattern. Automated and monitored patterns have also been discovered and represented with this system.

In this section we briefly describe the LFPUBS and its function. While we did not describe the algorithm used for LFPUBS, we did describe how the algorithm is worked.

4.3.2 | MReasoner

This MReasoner is a system that has a language for defining context of interest, as well as the ability to track certain environmental conditions and act upon them as they arise [60].

Ibarra et al. [60] introduced an Intelligent Development Environment (IDE) for developing reasoning systems. The IDE capabilities include reasoning development, automatic LFPUBS rule translation (LFPUBS2M is integrated into the system), and deployment and execution of development systems. The following section describes these three functionalities in detail.

4.3.2.1 | Reasoning development: Forward Reasoning

When employing an inference engine, it is one of two basic methods of reasoning, and it can be logically represented as repeated application. It starts with the data that is available and then utilises inference rules to extract more data (from a user, for example) until a goal is met. Forward reasoning is used by an inference engine to explore through the inference rules until it finds one with a known true antecedent (If clause). When such a rule is discovered, the engine can infer the consequence (Then clause), leading in the addition of new data to its database. This approach will be repeated till a target is reached via inference engines. The forward reasoning algorithm described in [60] is implemented by the MReasoner. Furthermore, a database is employed as a means of communication between the reasoner and the real world, enhancing the structure's flexibility by allowing it to operate without knowing what is involved in the data gathering method or the actuators' actions.

4.3.2.2 | Automatic LFPUBS Rule Translation: (LFPUBS2M)

Another feature is the ability to convert the LFPUBS Output file into systems specifications, allowing it to be directly coupled or modified by another automatic module or developer before being inserted into the reasoning module. The reasoner may be configured using the System Specification because it is expressed in the same syntax. System Specifications are rules that specify the nature and initial state of each one in order to ensure that the reasoner can reason.

It is also known that the rules created by LFPUBS cannot be introduced to the system without any supervision. If the system can concatenate the rules, it must have a single means to relate them, so if there are multiple paths that can be concatenated, it will create an illogical rule. For reducing erroneous behaviour and conflict probabilities, rules are filtered before being introduced to the LFPUBS2M system, which is integrated into the MReasoning system.

4.3.2.3 | Deployment and Execution of the system

It is desirable to be able to deploy and execute reasoning systems so that Intelligent Environments can be created easily. In addition to these features, depending on the development stage, the system under development can be executed from the IDE in three different ways:

- Simulation (Time expressed in iterations)
- Simulation (Time expressed as Real Time)

- Deployment and execution of the real system

Deployment and execution of the real system have been used in this work to validate the translation, and reason with it.

4.3.2.4 | Summary of MReasoner

To reason and achieve a connection between the real environment and the system, the MReasoner was employed. If an instant translation of the rules is required, the translation system has been implemented into the system.

4.3.3 | System evaluation with a real scenario

The evaluation process described in section 4.2.3 resulted in generating a simulation dataset for user A and user B. This section explains the details of that evaluation and explains how LFPUBS and LFPUBS2M process the simulation data to generate M rules, thereby providing automation services to user A and user B. Note, All the pattern and rules produced for User A and B are not shown here and can be found in [9]. This section mainly focus to illustrates the mechanism of LFPUBS and LFPUBS2M how they produce the rules and patterns.

Before generating the initial simulation dataset, the developer needs to understand how LFPUBS identifies the most frequent pattern from the data. The developer initially defines the sequences of activities performed by a user (avatar) within the simulator. Then, LFPUBS discovers the most frequent activities from the dataset. So, the developer should be aware of this behaviour before generating a simulation data set. The researcher is trying to provide automation services to the user through a system where the system still has not formally collected any data (activity recognition system naturally collects user daily activity data). So, the developer needs to know how chosen activity recognition system works for data. How much data needs to be inserted into the system to make the system knowledgeable for the user. Future research could experiment with different types of activity recognition algorithms with different amounts of data. In section 4.3.1, we discovered that LFPUBS uses the Apriori algorithm to find basic frequency sets in a large amount of data. For this research, twelve

```
2020-03-14T08:03:47,BedRoomDoor,ON,100
2020-03-14T08:03:49,BedRoomDoor,OFF,0
2020-03-14T08:03:52,KitchenDoor,ON,100
2020-03-14T08:03:54,KitchenDoor,OFF,0
2020-03-14T08:03:54,KitchenDoor,ON,100
2020-03-14T08:03:57,KitchenMotion,OFF,0
2020-03-14T08:03:57,KitchenMotion,ON,100
2020-03-14T08:03:58,KitchenMotion,OFF,0
2020-03-14T08:03:59,KitchenMotion,ON,100
2020-03-14T08:03:59,KitchenMotion,OFF,0
2020-03-14T08:04:00,KitchenMotion,ON,100
2020-03-14T08:04:01,KitchenMotion,OFF,0
2020-03-14T08:04:01,KitchenMotion,ON,100
2020-03-14T08:04:01,Kettle,ON,100
2020-03-15T08:10:34,BedRoomDoor,ON,100
2020-03-15T08:10:37,BedRoomDoor,OFF,0
```

Figure 4.3: Example of a converted dataset used for the Transformation layer.

weeks of synthetic data were created using the simulator to test user A and B's scenarios. Between eight and twelve weeks, the data was generated based strictly on the answers given by A and B, and the rest of the data was generated by random simulation. For this reason, it is guaranteed that the LFPUBS pattern reflects user behaviour. LFPUBS and MReasoner internal structure are explained in the section 4.3.1. The following sections illustrate how data travels through the LFPUBS system and how knowledge is extracted from the data as a pattern.

4.3.3.1 | Knowledge extracting by LFPUBS

As explained in the previous section, the transform layer transforms the raw data for the learning layer to describe the meaningful information. See Figure 4.3 for an example of converted data where data itself is meaningful. Next, the data is inserted into the LFPUBS system to identify frequent patterns. After defining all the parameters, the system extracts knowledge from the dataset. An example of the graphical representation of the acquired knowledge is shown

```
Action Map 0

(General Conditions)

context (DayOfWeek (=,Monday,Tuesday,Wednesday,Thursday,Friday))&
context (TimeOfDay(>,08:09:37)) & context (TimeOfDay(<,08:32:17))

(Action Pattern 0)
ON occurs (simple,(OFF,KitchenDoor (0)), t0) Probability: 1.0
IF context()
THEN do (simple,(ON,KitchenDoor (1)), t) when: t = t0 + 1.0 s.

(Action Pattern 1)
ON occurs (simple,(OFF,Kettle (0)), t0) Probability: 1.0
IF context()
THEN do (simple,(OFF,KitchenDoor (0)), t) when: t = t0 + 10.0 s.

(Action Pattern 2)
ON occurs (simple,(ON,KitchenDoor (1) ), t0) Probability: 0
IF context ()
THEN do (simple,(ON,Kettle (0)), t) when: t = t0 + 1.0 s.
```

Figure 4.4: Example of a sample of knowledge extracted using LFPUBS from ADL dataset.

in Figure 4.3.3.1. Furthermore, Figure 4.4 shows the sample of the extracted knowledge in the L_{LFPUBS} language, which is the final product of the LFPUBS. Accordingly, the pattern received from LFPUBS is divided into two groups, Action Patterns and General Conditions, which are explained below.

General conditions Figure 4.4 illustrates the calendar information which given the boundaries with context will be satisfied through the respective Action Patterns. So, the general condition must be satisfied before executing the Action Patterns.

Action Pattern Action Patterns represent every link between two nodes or events, forming four parts (Pattern ID, Event, Condition and Action).

Pattern ID The Pattern ID is a unique number that identifies a pattern in a sure Action Map. (Action Pattern 0)

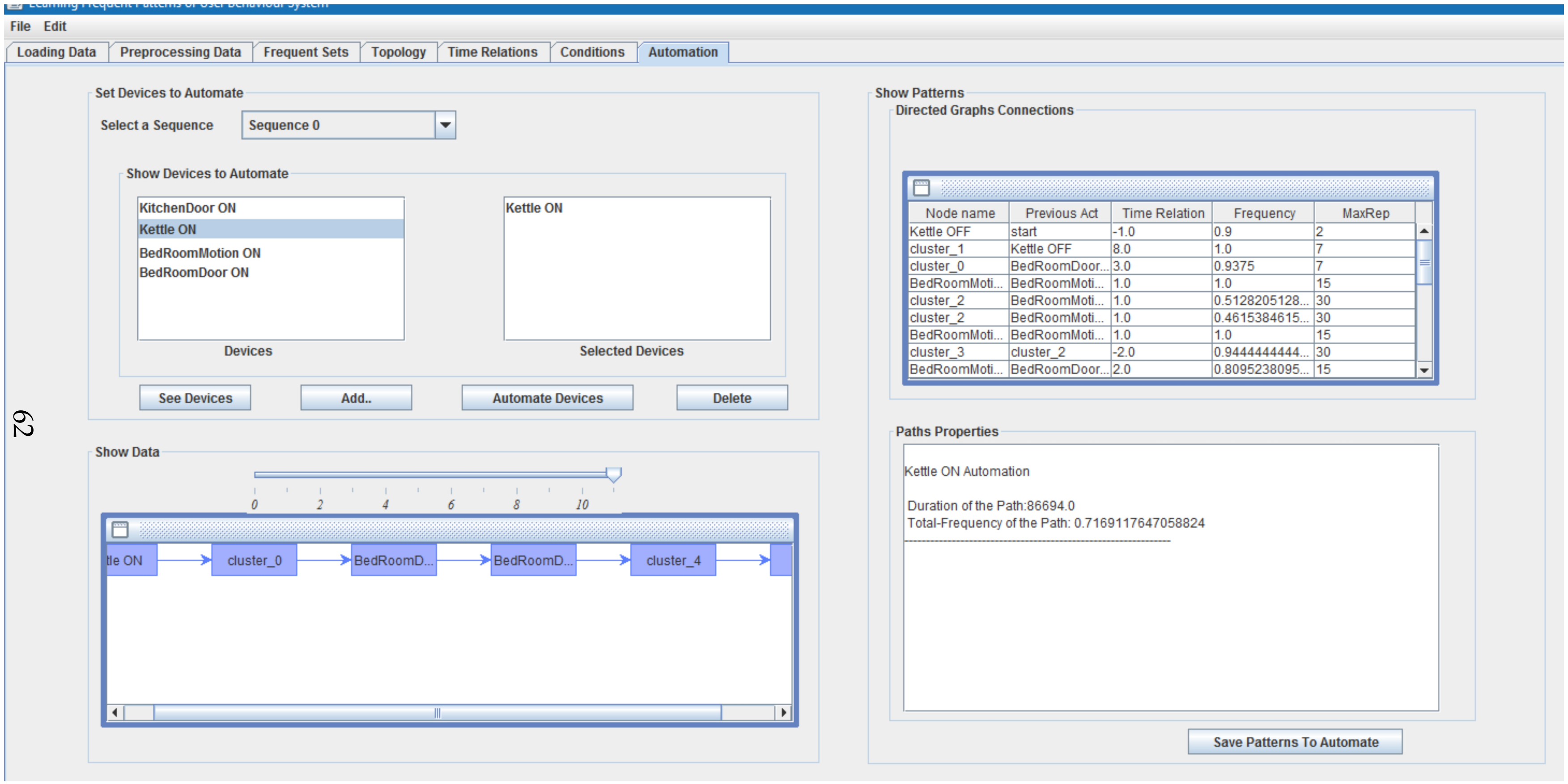


Figure 4.5: Graphical representation of knowledge discovered using LFPUBS.

