A Hybrid Approach to Recognising Activities of Daily Living from Patterns of Object Use.

Isibor Kennedy Ihianle
Architecture, Computing and Engineering (ACE),
University of East London

A thesis submitted for the degree of
Doctor of Philosophy

2018

Supervisors

Dr Usman Naeem
Senior Lecturer in Computer Science
School of Architecture, Computing and Engineering (ACE)
University of East London,
Docklands Campus,
4-6 University Way,
London
E16 2RD

Dr Syed Islam
Senior Lecturer in Computer Science
School of Architecture, Computing and Engineering (ACE)
University of East London,
Docklands Campus,
4-6 University Way,
London
E16 2RD

This thesis is dedicated to
my lovely wife Faith and my children
Nosakhare, Sarah and Elizabeth.

Acknowledgements

Firstly, I would like to thank my supervisors, Dr Usman Naeem, Dr Syed Islam and Dr Abdel-Rahman Tawil for giving me the opportunity to undertake a PhD. I would also like to thanks them for many fruitful discussions and advice over the past few years, and for giving me the encouragement and motivation to get to this point, which seemed so far away so recently! I would also like to thank many of my great colleagues from the School of Architecture, Computing and Engineering, the University of East London for helping to create engaging research environments that were fun to work in. A big thank you goes to Mrs Naomi Larner my line manager and Head of Department Engineering and Motor Vehicle at the Cambridge Regional College for her support and encouragement. Many thanks also go to my former colleagues at the British Institute of Technology and E-Commerce (BITE) where it all began. Thanks to Dr Muhammad Farmer, Mr Abdel Rehman Tanveer, Mr Hamim Kariburyo. Additionally, I would also like to thank Mr Solomon H. Ebenuwa who took his time to read my research proposal and offered good constructive feedback. I cannot forget the numerous discussions we had over the phone and beer. Mr Jacob Ajiere who proofread my work most of the times. A special mention goes to Ms Avinder Bhinder of the Graduate School who was always there to help and to pick my numerous calls for help. Finally, I'd like to specially thank my wife, Faith and my children Nosakhare, Sarah and Elizabeth for their support. I count myself as being incredibly lucky to have such a beautiful family. You all stood by me, encouraged me and provided love and support in spite the difficulties I faced all through the research period. Your patience and understanding kept me going. Without you all, I wouldn't have got this far. I cannot forget my mother for her enduring support, wisdom and the belief that I can do it.

Abstract

Over the years the cost of providing assistance and support to the ever-increasing population of the elderly and the cognitively impaired has become an economic epidemic. Therefore, the emergence of Ambient Assisted Living (AAL) has become imperative, as it encourages independent and autonomous living by providing assistance to the end user by conducting activity and behaviour recognition. Accurate recognition of Activities of Daily Living (ADL) play an important role in providing assistance and support to the elderly and cognitively impaired. Current knowledge-driven and ontology-based techniques model object concepts from assumptions and everyday common knowledge of object used for routine activities. Modelling activities from such information can lead to incorrect recognition of particular routine activities resulting in possible failure to detect abnormal activity trends. In cases, where such prior knowledge are not available, such techniques become virtually unemployable. A significant step in the recognition of activities is the accurate discovery of the object usage for specific routine activities. This thesis presents a hybrid approach for automatic consumption of sensor data and associating object usage to routine activities using Latent Dirichlet Allocation (LDA) topic modelling. This process enables the recognition of simple activities of daily living from object usage and interactions in the home environment. In relation to this, the work in this thesis addresses the problem of discovering object usage as events and contexts describing specific routine activities, especially where they have not been predefined. The main contribution is the development of a hybrid knowledge-driven activity recognition approach which acquires the knowledge of object usage through activity-object use discovery for the accurate specification of activities and object concepts. The evaluation of the proposed approach on the Kasteren and Ordonez datasets show that it yields better results compared to existing techniques.