Event The pattern defined by the ON clause describes the event that occurs and triggers the relationship specified in the pattern. The components of an event are the device or sensor implied in action, the nature of the action or sensor's status and the timestamp of such an action. The number between the brackets after the device represents the nature of this device. If one single step has been performed several times but with different nature, the number between the brackets will be different.

Condition The IF clause defines the necessary conditions under which the action specified in the THEN clause is appropriate to the event listed in the ON clause. Because it is almost impossible for an Event-Action to be valid under any condition, specific conditions are necessary to represent accurate patterns. Therefore, several types of conditions are provided, and can be split into two types: • Information coming from sensors: Temperature, humidity, pressure • Calendar information: "Time of Day" or "Day of Week". Priority is defined as how many cycles the system has spent to get that condition, the higher the priority, the more frequent or obvious the condition for the system.

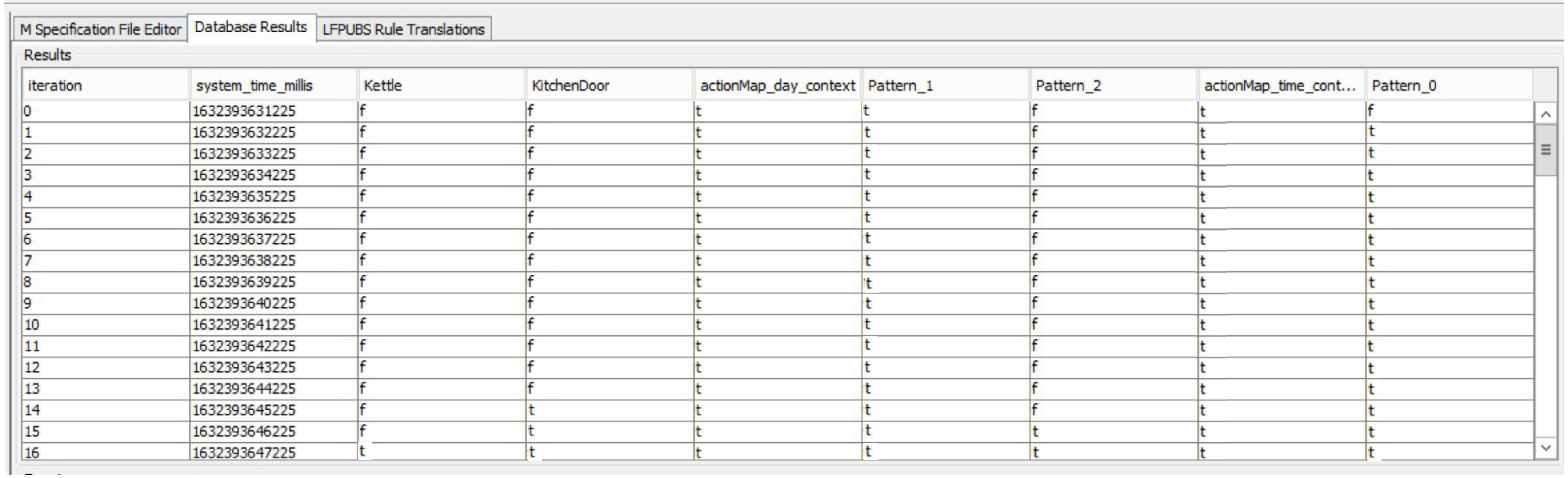
Action Finally, the THEN clause defines the user's action usually carried out by giving the ON clause and satisfies the conditions specified in the IF clause. The Time Relation between the Event and Action can be either quantitative or qualitative, with the usefulness of each type of relationship being different. Depending on the nature of the action, it will be translated differently. If the action is a sensor, it cannot be automated; thus, the objective of this pattern is to monitor the behaviour of the user. If the action is an actuator, the device is activated or deactivated by this pattern.

4.3.3.2 | Translating LFPUBS Pattern to "M" Rules

As seen from section 4.3.2, the MReasoner have three functionalities and one of them is automated rules translation. However, there was a drawback to this part of the system, which was overcome by Aranbarri-Zinkunegi [14] by introducing an updated version of LFPUBS2M. LFPUBS2M is a translator that works as a bridge between LFPUBS and MReasoner enabling the conversion of LFPUBS generated patterns into "M" rules, see Figure 4.6.

```
states( Kettle , KitchenDoor , actionMap_day_context , Pattern_1 , Pattern_2 , actionMap_time_context ,
Pattern_0 );

is( KitchenDoor );

holdsAt( #Kettle ,0 );
holdsAt( #KitchenDoor ,0 );
holdsAt( #actionMap_day_context ,0 );
holdsAt( #Pattern_1 ,0 );
holdsAt( #Pattern_2 ,0 );
holdsAt( #actionMap_time_context ,0 );
holdsAt( #Pattern_0 ,0 );

ssr( ( weekDayBetween(monday-friday ) ) ->actionMap_day_context ) ;
ssr( ( #weekDayBetween(monday-friday ) ) ->#actionMap_day_context );
ssr( ( clockBetween(08:09:37-08:32:17 ) ) -> actionMap_time_context );
ssr( ( #clockBetween(08:09:37-08:32:17 ) ) -> #actionMap_time_context );


ssr( ( [-][00:00:01]KitchenDoor ^ actionMap_time_context ^ Pattern_1 ^ actionMap_day_context ) -> Pattern_0 );
ssr( ( [-][00:00:10]#Kettle ^ actionMap_time_context ^ actionMap_day_context ) -> Pattern_1 );
ssr( ( [-][00:00:01]KitchenDoor ^ actionMap_time_context ^ Pattern_0 ^ actionMap_day_context ) -> Pattern_2 );
ssr( (Pattern_2 ) ->Kettle );
```

Figure 4.6: LFPUBS pattern converted to "M" rules.



iteration	system_time_millis	Kettle	KitchenDoor	actionMap_day_context	Pattern_1	Pattern_2	actionMap_time_cont...	Pattern_0
0	1632393631225	f	f	t	t	f	t	f
1	1632393632225	f	f	t	t	f	t	t
2	1632393633225	f	f	t	t	f	t	t
3	1632393634225	f	f	t	t	f	t	t
4	1632393635225	f	f	t	t	f	t	t
5	1632393636225	f	f	t	t	f	t	t
6	1632393637225	f	f	t	t	f	t	t
7	1632393638225	f	f	t	t	f	t	t
8	1632393639225	f	f	t	t	f	t	t
9	1632393640225	f	f	t	t	f	t	t
10	1632393641225	f	f	t	t	f	t	t
11	1632393642225	f	f	t	t	f	t	t
12	1632393643225	f	f	t	t	f	t	t
13	1632393644225	f	f	t	t	f	t	t
14	1632393645225	f	t	t	t	f	t	t
15	1632393646225	f	t	t	t	t	t	t
16	1632393647225	t	t	t	t	t	t	t

Figure 4.7: Example of a database result after MReasoner executed the "M" rules.

4.3.3.3 | MReasoner for automation

The technical infrastructure of the MReasoner has been described in section 4.3.2. This section demonstrates the scenario of how MReasoner uses "M" rules to automate the system. As mentioned, MReasoner IDE three ways developing the scenario where this project uses the real-environment execution.

MReasoner uses a database system to save all the devices to the log reader. So, before executing the system, all the new devices (sensor, actuator) should be added to the system database.

The system checks the actuator status first after loading the rules obtained from LFPUBS2M, in this case Figure 4.3.3.2, the Kettle 0. Later, the MReasoner checks the rules one by one. It first checks the calendar information. After satisfying the information, MReasoner checks the incoming events from the log reader. As MReasoner works in a real-world environment, the system constantly examines the log files of the Vera-router checking for the current status of devices. Iteration one shows the kettle false, kitchen door false, actionMap_day_context true, actionMap_time_Context true, Pattern_0 false, Pattern_1 true and Pattern_2 false. Consequently, iteration two shows the same except Pattern_0. In iteration two, Pattern_0 is true because Pattern_1 is true. Now the iteration waits for the kitchen door to open. Iteration fourteen KitchenDoor true. Hence, iteration fifteen Pattern_2 true, which triggered the Actuator Manager module to call the service in the router to switch on the Kettle in iteration sixteen.

Recall from beginning of section 4.3.3 in Table 4.5, the first column lists the activity name detected to provide the expected services. In the test results evaluated for user A, the bedroom light does not turn on when the user A wakes up. Furthermore, we can see that both the "Make tea" and "Go outside" activity triggering times were late. Careful analysis revealed that the bedroom light did not turn on because of a faulty PIR (passive infrared) sensor and MReasoner failed to make the automation delay.

Table 4.5: The results received after evaluating the scenario.

Activity name	User	Expected services	Service received
Wake up	user A	The bedroom light on when user wake up	No light on
	user B	The bedroom light on when user wake up	Light on
Use bathroom	user A	Bathroom light on when user wants to use the bathroom	Light on
	user B	Bathroom light on when user wants to use the bathroom	Light on
Use shower	user A	Shower light on when user go for a shower	N/A
	user B	Shower light on when user go for a shower	Light on
Make tea	user A	Kettle on when the user decides to make tea	Delay to trigger the kettle
	user B	Kettle on when the user decides to make tea	Kettle on
Go outside	user A	All house light turn off when user left the house	Delay to turn off all the light
	user B	All house light turn off when user left the house	Light off

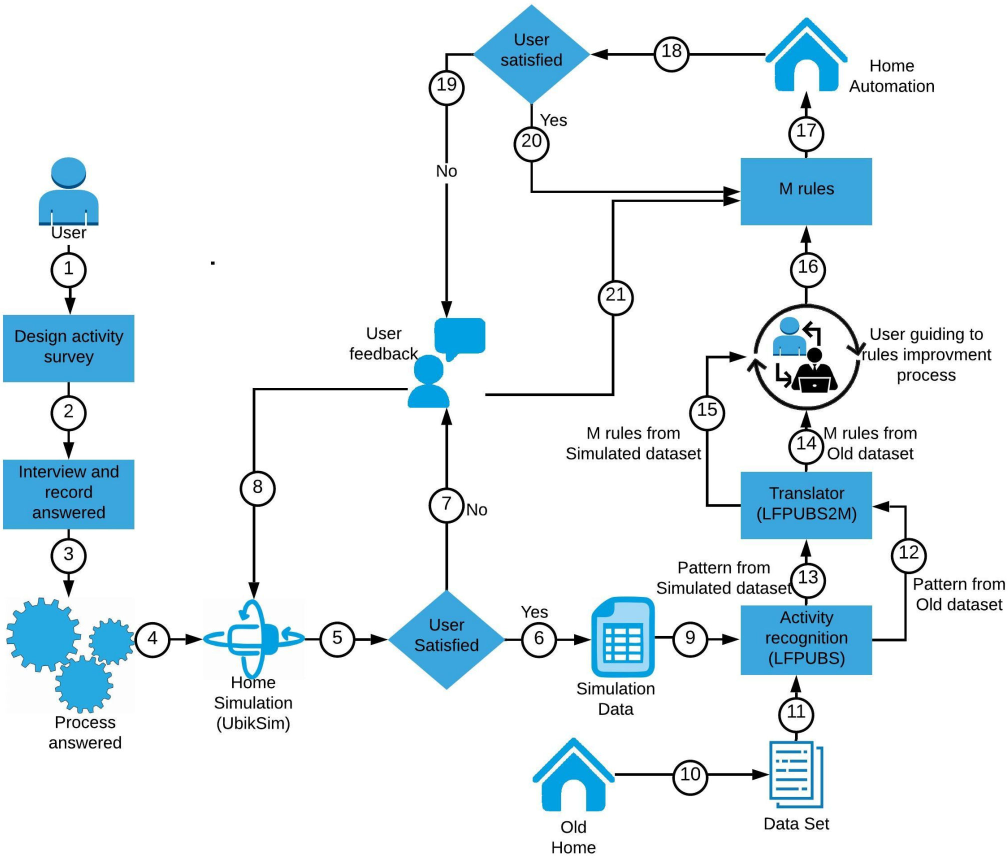


Figure 4.8: System Architecture: READY system extended to UTL.

4.4 | User-guided Transfer Learning (UTL)

The objective of the system is to provide automation services to the user as soon as the user starts living in a smart home. The MReasoner computational model ensures that automation happens through the execution of "M" rules.

As shown in Section 4.3, the system can initially provide a level of automation service with "M" rules selected based on prior experience and user preferences from a simulation dataset. However, a significant question now needs to be addressed. How accurate is the set of "M" rules in use, and can these rules be further modified to more efficiently satisfy the users' needs for smart home automation?

With transfer learning, a system can leverage experience from a previous task to improve the performance of the new task [45]. To answer our question, we propose a User-guided Transfer Learning (UTL) approach, where new house knowledge is improved by old house knowledge to increase overall automation functionality and effectiveness. We also utilise user knowledge and feedback to ensure that improvements are appropriate for the user.

It is important to note that different house shapes generate different data types. To limit the complexity of this problem, we assumed that the old smart home has the same layout as our new smart home with similar activities.

Figure 4.8 shows an overview of the process. First, the old smart home data need to be collected and processed. Data is then entered directly into the LFPUBS. LFPUBS extracts the most frequently observed patterns of user behaviour from the dataset. This pattern data is then passed to the LFPUBS2M translator to generate the relevant "M" rules.

Similarly, simulation data is also entered into the LFPUBS system to find the most frequent patterns that emerge from the simulation dataset. These patterns are also passed through the LFPUBS2M for translation into "M" rules.

Two sets of rules are ready to be considered; one set received from the simulation dataset (section 4.3) and another from the old smart home dataset. The developer analyses the old smart home dataset rules to the current set of rules one by one (details in Table 4.6). If the developer identifies a benefit, and if the user approves of the suggestion, the improvements will be implemented by the M system. User feedback can be sought and incorporated into future sets of upgraded rules as often as necessary.

4.4.1 | UTL evaluation with a real scenario

This section recalls the scenario from page section 4 for evaluation using the UTL method. Section 4.3.3 shows that the system is capable of providing the requisite level of smart home services to user A and user B.

The old smart home dataset for user A and user B is vital in continuing the evaluation process. In reality, however, for the experiment, it is often either impractical or impossible to find the old smart home data for user A and user B, which creates a data unavailability problem.

To overcome the old data unavailability problem, research participants performed the above scenario in the morning, in the Middlesex University Smart space Laboratory for four weeks and saved the resulting dataset to the server. Patterns were then generated from this dataset by LFPUBS (figure 4.8, step 12), and LFPUBS2M was used to translate these patterns to MReasoner rules (figure 4.8, step 14).

As described previously, one of the reasons that LFPUBS and MReasoner were selected is that the outputs produced are human-understandable. This feature critically supports the developer throughout the system development process.

To fully understand our proposed approach, we encourage the reader to focus on the "User guiding rules for improvement process" section in figure 4.8, where the developer and user sit together with two sets of rules. Developers consider only those rules which do not exist in a simulated set of rules. Considering that the user had minimal knowledge about the technology, the developer must then explain the purpose of each rule in user-friendly language and question the need for any new rules.

In table 4.6 (in Gray colour), the developer found a new set of rules showing that the user A takes a bath on Friday morning. So, at this point, the developer will ask the user A if they take a bath every Friday morning. If the answer is yes, then a new rule would be added to the system. If not, then a new rule would be unnecessary. However, user A may now decide that they want to add this facility as an automation service during the evaluation process. Alternatively, they could have omitted to specify this service in the first phase. Irrespective, the addition of a new automation rule by the developer is straightforward.

Table 4.6: The results received after evaluating the UTL.

Activity name	User	Expected services	Service received from simulated dataset	Service offered by old home dataset
Wake up	user A	The bedroom light on when user wake up	No light on	Light on
	user B	The bedroom light on when user wake up	Light on	Light on
Use bathroom	user A	Bathroom light on when user wants to use the bathroom	Light on	Light on
	user B	Bathroom light on when user wants to use the bathroom	Light on	Light on
Use shower	user A	Shower light on when user go for a shower	N/A	Turn on light on Friday
	user B	Shower light on when user go for a shower	Light on	Light on
Make tea	user A	Kettle on when the user decides to make tea	Delay to trigger the kettle	Kettle on without delay
	user B	Kettle on when the user decides to make tea	Kettle on	Kettle on
Go outside	user A	All house light turn off when user left the house	Delay to turn off all the light	Light off without delay
	user B	All house light turn off when user left the house	Light off	Turn house light off as user go outside morning at 5 a.m as well

For user B, the current rules state that they go outside every weekday at 5 AM. However, the user B confirmed that this rule would not be necessary at the new house. So, the developer does not add this rule into the system.

5

System testing and evaluation

The most valid and reliable method of testing within the Intelligent Environments (IE) area is to test a smart home automation solution in a real environment and observe user interactions over an extended period of time. Specifically, it is the interaction between the user and the system that makes it possible to assess whether the system indeed provides the promised services.

This paper has discussed various approaches to both testing and evaluating automated smart homes. This section delves into the experimental results gathered from a comprehensive test of the READY method.

Validation is a challenging endeavour because home automation systems are complex collections of sensors, networks, databases, humans, software, infrastructure, and environments. If any one of these elements fails, then the system as a whole will not produce the correct results. For example, in Section 4.3.3, the test failed due to a faulty Passive Infrared (PIR) sensor. For this reason, when testing a system of this level of complexity, it is imperative to consider each and every component of the system so that any system failures can be traced back to their source.

To address this need, Augusto et al. [16] introduced the COntext-Aware systems Testing validation (COATI) method. The COATI method considers a smart home to be a complete system with the resources needed to deliver services in a specific context. These resources are referred to as *enablers*.

The approach also proposes a table (called *check table*) that highlights the minimum system configuration required for a context-specific solution to work in order to avoid a situation where an element may fail. The COATI method was adopted for system testing of the current project. Section 5.1.2 illustrates in depth how COATI assists in testing the system. To test the prototype, we examined system performance in two scenarios, described below.

Scenario (Morning): The user wakes up, uses the bathroom, goes to the kitchen to make breakfast, eats breakfast, goes back to the bedroom, gets ready, and goes outside.

In this scenario, the user would expect several devices (e.g., lights, tea kettle) in the kitchen, dining room, bedroom, corridor, bathroom and shower to automatically switch on and off.

Scenario (Evening): The user comes back from the office, changes clothes, uses the bathroom, goes to the sitting room, reads the newspaper, goes to the kitchen, makes dinner, eats dinner, and goes to bed.

Again, the user would expect all automated devices to turn on when appropriate and switch off by the time they go to bed.

These scenarios were used to test the ability of the newly developed READY approach, in combination with user-guided transfer learning, to mitigate the cold start problem. Thus, a two-pronged testing strategy was undertaken, with the first focused on testing the READY method and the second focused on testing the UTL method.

5.1 | Solving the cold start problem using the READY approach

The READY approach was tested on the above two scenarios with a set of experimental tools that aimed to determine whether the system could satisfy the specified requirements. Although there are many ways to test smart home automation systems, the present study adopted a System Action approach (see LFPUBS Section 4.3.1) to observe user interaction.

The assessment was conducted within the Smart Spaces Lab at Hendon Campus in Middlesex University. The lab environment consisted of a house with a living room, (Fig 5.2), bathroom, shower, large bedroom (Fig 5.3) and office. Sensors were installed in the house shown in Figure 5.1.

Table 5.1: Name, type and quantity of sensors installed in the smart space.

Sensor's name	Sensor type	Sensor's quantity
Motion sensors	PIR	6
Door sensors	Door	11
Object sensors	Power	3
Light sensors	Light	6

Specifically, PIR motion sensors were installed in the kitchen, corridor, entrance, living room, bedroom, bathroom and shower to detect user movement. Additionally, door sensors were installed in every door of the house, including the doors of the kitchen cupboards and refrigerator, to detect when doors were opened or closed. These sensors were used as indicators of user interactions with the objects. For instance, BedroomLamp referred to the status of the lamp installed in the bedroom. Similarly, Kettle indicated the status of the tea kettle in the kitchen.

Five individuals were invited to take part in the system validation process, which was performed in accordance with Middlesex University data protection regulations. Prior to testing, each individual was provided with a basic knowledge about the smart home and the validation process.

Given that each automation solution is unique to a single user, only one participant could test the system at a time. To test the scenarios described above, participants were invited to the lab for both morning and evening sessions.

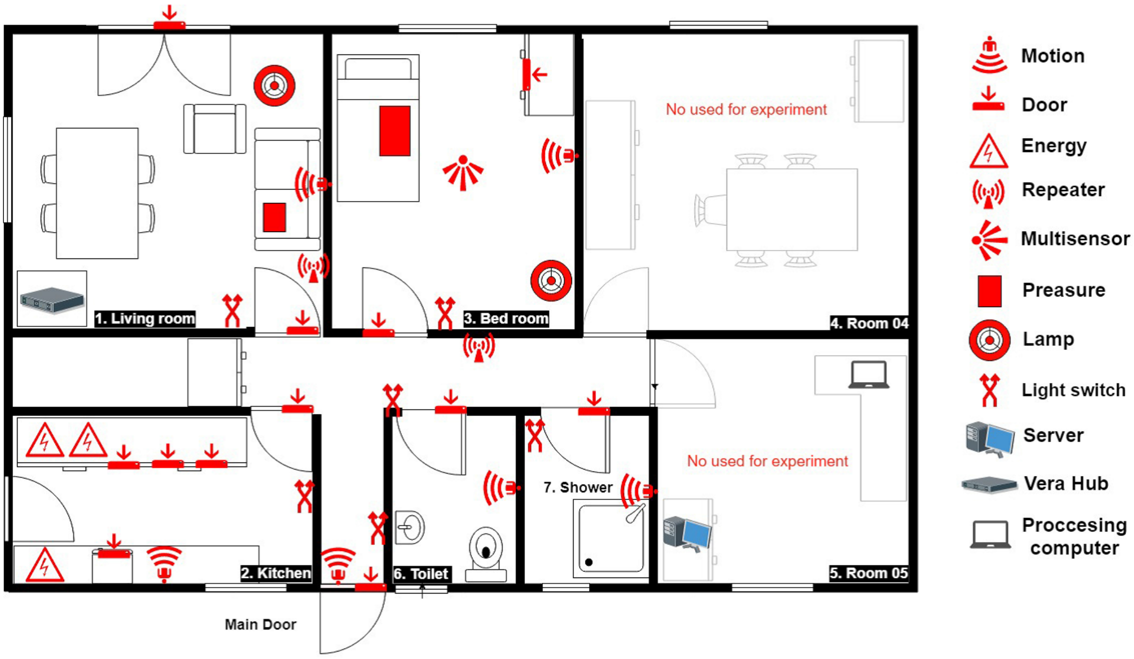


Figure 5.1: A map of the lab including sensor hardware.

Figure 5.2: Kitchen of the Smart Spaces Lab.

Figure 5.3: Bedroom of the Smart Spaces Lab.

5.1.1 | First session

In the first session, participants responded to a questionnaire (section 4.1) asking for information on how they perform morning and evening daily living activities. Although participants likely perform a variety of activities, only eight activities were selected for the purposes of testing. These activities exemplified tasks or behaviours that were presumed to be applicable to a majority of prospective users (see Table 5.2). Participant responses were recorded and organized (Table 5.3) to make the simulation development more straightforward.

Table 5.3 shows a simulation of participant behaviour in a virtual smart home. More details on the virtual smart home development and behaviour simulation process are available in section 4.2.

Table 5.2: The activities considered for the validation process.

Scenario	Simple Activity	Complex Activity
Morning	Wake up, use bathroom, use shower, go outside	Make tea
Evening	Enter home, Use bathroom, sleeping	Make tea, relaxing

In the second session, user behaviour was simulated (see Table 5.3). Then participants were invited to examine the simulation and record their feedback on how well it captured the way in which they would carry out those eight activities. Table 5.6 summarises the feedback obtained. For instance, as can be seen in Table 5.6, User A usually rested in bed for a period of 5 to 10 minutes after awakening. They left the house before 7:30 am and went to bed before 10 pm. User B accepted the simulation without any feedback. User C requested changing the timing and sequence of actions for making tea. The timing around returning home was also adjusted for User C based on their typical return time of about 8 pm. User D accepted the simulation without any complaint, though he did seek prior assurance that the simulation would correctly simulate the actions and timing associated with making tea and relaxing. Finally, User E accepted the simulation without any feedback.

As stated in section 4.2, the avatar carried out the activities in a predefined way. These passive interactions between the avatar and the virtual sensors were saved in a server for later use by LFPUBS (see section 4.3). In the following section, we explain the parameters used for this project.The performance of the avatar's predefined activities made it possible to generate labelled data. Therefore, LFPUBS knew in advance what knowledge it would uncover. Table 9 displays the sequence of activities carried out by the avatar.

Table 5.3: Participants answered organized to creating the simulation

User	Time range	Scenario	Activities and sequences
user A	08:30-09:30AM	Morning	Wake up → Use bathroom →Make tea →Go outside
	07:00-10:00PM	Evening	Enter home → Use bathroom →Relaxing →Make tea →Sleeping
user B	06:00-07:00AM	Morning	Wake up → Use bathroom →Make tea →Use shower →Go outside
	08.00-10.00PM	Evening	Enter home →Make tea→Relaxing→Make tea →Sleeping
user C	08:30-09:00AM	Morning	Wake up → Use bathroom →Use shower →Make tea →Go outside
	08:00-10:00PM	Evening	Enter home →Make tea →Sleeping
user D	06:00-07:00AM	Morning	Wake up → Use bathroom →Use shower →Make tea →Go outside
	08:00-11:00PM	Evening	Enter home → Use bathroom →Make tea →Sleeping
user E	08:30-09:00AM	Morning	Wake up → Use bathroom →Make tea →Use Shower →Go Outside
	08:00-10:00PM	Evening	Enter home → Make tea →Relaxing →Sleeping

LFPUBS, an activity recognition system, operates by identifying frequent patterns of user behaviour. More specifically, the LFPUBS system develops its topology by considering discovered repetitive actions. According to Aztiria et al. [27] , developed topologies cannot guarantee the inclusion of all frequent relationships because frequent relations are often discovered without first establishing a minimum support. Further, a relation may be classified as frequent so long as it reaches the pre-set confidence level, even if it occurs infrequently.

In contrast, frequent sets use a minimum confidence level that also functions as the minimum support level, meaning an action must fulfil the requirement of including minimum levels in a frequent set. This, in essence, is why frequent relations are excluded from topologies. The basic algorithm is used to calculate Time Relations. The number of Conditions was low due to a few of the context sensors used for the experiment. The purpose of automation is to find the most reliable automation path. Table 5.4 defines the parameters considered to achieve the most reliable path.

The main purpose of this session was to install the system with the rules received from the previous section in the Smart Space Lab, invite participants to perform the scenarios naturally, and observe that automation.

Before executing the tests, all sensors and actuators were inspected to ensure that they performed correctly. They were saved in the system database so they could be connected back to the device number of the log reader. Section 4.3 outlines the steps by which MReasoner executes the rules.

As mentioned at the beginning of this section, the COATI approach influenced the creation of a customised check table. Using this table, it was possible to ensure that all system components were working correctly, and to proceed by turning the focus to rule execution. Table 5.7 demonstrates an example for User A illustrating how the table was used to check each component of the system.

Table 5.4: LFPUBS parameters values to process the simulation dataset.

Activity Name	Confidence Level	Automation Device
Wake up	80%	BedroomLight
Use bathroom	85%	BathroomLight
Use shower	90%	ShowerLight
Make tea	90%	Kettle
Go outside	90%	CorridorLight
Enter home	85%	CorridorLight
Sleeping	80%	BedroomLight
Relaxing	90%	TableLamp

The first column shows the parameters that were considered; the second column catalogs enablers, which are specific contexts that require a certain number of resources from the infrastructure for the context to happen [16]; the third column shows the initial values of the enablers prior to testing; finally, the fourth column displays the number of tests needed to be conducted for the focal context. There is no particular limit to the number of tests required to determine a successful result for any of the experiments. Thus, the process continues until the test is successful. Table 5.7 shows the example for User A, for whom the first test failed due to a faulty kitchen movement sensor. After repairing the sensor, however, the test was successful.

Pertaining to user tests of the system, the five participants were invited one at a time into the lab. Before testing, the table was checked to ensure that all Enablers were recording the initial value. Then, participants performed their daily activities in any way they so choose. When the sequence of activities was complete, the MReasoner log was checked to identify and repeat any failed tests. This testing process produced ten total tables for the five users, which can be found online in [8].

After finishing this part of the assessment, we asked the participants six questions (Table 5.5) to measure their acceptance and satisfaction with the new system. The answer received from the user were analysed against a three-point Likert scale (Figure 5.4). After finishing this testing, participants were asked to answer six questions (Table 5.5) to measure their acceptance of and satisfaction with the new system on a three-point Likert scale (Figure 5.4).

Eighty percent of users responded that smart home automation was beneficial from day one. Further, the availability of the smart home simulation had helped the user adapt to their new home. Interestingly, users did not find substantial similarities between the simulated and real homes. However, they did observe that smart homes provided them with an idea of how the actual house would perform. A majority of users agreed that there were significant similarities between actual and expected simulation behaviours. In general, users were satisfied with the time taken for home automation performance. Sixty percent of users agreed that automation actions occurred within a reasonable time.

Table 5.5: Measuring the acceptance of the system and user satisfaction.

no	System acceptance and user satisfaction question
1	How useful is it that the smart home provides services from day one?
2	How similar were the simulated and real smart home solution?
3	How close was the simulated behaviour to the answers you provided?
4	How useful was the simulation in adjusting to the real house?
5	How well did the house provide its automation services?
6	What improvements would you make to the system?

5.1.2 | Enhance the understanding of the new home using the UTL approach

This part of the testing is supplementary to the previous part of the testing. Three of the original five participants participated in this part of the validation process, namely users A, B, and C.

In section 4.4.1, we discussed how we avoided the unavailability of the old data set by utilizing the dataset created by each user as they performed daily activities over four weeks in the Smart Space Lab. Data saved to the lab server was processed and used as an input for LFPUBS. The LFPUBS parameter was constant for all data sets in the first part of the validation process because when the simulator generated the data, we assured the data accuracy. On the other side, LFPUBS parameters changed to process different sets of real smart home data. Table 5.8 shows the LFPUBS parameters provided the most reliable automation path. The output result was used for LFPUBS2M to translate into M rules. The rules are available in [9].

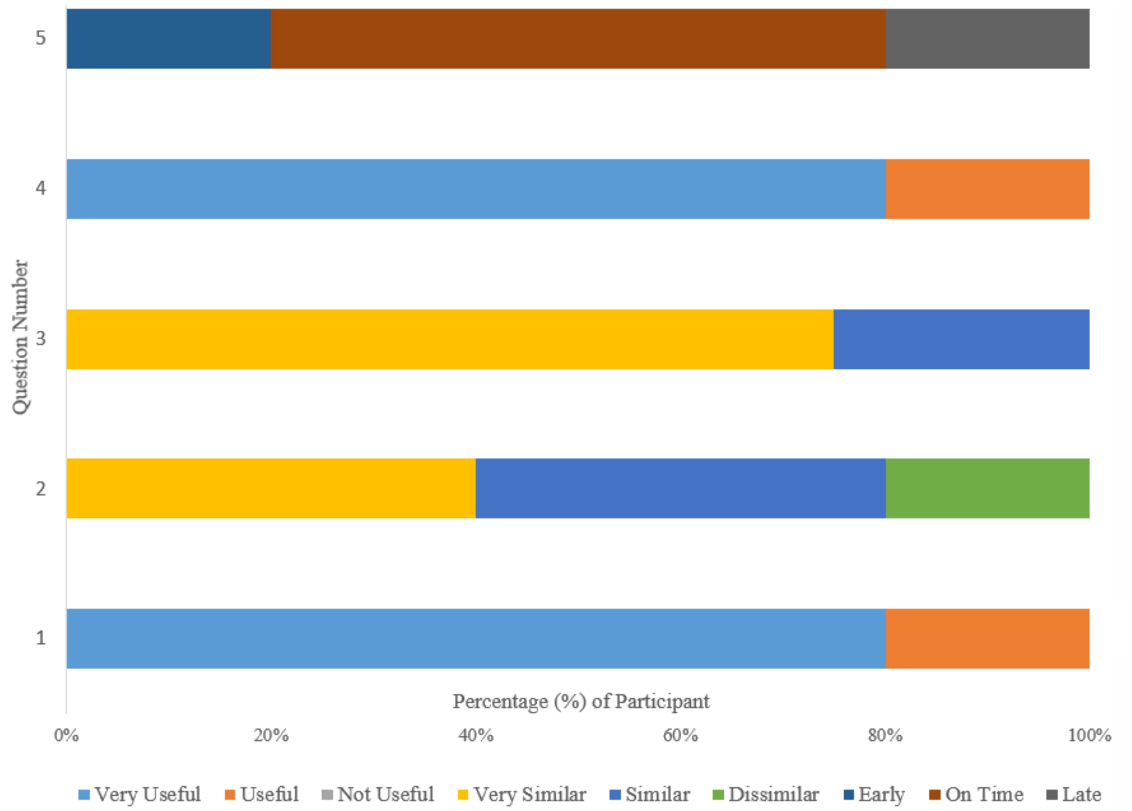


Figure 5.4: Users' responses based on questions in Table 5.5.

Table 5.6: The feedback received from the user's.

Activity Name	User A	User B	User C	User D	User E
Wake up	After waking up user wait 5-10 min on the bed	Accepted, no-feedback	Accepted, no-feedback	Accepted, no-feedback	Accepted, no-feedback
Use bathroom	Accept the simulation	Accepted	Accepted	Accepted	The sequence is not right, user drinks tea before goes to bathroom
Use shower	Not applicable	Accepted	Accepted	Accepted	Accepted but user has a concern about the duration of the activity use shower
Make tea	Accepted	Black Tea	Correct the sequence	Milk tea	accepted
Go outside	User left the house before 7:30 am	Accepted	Accepted	Accepted	Accepted
Enter home	Accepted	Accepted	User enter home after 8 PM at Monday	Accepted	Accepted
Sleeping	User sleep around 10 PM	Accepted	User does have any particular time to goes for sleeping	Accepted	Accepted
Relaxing	Relax in Bedroom	Relax on the setting room	sleeping	Relax on sofa	Relax on setting room

Table 5.7: Participants answered organized to creating the simulation

Morning Scenario	Enablers	Assumptions Initial Values	Test 1	Test 2
Context Description			Facilitate the user daily morning activities to automate the home equipment.	
Expected Outcome(s)			The lights in the bedroom, corridor, bathroom, kitchen, shower, and on the table switch on automatically when required, as well as the kettle in the kitchen. All automated devices will be switched off if the user forgets to switch them off before leaving the house.	
Real Outcome(s)			The kettle does not turn on	The user received the required services.
	EntranceMotion	EntranceMotion=0	EntranceMotion=1	EntranceMotion=1
	CorridorMotion	CorridorMotion=0	CorridorMotion=1	CorridorMotion=1
	BedroomMotion	BedroomMotion=0	BedroomMotion=1	BedroomMotion=1
	KitchenMotion	KitchenMotion=0	KitchenMotion=0	KitchenMotion=1
	BathroomMotion	BathroomMotion=0	BathroomMotion=1	BathroomMotion=1
	ShowerMotion	ShowerMotion=0	ShowerMotion=1	ShowerMotion=1
	SettingMotion	Not applicable for this context	Not applicable for this context	Not applicable for this context
Sensors	EntranceDoor	EntranceDoor=0	EntranceDoor=1	EntranceDoor=1
	FrezzerDoor	FrezzerDoor=0	FrezzerDoor=1	FrezzerDoor=1
	Cupboard	Cupboard=0	Cupboard=1	Cupboard=1
	Kettle	Kettle=0	Kettle=0	Kettle=1
	BathroomDoor	BathroomDoor=0	BathroomDoor=1	BathroomDoor=1
	SmallPaddle	Not applicable for this context	Not applicable for this context	Not applicable for this context
	BigPaddle	BigPaddle=1	BigPaddle=1	BigPaddle=1
	TableLamp	Not applicable for this context	Not applicable for this context	Not applicable for this context
Network	Z-wave (Vera hub)	Vera has a connection with sensors involved	There is a connection with all the sensors and update their value	There is a right connection
Database	Monitoring Database	Added the sensors and actuators to the database	Database updated, the sensors and actuators status value has been changed	Database updated, the sensors and actuators status value has been changed
Reasoner	Connection with sensors and server	The tools connect with Vera and MReasoner	The info from Vera is updating in the tool. There is a connection with the server.	The info from Vera is updating in the tool. There is a connection with the server.
User	user A	user A	user A	user A

Table 5.8: LFPUBS parameters values to process the real dataset.

Activities	User A	User B	User C	Automation device
Wake up	70%	70%	75%	BedroomLight
Use bathroom	95%	70%	80%	BathroomLight
Use shower	90%	70%	95%	howerLight
Make tea	90%	85%	90%	Kettle
Go outside	80%	85%	80%	CorridorLight
Enter home	90%	90%	80%	CorridorLight
Sleeping	70%	70%	90%	BedroomLight
Relaxing	70%	85%	90%	TableLamp

Now that we have two sets of rules, we can borrow the simulation dataset from the previous validation section and obtain another set of rules from the real dataset. We only consider those rules which are not available in the simulated set of rules.

After analysing the two sets of rules, we separated those rules that only exist in the rules of the real dataset to find out if the rules can provide new services. We invited each participant to register their interest in the new services. If a participant expressed interest, we created a new set of rules. Table 5.9 shows the new services proposed for each user. The modification rules are available in [9]. The new services are implemented to the home with considering the user acceptance.

Table 5.9: New Service detected from the old smart home dataset.

Participants	Services detect from real dataset
user A	User enter home at 4 PM on Friday
user B	No new service detected
user C	The system detect the bed room light off between 9-10 PM

5.1.3 | Discussion of the testing outcome

Smart home adaptation has been a popular research domain for a long time. Over this time, researchers have developed and used a wide range of approaches to improve the smart home adaptation process. Our system development approach engages the user in a way that makes it appear that the system subsequently developed was based upon a blueprint of the user requirements, empowering the smart home to provide specified services to its resident as soon as it is installed. Further, the user is so closely involved in the system development process that they are fully aware of the system's operation before they start living in the house.

In light of the results outlined in section 5.1, it is reasonable to conclude that the system can provide the smart home services to the user as soon as occupancy begins. A practical and satisfactorily performing home automation system increases user peace of mind and will increase the popularity of the smart home.

Researchers have been using simulation as a tool for the development of smart home services for more than a decade. Its use, however, has been limited to experimental purposes as it does not ordinarily come packaged as an integrated component of a home automation system.

The READY approach uses simulation for core system development. As an interface, simulation can use both observed and survey response behaviour and effectively transfer this behaviour to the system. In this research, we sought to highlight the effectiveness of various tools operating together to solve a specified problem rather than extol the benefits of specific tools, and it was for this reason, that we did not focus on developing sophisticated simulation tools. Ultimately, we discovered that users identified substantial differences between the simulation generated by UbikSim and the real home.

We did establish that simulation was sufficient to provide each user with a basic idea about the performance of automation services within their future house. Further, we found that a simulated solution was more effective in eliciting likely user behaviour in the real house and generating valuable datasets for pattern recognition and analysis.

LFPUBS is an ideal tool for identifying frequent and recurring patterns within datasets. The greater the dataset available for analysis, the better it was observed to work. LFPUBS finds it much easier to identify patterns within datasets generated by simulation than real datasets. This may be due to the absence of sensor-generated errors, such as a missed or incorrectly turned on or off sensor. We found that sufficient data to develop a system solution was generated by simulation, thereby meeting the experiment requirements.

Four weeks of daily activities data generated by user interaction with the home using the UTL approach was time-consuming but did reveal several exciting patterns that the simulation was unable to identify. These patterns became new automation services that the user tested during the next part of the testing (Table 5.9).

A review of user feedback (Figure 5.4) confirmed that the system was able to provide the required automation services successfully. However, two users observed that kettle automation did not happened as expected when making tea. Inspection of the relevant sensors revealed that the Kitchen movement sensor reset time was more than 20 sec. This incorrect parameter setting thus resulted in an unexpected delay within the automation process.

There were a number of challenges researchers encountered. The study was limited to a small group of user participants. The UbikSim and LFPUBS tools had to be extensively customised to ensure that they would work seamlessly together for this project. Each generated smart home dataset was unique to a single occupant. Constant monitoring of the system was necessary to ensure that the data set used to provide automation services was the correct one for the specific user.

5.2 | Evaluation

As discussed in earlier chapters, U-CIEDP is a system development method. The U-CIEDP methodology is centred on the end user and developed with the user's interests at heart. Specifically, the U-CIEDP model consists of several small loops that allow for system refinement based on user feedback. Unlike the U-CIEDP approach, the READY method does not go through the entire process at once. Instead, it integrates each element after it meets the target features. This can be seen, for example, in Figure 4.1, Step 5,7 & 8 where the simulator is depicted in a recursive loop with the objective of producing a synthetic dataset based on user feedback. In this way, the developed system is tested and validated by the user at each stage of development. Following the development of the READY method, five users were invited to test each component of the system at Middlesex University's Smart Space Lab.

A major goal of this section to evaluate the READY method by gathering feedback from end level users. To achieve this, professionals from different industries were invited to join the evaluation process. The system evaluation was conducted online due to COVID-19 travel restrictions. The evaluation approach was structured as follows:

Twelve external participants were involved in the evaluation process. Although the participants came from different industries, they were all involved, in some capacity, with smart homes. Five of the users were executives of smart home automation companies; five were care assistants; and two were researchers in the smart home domain. The system assessment process took place online. Prior to completing the assessment, the author created a consent form using Qualtrics[1] and sent a link to each user so they could give their consent. After consent was provided, a Zoom[2] link was distributed so each user could join an online and watched demonstration of the READY system.

Two sets of questionnaires were supplied to the participants during the assessment. The first section of the questionnaires was issued before the demonstration of the system while the second section of the questionnaires was issued after the demonstration. The questionnaire can be found in [7]. The assessment session took 40-60 minutes in total.

5.2.1 | Results

The first part of the questionnaires (Q1 & Q2) measured the participants' knowledge of smart homes. Over 90% of participants demonstrated an exceptional understanding of smart home technology. Moreover, 80% of participants were aware of the advanced capabilities of smart homes, such as detecting health emergencies and diseases and providing advice on lifestyle changes. In spite of these high levels of smart home knowledge at baseline, all of the participants

[1]https://www.qualtrics.com
[2]www.zoom.com

(i.e., 100%) believed that their knowledge of smart home services improved (Q5) after viewing the demonstration.

The next part of the questionnaires (Q3) measured the importance of user involvement in home automation. More than 80% of participants said it was extremely important to provide user guidance about how and when automation services should be performed.

Next, the questionnaires (Q4) measured the usefulness of the smart home system. All of the participants (i.e., 100%) believed it was vital to ensure that users had access to smart home services as soon as they moved into their home. This belief underscores both the need for and the value of the READY system, which was created with the intent of providing users with smart home services right from the outset.

Finally, the questionnaires touched on how effectively the system was in carrying out the user's desired activities. As mentioned, U-CIEDP is a system development method in which the user has an opportunity to provide input at every step of the process. This is done to decrease the likelihood of user complaints in the future. Indeed, survey responses showed that 95% of the participants in the evaluation felt confident that the system was able to effectively carry out the user's desired activities (Q6). One participant, however, felt unsure of this without being able to personally use the system.

5.2.2 | Discussion

As with the evaluation of the U-CIEDP method, the evaluation of the READY method involved a number of key stakeholders. This evaluation involved having participants observe a demonstration of the system online and subsequently provide feedback. The evaluation was based on three core criteria needed for a successful system—immediacy, personalisation and effectiveness. Each of these is discussed in turn below.

5.2.2.1 | Immediacy

Providing services right away was a key objective of the project. This need was confirmed in the evaluation, as all 12 participants strongly agreed that the system should start working as soon as moves into the home. The five home care assistants, in particular, stressed that placing an elderly person in a smart home that does not provide services instantly could put the occupant at risk. Relatedly, a few participants argued that that user acceptance would be limited if smart home services could not be initiated as soon as the user started living in the house. Although the READY system is capable of providing basic home automation as soon as a user moves in, future research should focus on adding advanced smart home services such as fault detection, anomaly detection and other related services. This would likely have a positive impact on user acceptance.

5.2.2.2 | Personalisation

Personalisation was another key objective of the project. READY uses the U-CIEDP approach where the user is at the core of system development. The READY approach begins with an interview with the user for the purpose of programming initial user requirements. Future system refinements, however, come directly from user feedback. In this way, the system both comes and remains personalized to its specific user. In reviewing survey responses, 100% of the participants believed that a user would be able to assist the developer in personalising the system. However, 80% of the participants expressed uncertainty over users' abilities to successfully control the automation process on their own. This is likely because a large majority (i.e., 80%) of those who participated in the evaluation were end-level users, meaning they may not have had the necessary depth of knowledge to make an informed judgement. However, the two participants who did have an intricate knowledge of smart home systems—the smart home researchers—agreed that the user would be able to independently control over the automation process.

5.2.2.3 | Effectiveness

The third criterion of importance was creating a system that could actually provide the services needed by the user. The READY approach was developed to be flexible, consisting of several loops which allow the system to be refined based on user input. With each successive piece of feedback, the system adapts, enabling it to become more effective at meeting user requirements. Importantly, this process treats the user as an active participant in the system development process, offering the user a sense of agency in adding or modifying the smart home features they most need. As a case in point, after the user has been living in the smart home for a short period of time, the user is asked if the system is working for them as desired. If the user has any concerns or desired changes in mind, then the rules of the system can be modified. In this way, the READY method keeps the user in the loop until the system is operating at a maximum efficiency. Nearly all the participants in the evaluation validated this point, with 95% expressing confidence that the READY system could effectively provide home services.

6

Discussion

This chapter provides a synopsis of the research presented in this thesis and a discussion of the overall achievements and outcomes of the study based on the aims and goals outlined in Section 1.4. This chapter will also delve into the key implications and limitations of this research.

The present research began with a detailed investigation of the existing literature on smart homes (explored in greater depth in [11]). Smart homes were originally introduced in the early 1990s to enhance users' quality of life by providing home automation services [69]. These services rely on technologies that have a unique ability that enables both accurate detection and classification of users' activities. The literature review revealed that there is a major gap pertaining the data-driven approaches to automation that should be addressed. Specifically, this relates to the need for an effective solution to the "cold start" problem, where a lack of input data in the initial days of house renders otherwise powerful machine learning algorithms incapable of providing personalised services to a user when they first move into their new home.

Indeed, the research sought to address the leading research question, which related to enriching the smart home system with user daily activity data prior to a user's move-in date so that the system would be able to provide home automation services immediately. To do so, an integrated system using a user-centric approach was developed that comprised survey, simulation, activity recognition and a transfer learning approach.

In Section 4.1, a survey was utilized that collects data related to user behaviour during common daily activities as indicated in the first research question (Objective A1-O1). This interview was conducted face-to-face with the user because errors made at this point had the potential to affect the final output of the system. However, following the user-centric approach, the user was always connected to the development process. Another advantage of a face-to-face interview is that sometimes during the interview the user does not have the necessary knowledge about the question, so he/she can be informed immediately. Therefore, face-to-face is the best solution to minimise misunderstanding. An example of this survey questionnaire is also available online [7].

As the main aim of the interview was to understand the daily routine and preferences of the user, the research presents a technique where the user has answered separately, focusing on two main criteria. The first of these is what activities are performed and for how long, and the second relates to how the activities are performed. The initial question identified the user's daily routine, and the second question identified preferences that helped answer the second research question (Objective A1-O2). Our proposed methodology also includes a technique capable of transforming interviewed answers with less effort and enabling developers to create a simulation more quickly and efficiently. Critically, this technique requires activities to be categorised into simple and complex activities and then named appropriately.

The research used a simulator (section 4.2) to generate the synthetic dataset (A2-O1). Any existing simulator would have been appropriate for this research; however, it was essential that it possess a few specific characteristics mentioned in section 3.2.2 to fulfil the research requirements. It was necessary for the developed simulator to perform at least three functions. First, it was important that the simulator simulated the replica of the target house as this helps the user become familiar with the new house. Afterwards, the simulator allows the developer to simulate the user's behaviour. Finally, the simulator should have a database feature to record all the activities in the database as synthetic data. This is linked with the third research question (objective A2-O2).

The generated synthetic data make it possible to assess whether the activity recogniser could accurately identify user behaviour patterns. The findings suggest a requirement for compatible activity recognition in section 3.2.3. Any tools can be used for this purpose, however, it is advised to use those tools which have the capacity to represent the data and acquire knowledge. This is in turn linked with the third research question.

It can, therefore, be said that the present research accomplished its primary goal of developing a smart home technology capable of providing users with personalised smart home services that function as intended from the moment they move in. However, it is essential to consider the reliability of this technology as its validation relies heavily on synthetic data. In order to address this issue, the research introduced a transfer learning technique known as the UTL method in section 4.4.1. This method leverages data from a previous task and applies the information to a new but similar task which addresses the fourth research question. Here, the UTL method was used to generate rules from actual (i.e., not synthetic) smart home data and transfer it to the new smart home context (A4-O1). This approach was combined with user feedback to determine the most suitable set of smart home automation rules (A4-O2). Hence, the results presented here reflect the validation of the READY approach based not only on synthetic data but also on actual user data.

Researchers investigated existing methodologies and tools through the transfer learning approach to proffer a solution to the cold start problem in the new smart home (see Chapter 2.3 for more detail). Our research mitigated this problem using the novel READY approach and the transfer learning approach with the user's contribution called UTL. As stated earlier, the READY and UTL approaches can use any tool as long as such a tool fulfils the requirements mentioned in Chapter 4. After careful study, we discovered that some tools are closely related to system requirements and, with little or no modification, can be helpful to system development. After gathering raw data from the user's daily activity, the READY approach requires a simulator that stimulates the targeted house and models the user's behaviour based on daily activity data. After careful analysis, the UbikSim [90] was selected, although it had not been developed specifically for smart homes. Despite its shortcomings, the core of this software was close to requirements, and we added several features which make it suitable for the READY. The most important of these features are: 1) adding a database that records all the activities performed by the avatar; 2) improving the sensors that allow the developer to customize the sensing area; and 3) making the automation process more user-friendly where the developer can draw a complete scenario based on their requirements.

The second most essential requirement for the READY approach is the observation of the knowledge discovery process in data. Several algorithms are available for knowledge discovery, but the READY approach mainly focuses on knowledge presentation alongside discovery. For this reason, the research uses the LFPUBS learning system that extensively discovers all aspects of user's behaviour (frequent actions, order (topology), time relations and conditions) from the data and represents them explicitly. Moreover, LFPUBS2M and Mreasoner are used as supplementary tools to confirm that the data-generated rules effectively provide the automation services and ultimately the system can improve the user's daily living experience in the new smart home.

As can be abstracted from the above discussion, the focus of this research was not on any tool or a single method (e.g., UbikSim, LFPUBS) but rather the combination of several of these methods and tools, which resulted in the READY approach. The READY approach offers key advancements in smart home technology by addressing the cold start problem and placing the user at the centre of system development. Specifically, the READY approach begins a fluid dialogue with the user. Afterwards, several small loops occur in system development. Each progression in the system development process utilizes user feedback to refine programming. In this way, each aspect of system development is validated by the user. The UTL method is another validation source, drawing on actual data to formulate new system rules.

One of the critical dimensions of the READY is that other teams can reuse the method partially or fully. For example, producing synthetic data could facilitate researchers to test their new algorithm. As mentioned before, the core steps of the method are flexible, so anyone can choose their desirable tools based on their requirements to build a new similar system.

Although it is clear that the READY approach has built-in validation capabilities, the present study also called on several users to test the system in person at Middlesex University. These users, individually, visited the Smart Space Lab where they performed a set of activities each day continuously for four weeks.

Unfortunately, only five users could test the system due to time and resource constraints. Future research, however, should aim to test the system with more users. Beyond the in-person validation, this study also conducted an online demonstration with 12 stakeholders, including smart home automation executives, smart home researchers, and home care assistants. These stakeholders showed tremendous appreciation for the system, reviewing it positively and expressing interest in visiting the lab in person.

Having the system validated with end users possessing certain conditions (people with dementia, for example) would have added more quality to the research. Regarding this matter, and as mentioned previously, the system was also validated by a group of participants who directly or indirectly work in the care home. They were impressed with what they saw and how the system could detect user behaviour from the moment users began living in the house. Despite the system being developed for all kinds of users, the system modification for the user with special needs will be more complex because the whole database needs to change when new services are added to the system.

7

Conclusions & Future Work

7.1 | Conclusions

Smart homes have rapidly become popular, particularly due to such advancements that enable them to provide more complex and more varied services. In addition, smart homes offer a variety of contributions to society with their emergency assistance systems, automated timers, security features, fall prevention features, and warnings as well as alerts that are automatically sent to facilities in urgent cases. These features have helped inspire confidence and trust in users who need home assistance; however, they also want to retain their independence and their concerned family members.

As mentioned, user activity recognition is the key mechanism through which smart homes infer the services which their user needs. Although research has progressed tremendously since activity recognition became a topic of note in the early 1990s, data-driven approaches remain unable to overcome the cold start problem, or the need for massive amounts of user-specific data as soon as a user moves into a new smart home. The research presented in this thesis investigates this area with the overarching intention to find a solution to this problem.

The research introduces and validates a novel useR-guided nEw smart home ADaptation sYstem (READY) approach that combines user feedback with machine learning algorithms to build a rich data set that enables a smart home to provide personalised services from the outset. The READY approach draws on the four core aspects of survey, simulation, activity recognition and transfer earning. At first, daily user activity data is collected using a survey. Next, user data is simulated. Following this, activity recognition is conducted (Chapter 4).

Besides offering a solution to the "cold start" problem plaguing data-driven approaches to smart home personalisation, one of the most potentially valuable contributions of this research is that it treats the user as a key stakeholder throughout the process. Users have the opportunity to provide feedback at each stage of development, which helps not only to familiarise users with their smart home system but also inspires trust and promotes the acceptance of smart home technology in general.

The approach was evaluated in real smart home setup, and a group of internal and external participants were involved in the evaluation process. The results demonstrated that all participants believed this system is essential for a new user adaptation to smart home (see Chapter 5). Over 95% of participants expressed confidence that READY could effectively provide the automation services as soon as a user starts living in the house. Furthermore, the participants were very impressed with the system's flexible capability in that it can be refined again and again using user feedback, thereby increasing confidence in using the system and its personalization according to their own preferences.

The thesis also presents and validates a new transfer learning technique (UTL) that compliments the READY approach. Typically, transfer learning techniques use old domain knowledge to improve the new domain. Instead, the UTL approach involves the user in improving the new domain, which increases the accuracy and assurance of READY approach.

The evaluation and validation of the research were carried out under the framework of the User-Centric Intelligent Environments Development Process. Several contributions can be reported related to this research project, but the main contribution can be summarised as follows:

- A literature survey discussed the existing approach to the new smart home adaptation.
- A technique is introduced to process user answers and generate data supporting the house to understand the new user's habit.

- The study represents the user's involvement in refining the system after its development, allowing the user to control the new smart home system.
- The study introduced user-tailored transfer learning to improve the new smart home adaptation process.

7.2 | Recommendations for further work

In an age where there is increasing availability of large amounts of data, machine learning has become a burgeoning area of research. This is especially true within the smart home domain, in which researchers are racing to developing the most accurate algorithm for activity recognition. The present study represents a crucial step forward in this quest. However, future research can build on this study by considering the following:

- *Conducting smart home simulations online with a web-based simulation tool.* In this study, the interview between the user and the developer on the user's daily activities was conducted face-to-face. However, costs (i.e., time and money) could be reduced by developing a web-based simulation tool and conducting interviews online.
- *Validating activity recognition and automation with several different tools.* The current study used LFPUBS and MReasoner for activity recognition and automation, which was the best option for this preliminary stage of research. As research progresses and more resources become available to invest in the development of the READY approach, it would be beneficial for researchers to test new and additional validation tools to ensure that the technology continues to work as intended.
- *Loosening the assumption that the source and target houses are configured in exactly the same way.* To minimise the complexity of the research in relation to the UTL method, source and target house configurations were assumed to be identical. However, in reality, this is generally not the case. Thus, to increase the ecological validity (i.e., generalizability) of the findings, future research should try to apply UTL to houses with different configurations.

- *Considering multiple user research.* The current study focused only on one user within a smart home. Given that users often live with others, future research should focus on expanding the READY approach to provide personalised services for multiple home users.
- *The system validates with users with certain conditions.* The system is developed for any user, and it is tested by several common users. However, people with special needs may demand services according to their particular requirements. Such services also can be accommodated with this approach. However, several iterations will be required in order to fulfil the user's exact expectations. In some cases, the iteration process is expensive, and the research does not undertake the system development cost, which could be an essential factor.
- *Potential to work in a real smart home.* The developed system has been tested and validated in a lab environment (Smart home lab) where the participants perform their daily activities based on a given scenario. The real home context is comparatively complex where the occupant performs their activities more naturally; however, the system is developed to consider that the user performs their daily activities based on their habit. Therefore, system testing in real smart homes could introduce new challenges.

References

[1] Human-centred design processes for interactive systems, International Organization for Standardization. 1999.

[2] Sweet home 3d, 2015. URL `http://www.sweethome3d.com/it/`.

[3] *United Nations. Department of Economic and Social Affairs, World Population Ageing*. United Nations Publications, 2015.

[4] Rakesh Agrawal and Ramakrishnan Srikant. Fast algorithms for mining association rules. In *Proc. of 20th Intl. Conf. on VLDB*, pages 487–499, 1994.

[5] Reza Akhavian and Amir Behzadan. Wearable sensor-based activity recognition for data-driven simulation of construction workers' activities. In *Winter Simulation Conference*, pages 3333–3344, 2015.

[6] Riyad Al-Shaqi, Monjur Mourshed, and Yacine Rezgui. Progress in ambient assisted systems for independent living by the elderly. *SpringerPlus*, 5(1):624, May 2016.

[7] S. M. Murad Ali. A new personalized smart home validation questionnaire. 2021. URL `https://figshare.com/s/a3e2e89ac64b0ef97a50`.

[8] S M Murad Ali. Validation 1:solving the cold start problem using the ready approach, Nov 2021. URL `https://figshare.com/s/bbbd061ae81f1c830b34`.

[9] S M Murad Ali. Validation 2: Enhance the understanding of the home using the utl approach, Nov 2021. URL `https://figshare.com/s/248d73602cee8e6ed9af`.

[10] S M Murad Ali. Questionnaire for smart home user. 2021. URL `https://figshare.com/s/7ab909838bf10a580fca`.

[11] S M Murad Ali, Juan Carlos Augusto, and David Windridge. A survey of user-centred approaches for smart home transfer learning and new user home automation adaptation. *Applied Artificial Intelligence*, 33(8):747–774, 2019. doi: 10.1080/08839514.2019.1603784. URL `https://doi.org/10.1080/08839514.2019.1603784`.

[12] Nasser Alshammari, Talal Alshammari, Mohamed Sedky, Justin Champion, and Carolin Bauer. Openshs: Open smart home simulator. *Sensors*, 17(5), 2017.

[13] Mohsen Amiribesheli and Abdelhamid Bouchachia. Smart homes design for people with dementia. In *International Conference on Intelligent Environments*, pages 156–159, 2015.

[14] M.E. Aranbarri-Zinkunegi. Improving the pattern learning system integrating the reasoning system. 2017.

[15] Damla Arifoglu and Abdelhamid Bouchachia. Activity recognition and abnormal behaviour detection with recurrent neural networks. *Procedia Computer Science*, 110:86–93, 2017.

[16] J. C. Augusto, M. Jose Quinde, and C. L. Oguego. Context-aware systems testing and validation. In *2019 10th International Conference on Dependable Systems, Services and Technologies (DESSERT)*, pages 7–12, 2019.

[17] Juan Carlos Augusto. The logical approach to temporal reasoning. *Artificial Intelligence Review*, 16(4):301–333, Dec 2001.

[18] Juan Carlos Augusto. The logical approach to temporal reasoning. *Artificial Intelligence Review*, 16(4):301–333, Dec 2001.

[19] Juan Carlos Augusto. A general framework for reasoning about change. *New Generation Computing*, 21(1):209–246, 2003.

[20] Juan Carlos Augusto. Ambient intelligence: The confluence of ubiquitous/pervasive computing and artificial intelligence. In Alfons J. Schuster, editor, *Intelligent Computing Everywhere*, pages 213–234, London, 2007. Springer London.

[21] Juan Carlos Augusto and Chris D Nugent. Smart homes can be smarter. In *Designing Smart Homes, The Role of Artificial Intelligence*, pages 1–15, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg.

[22] Gorka Azkune, Aitor Almeida, Diego López-de Ipiña, and Liming Chen. Combining users' activity survey and simulators to evaluate human activity recognition systems. *Sensors*, 15 (4):8192–8213, 2015.

[23] Asier Aztiria. Learning frequent behaviours of the users in intelligent environments. *J. Ambient Intell. Smart Environ.*, 2:435–436, 2010.

[24] Asier Aztiria and Juan Carlos Augusto. Context-aware discovery of human frequent behaviours through sensor information interpretation. In Hans W Guesgen and Stephen Marsland, editors, *Human behavior recognition technologies: Intelligent applications for monitoring and security*, volume 2, pages 14–32, 2013.

[25] Asier Aztiria, J. Augusto, A. Izaguirre, and D. Cook. Learning accurate temporal relations from user actions in intelligent environments. In *Ubicomp 2009*, 2009.

[26] Asier Aztiria, A. Izaguirre, R. Basagoiti, J. Augusto, and D. Cook. Discovering frequent sets of actions in intelligent environments. In *Intelligent Environments*, 2009.

[27] Asier Aztiria, Juan Carlos Augusto, Rosa Basagoiti, Alberto Izaguirre, and Diane J Cook. Learning frequent behaviors of the users in intelligent environments. In *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, volume 43, pages 1265–1278, 2013.

[28] UABUA Bakar, Hemant Ghayvat, SF Hasanm, and SC Mukhopadhyay. Activity and anomaly detection in smart home: A survey. In *Next Generation Sensors and Systems*, pages 191–220. 2016.

[29] Matthew Ball and Vic Callaghan. *Managing Control, Convenience and Autonomy - A Study of Agent Autonomy in Intelligent Environments.*, volume 12 of *Ambient Intelligence and Smart Environments*, pages 159–196. IOS Press, Amsterdam, The Netherlands, 2012.

[30] Ling Bao and Stephen Intille. Activity recognition from user-annotated acceleration data. *Pervasive computing*, pages 1–17, 2004.

[31] Bruno Bouchard, Sylvain Giroux, and Abdenour Bouzouane. A smart home agent for plan recognition of cognitively-impaired patients. *Journal of Computers*, 1(5):53–62, 2006.

[32] Serge Thomas Mickala Bourobou and Younghwan Yoo. User activity recognition in smart homes using pattern clustering applied to temporal ann algorithm. *Sensors*, 15(5):11953–11971, 2015.

[33] Oliver Brdiczka, James L Crowley, and Patrick Reignier. Learning situation models in a smart home. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 39(1): 56–63, 2009.

[34] Allah Bux, Plamen Angelov, and Zulfiqar Habib. Vision based human activity recognition: a review. In *Advances in Computational Intelligence Systems*, pages 341–371. 2017.

[35] Roberto Casas, Rubén Blasco Marín, Alexia Robinet, Armando Roy Delgado, Armando Roy Yarza, John Mcginn, Richard Picking, and Vic Grout. User modelling in ambient intelligence for elderly and disabled people. In *International Conference on Computers for Handicapped Persons*, pages 114–122, 2008.

[36] S. Ceccacci and M. Mengoni. Designing smart home interfaces: Traditional vs virtual prototyping. In *Proceedings of the 10th International Conference on PErvasive Technologies Related to Assistive Environments*, PETRA '17, pages 67–74, New York, NY, USA, 2017. ACM.

[37] Silvia Ceccacci, Michele Germani, and Maura Mengoni. A method to design a smart home interface. In *Smart Product Engineering: Proceedings of the 23rd CIRP Design Conference, Bochum, Germany, March 11th - 13th, 2013*, pages 915–925, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.

[38] Olivier Chapelle, Vladimir Vapnik, and Y. Bengio. Model selection for small sample regression. *Machine Learning*, 48, 12 2000. doi: 10.1023/A:1013943418833.

[39] Liming Chen, Chris D Nugent, and Hui Wang. A knowledge-driven approach to activity recognition in smart homes. *IEEE Transactions on Knowledge and Data Engineering*, 24(6): 961–974, 2012.

[40] Siyun Chen, Ting Liu, Feng Gao, Jianting Ji, Zhanbo Xu, Buyue Qian, Hongyu Wu, and Xiaohong Guan. Butler, not servant: A human-centric smart home energy management system. *IEEE Communications Magazine*, 55(2):27–33, 2017.

[41] Yi-ting Chiang and Jane Yung-jen Hsu. Knowledge transfer in activity recognition using sensor profile. In *Ubiquitous Intelligence & Computing and 9th International Conference on Autonomic & Trusted Computing (UIC/ATC), 2012 9th International Conference on*, pages 180–187, 2012.

[42] Yi-Ting Chiang, Ching-Hu Lu, and Jane Yung-Jen Hsu. A feature-based knowledge transfer framework for cross-environment activity recognition toward smart home applications. *IEEE Transactions on Human-Machine Systems*, 2017.

[43] Nihan Kesim Cicekli and Yakup Yildirim. Formalizing workflows using the event calculus. In Mohamed Ibrahim, Josef Küng, and Norman Revell, editors, *Database and Expert Systems Applications: 11th International Conference, DEXA 2000 London, UK, September 4–8, 2000 Proceedings*, pages 222–231, Berlin, Heidelberg, 2000. Springer Berlin Heidelberg.

[44] Diane Cook and Sajal Das. *Smart Environments: Technology, Protocols and Applications (Wiley Series on Parallel and Distributed Computing)*. Wiley-Interscience, Hoboken,New Jersey, 2004.

[45] Diane Cook, Kyle D Feuz, and Narayanan C Krishnan. Transfer learning for activity recognition: A survey. *Knowledge and information systems*, 36(3):537–556, 2013.

[46] Diane J. Cook. Multi-agent smart environments. *Journal of Ambient Intelligence and Smart Environments*, 1(1):51–55, 2009.

[47] Kyle Dillon Feuz and Diane J. Cook. Heterogeneous transfer learning for activity recognition using heuristic search techniques. *International Journal of Pervasive Computing and Communications*, 10(4):393–418, 2014.

[48] A. Galton and J.C. Augusto. Stratified causal theories for reasoning about deterministic devices and protocols. In *Proceedings Ninth International Symposium on Temporal Representation and Reasoning*, pages 52–54, 2002. doi: 10.1109/TIME.2002.1027473.

[49] Konstantinos Giannakouris et al. Ageing characterises the demographic perspectives of the european societies. *Statistics in focus*, 72:2008, 2008.

[50] Sebastian Glende, Beatrice Podtschaske, and Wolfgang Friesdorf. Senior user integration: Ein ganzheitliches konzept zur kooperation von herstellern und älteren nutzern während der produktentwicklung. In *Conference: Ambient Assisted Living–AAL–2. Deutscher AAL-Kongress mit Ausstellung/Technologien–Anwendungen–Management*, volume 27, page 28, 2009.

[51] Valerie Guralnik and Karen Zita Haigh. Learning models of human behaviour with sequential patterns. In *Proceedings of the AAAI-02 workshop "Automation as Caregiver"*, pages 24–30, 2002.

[52] Victoria Haines, Martin Maguire, Catherine Cooper, Val Mitchell, Fran Lenton, Hina Keval, and CA Nicolle. User centred design in smart homes: research to support the equipment and services aggregation trials. Loughborough University Institutional Repository, Loughborough, 2005.

[53] Nils Y Hammerla, Shane Halloran, and Thomas Plötz. Deep, convolutional, and recurrent models for human activity recognition using wearables. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*, pages 1533–1540, 2016.

[54] M. Hanson. I'd rather die than be a burden on my daughter like many older people. url= https://www.theguardian.com/commentisfree/2016/dec/15/old-people-dementia-deathsocial-care-costs.html, lastchecked = December 15, 2016, 2016.

[55] Rim Helaoui, Daniele Riboni, and Heiner Stuckenschmidt. A probabilistic ontological framework for the recognition of multilevel human activities. In *Proceedings of the ACM international joint conference on Pervasive and ubiquitous computing*, pages 345–354, 2013.

[56] Brandon Ho, Dieter Vogts, and Janet Wesson. A smart home simulation tool to support the recognition of activities of daily living. In *Proceedings of the South African Institute of Computer Scientists and Information Technologists 2019*, SAICSIT '19, New York, NY, USA,

2019. Association for Computing Machinery. ISBN 9781450372657. doi: 10.1145/3351108.3351132. URL https://doi.org/10.1145/3351108.3351132.

[57] Derek Hao Hu and Qiang Yang. Transfer learning for activity recognition via sensor mapping. In *IJCAI Proceedings-International Joint Conference on Artificial Intelligence*, volume 22, page 1962, 2011.

[58] Ali Hussein, Mehdi Adda, Mirna Atieh, and Walid Fahs. Smart home design for disabled people based on neural networks. *Procedia Computer Science*, 37:117–126, 2014.

[59] Amy S Hwang and Jesse Hoey. Smart home, the next generation: Closing the gap between users and technology. In *AAAI Fall Symposium: Artificial Intelligence for Gerontechnology*, 2012.

[60] Unai Alegre Ibarra, Juan Carlos Augusto, and Asier Aztiria Goenaga. Temporal reasoning for intuitive specification of context-awareness. In *2014 International Conference on Intelligent Environments*, pages 234–241, 2014. doi: 10.1109/IE.2014.44.

[61] M.M Jakovljevi, A Njegu, and N Donov. Data-driven human activity recognition in smart environments. In *International Scientific Conference on ICT and E-Business Related Research*, pages 94–99, 2016.

[62] Tony Jebara. Generative versus discriminative learning. In *Machine Learning: Discriminative and Generative*, pages 17–60, Boston, MA, 2004. Springer US.

[63] Mohammadali Heidari Jozam, Erfaneh Allameh, Bauke de Vries, and Harry Timmermans. Prototype evaluation of a user centered design system for smart homes. In *Proceedings of the Conference on Design and Decision Support Systems in Architecture and Urban Planning*, pages 27–29, 2012.

[64] Cyryl Krzyska. Smart house simulation tool. Master's thesis, Technical University of Denmark, DTU, Lyngby, Denmark, 2006.

[65] Zhang Lei, Suo Yue, Chen Yu, and Shi Yuanchun. SHSim: An OSGI-based smart home simulator. In *IEEE International Conference on Ubi-media Computing*, pages 87–90, 2010.

[66] Gerhard Leitner. The future home is wise, not smart. In *The Future Home is Wise, Not Smart*. Springer, Cham, Springer International Publishing Switzerland, 2015.

[67] Jumphon Lertlakkhanakul, Jin Won Choi, and Mi Yun Kim. Building data model and simulation platform for spatial interaction management in smart home. *Automation in Construction*, 17(8):948–957, 2008.

[68] Li Liu, Yuxin Peng, Ming Liu, and Zigang Huang. Sensor-based human activity recognition system with a multilayered model using time series shapelets. *Knowledge-Based Systems*, 90: 138–152, 2015.

[69] R Lutolf. Smart home concept and the integration of energy meters into a home based system. In *International Conference on Metering Apparatus and Tariffs for Electricity Supply*, pages 277–278, 1992.

[70] Susan Mckeever, Juan Ye, Lorcan Coyle, Chris Bleakley, and Simon Dobson. Activity recognition using temporal evidence theory. *Journal of Ambient Intelligence and Smart Environment*, 2(3):253–269, aug 2010.

[71] Homay Danaei Mehr, Huseyin Polat, and Aydin Cetin. Resident activity recognition in smart homes by using artificial neural networks. In *International Smart Grid Congress and Fair*, pages 1–5, 2016.

[72] Thomas B Moeslund, Adrian Hilton, and Volker Krüger. A survey of advances in vision-based human motion capture and analysis. *Computer vision and image understanding*, 104(2): 90–126, 2006.

[73] Mueller-Freitag Moritz. 10 data acquisition strategies for startups, medium.com,. 2016. URL `https://medium.com/@muellerfreitag/10-data-acquisition-strategies-for-startups-47166580ee48#.u0cugwuwy`.

[74] Qin Ni, Ana Belén, García Hern, and Iván Pau De La Cruz. The elderly's independent living in smart homes: A characterization of activities and sensing infrastructure survey to facilitate services development. *Sensors*, 15(5):11312–11362, 2015.

[75] Mahesh Pal and Paul M. Mather. Decision tree based classification of remotely sensed data. In *Proceedings of the 22nd Asian Conference of Remote Sensing*, Singapore, November 2001.

[76] Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10):1345–1359, 2010.

[77] François Portet, Michel Vacher, Caroline Golanski, Camille Roux, and Brigitte Meillon. Design and evaluation of a smart home voice interface for the elderly: Acceptability and objection aspects. *Personal Ubiquitous Comput.*, 17(1):127–144, jan 2013.

[78] Sira Panduranga Rao and Diane J. Cook. Predicting inhabitant action using action and task models with application to smart homes. *International Journal on Artificial Intelligence Tools*, 13:81–100, 2004.

[79] Parisa Rashidi and Diane J Cook. Keeping the resident in the loop: Adapting the smart home to the user. *IEEE Transactions on systems, man, and cybernetics-part A: systems and humans*, 39(5):949–959, 2009.

[80] Parisa Rashidi and Diane J Cook. Transferring learned activities in smart environments. In *Intelligent Environments*, pages 185–192, 2009.

[81] Parisa Rashidi and Diane J Cook. Activity recognition based on home to home transfer learning. In *Workshops at the Twenty-Fourth AAAI Conference on Artificial Intelligence*, 2010.

[82] Parisa Rashidi and Diane J Cook. Multi home transfer learning for resident activity discovery and recognition. *KDD Knowledge Discovery from Sensor Data*, pages 56–63, 2010.

[83] Vijay Kumar Ravishankar, Winslow Burleson, and Diane Mahoney. Smart home strategies for user-centered functional assessment of older adults. *International Journal of Automation and Smart Technology*, 5(4):233–242, 2015.

[84] Vincent Ricquebourg, David Menga, David Durand, Bruno Marhic, Laurent Delahoche, and Christophe Loge. The smart home concept : our immediate future. In *2006 1ST IEEE International Conference on E-Learning in Industrial Electronics*, pages 23–28, 2006. doi: 10.1109/ICELIE.2006.347206.

[85] Ana-Maria Salai, Glenda Cook, and Lars Erik Holmquist. Intravox: A personalized human voice to support users with complex needs in smart homes. In Carmelo Ardito, Rosa Lanzilotti, Alessio Malizia, Helen Petrie, Antonio Piccinno, Giuseppe Desolda, and Kori Inkpen, editors, *Human-Computer Interaction – INTERACT 2021*, pages 223–244, Cham, 2021. Springer International Publishing. ISBN 978-3-030-85623-6.

[86] Ana-Maria Salai, Glenda Cook, and Lars Erik Holmquist. Situated buttons: A user interface to support users with complex needs and promote independent living. CHI EA '21, New York, NY, USA, 2021. Association for Computing Machinery. ISBN 9781450380959. doi: 10.1145/3411763.3451828. URL `https://doi.org/10.1145/3411763.3451828`.

[87] Claude Sammut and Geoffrey I. Webb, editors. *Expectation-Maximization Algorithm*, pages 387–387. Springer US, Boston, MA, 2010. ISBN 978-0-387-30164-8. doi: 10.1007/978-0-387-30164-8_291. URL `https://doi.org/10.1007/978-0-387-30164-8_291`.

[88] AM Jehad Sarkar, Young-Koo Lee, and Sungyoung Lee. A smoothed naive bayes-based classifier for activity recognition. *IETE Technical Review*, 27(2):107–119, 2010.

[89] Lalatendu Satpathy. *Smart Housing: Technology to Aid Aging in Place: New Opportunities and Challenges*. PhD thesis, Mississippi State University, 2006.

[90] Emilio Serrano, Juan A Botia, and Jose M Cadenas. Ubik: a multi-agent based simulator for ubiquitous computing applications. *Journal of Physical Agents*, 3(2):39–43, 2009.

[91] Pavle Skocir, Petar Krivic, Matea Tomeljak, Mario Kusek, and Gordan Jezic. Activity detection in smart home environment. *Procedia Computer Science*, 96:672–681, 2016.

[92] Vlado Stankovski and Jernej Trnkoczy. Application of decision trees to smart homes. In Juan Carlos Augusto and Chris D. Nugent, editors, *Designing Smart Homes: The Role of Artificial Intelligence*, pages 132–145, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg.

[93] Karl Weiss, Taghi M Khoshgoftaar, and DingDing Wang. A survey of transfer learning. *Journal of Big Data*, 3(1):9, 2016.

[94] Alper Yilmaz, Omar Javed, and Mubarak Shah. Object tracking: A survey. *Acm computing surveys (CSUR)*, 38(4):13, 2006.

[95] Xiangxin Zhu, Carl Vondrick, Charless C. Fowlkes, and Deva Ramanan. Do we need more training data? *International Journal of Computer Vision*, 119(1):76–92, Mar 2015. ISSN 1573-1405. doi: 10.1007/s11263-015-0812-2. URL `http://dx.doi.org/10.1007/s11263-015-0812-2`.

Publication

Authored publication

- Ali SMM, Augusto JC, Windridge D (2019) Improving the adaptation process for a new smart home user. In: Bramer M, Petridis M (eds) Artificial Intelligence XXXVI, Springer International Publishing, Cham, pp 421-434.
- Ali SMM, Augusto JC, Windridge D (2019) A survey of user-centred approaches for smart home transfer learning and new user home automation adaptation. Applied Artificial Intelligence 33(8):747-774.
- Ali, S.M.M., Augusto, J.C., Windridge, D. et al. A user-guided personalization methodology to facilitate new smart home occupancy. Univ Access Inf Soc (2022). https://doi.org/10.1007/s10209-022-00883-x

Co-authored publication

- J. C. Augusto, M. Quinde, J. G. Giménez Manuel, S. M. M. Ali, C. L. Oguego and C. James-Reynolds, "The SEArch Smart Environments Architecture," 2019 15th International Conference on Intelligent Environments (IE), 2019, pp. 60-63, doi: 10.1109/IE.2019.00010.
- J. Augusto, J. Giménez-Manuel, M. Quinde, Ch. Oguego, M. Ali and J. Reynolds (2020) A Smart Environments Architecture (Search), Applied Artificial Intelligence, (https://doi.org/ 10.1080/08839514.2020.1712778)