Contents

List of Figures

List of Tables

Abbreviations

AAL Ambient Assisted Living

AALP Active and Assisted Living Programme

ADL Activities of Daily Living

AI Artificial Intelligence

AmI Ambient Intelligence

ANN Artificial Neural Networks

CHMM Coupled Hidden Markov Models

CONON CONtext ONtology

COSAR Context-aware Activity Recognition

CRF Conditional Random Fields

DAML DARPA Agent Markup Language

DBN Dynamic Bayes Networks

DL description logic

DT Decision Trees

FHMM Factorial Hidden Markov Models

GPS Geographical Positioning System

HMM Hidden Markov Model

HSMM Hidden Semi Markov Model

ICT Information and Communication Technology

ISTAG Information Society Technologies Advisory Group

JESS Java Expert System and Shell

LDA Latent Dirichlet Allocation

NHS National Health Service

NB Naive Baiyes

NN Nearest Neighbour

OIL Ontology Interchange Language

OWL Web Ontology Language

PDA Personal Digital Assistant

PIR Passive Infra-Red

PLSA Probabilistic Latent Semantic Analysis

RDF Resource Description Framework

RFID Radio Frequency Identification

SPARQL SPARQL Protocol and RDF Query Language

SVM Support Vector Machines

SVM-BTA Support Vector Machines with Binary Tree Architecture

SWRL Semantic Web Rule Language

TOQL Temporal Ontology Querying Language

UNFPA United Nations Population Funds

Chapter 1

INTRODUCTION

1.1 Background

The United Nations Population Funds (UNFPA) reports that the world is ageing rapidly [131]. According to this, 12.3% of the global population is made up of people aged 60 years or older, and this number will rise to almost 22% by 2050. The Office for National Statistic (ONS) reports that in the United Kingdom, the fastest growth in the population will be in the older age group and one in twelve people will be over 80 years by 2039 [42]. Advancements in healthcare, improving standards of living and declining mortality have attributed to people living longer. As more people reach 60 years and above, we can expect the number of people below this age group to shrink. While ageing has its advantages, its disadvantages are quite enormous and far reaching. Amongst these are the growing demand on healthcare resources and the provisions of social care. As people age, they tend to become less physically independent, they begin to be frail and in most cases experience declining memory and cognitive impairment. They start to rely on help and possibly losing their independence. They experience difficulties in carrying out daily tasks hence requiring support and care. In addition to the loss of autonomy and independence, they become socially isolated resulting from changes in their living arrangements. With these, they are cut off from friends, other family members, and experience minimal movements to visit or even see places they would have loved to see. Alzheimer and Dementia are diseases commonly associated with ageing. While these diseases have stages and levels of severity, symptoms are the loss of cognitive abilities which affects the person's daily life and activities [8]. In most cases, family members are looked up to provide support and care for their affected elderly relatives. These provisions affect the entire family as financial resources, human resources and even time are invested in helping. Jobs and means of livelihood are given up making the elderly and aged a burden. Older persons and aged also affect the public sector and government because resources are needed to provide social care and support. This may be in the case of the provision of care homes, renumeration of care workers and support providers. According to [42], an National Health Service (NHS) bed costs the taxpayer an average of £1,925 a week in the United Kingdom. For a typical residential home, £558 is needed a week and £357 is the average cost for providing similar care at home.

To augment the provisions of care and support to the elderly seniors, technology driven solutions have been identified to help reduce the burden. It is the belief that through Assisted Living Systems or Ambient Assisted Living (AAL) systems equipped with sensors, computers, wireless networks and software applications for healthcare monitoring, the seniors can live independently in their preferred environment by utilising Information and Communication Technology (ICT) technologies for personal healthcare, support and assistance [2]. As a way of harmonising the core aims and objectives towards providing a unified goal of technology driven support and help for the aged, elderly and cognitively impaired through this emerging area of Ambient Intelligence (AmI), the Active and Assisted Living Programme (AALP) [2] has the following concepts:

- To extend the time people can live in their preferred environment by increasing their autonomy, self-confidence and mobility;
- To support the preservation of health and functional capabilities of the elderly
- To promote a better and healthier lifestyle for individuals at risk;
- To enhance security, prevent social isolation and support the preservation of the multifunctional network around the individual;
- To support carers, families and care organisations;
- To increase the efficiency and productivity of used resources in the ageing societies.

To achieve these, an AAL system could have connected sensors, wireless sensors and actuator networks, computer hardware and networks, software applications, and databases, to provide services in an Ambient Assisted environment [2] as illustrated in Figure 1.1. Typically, the system monitors and captures information from the user and the environment. With analysis and classifications based on this information, a reasoning and inference module can recognise what the user's activities are so as to make assistive decisions to provide support. This process is known as the Activity of Daily Living (ADL) recognition.

ADL recognition also known as Activity recognition is an emerging area of pervasive computing. It is very important due to its significance in the provision of support and assistance to the elderly, disabled and cognitively impaired. Pervasive computing combines technologies with wireless computing, internet capability and artificial intelligence

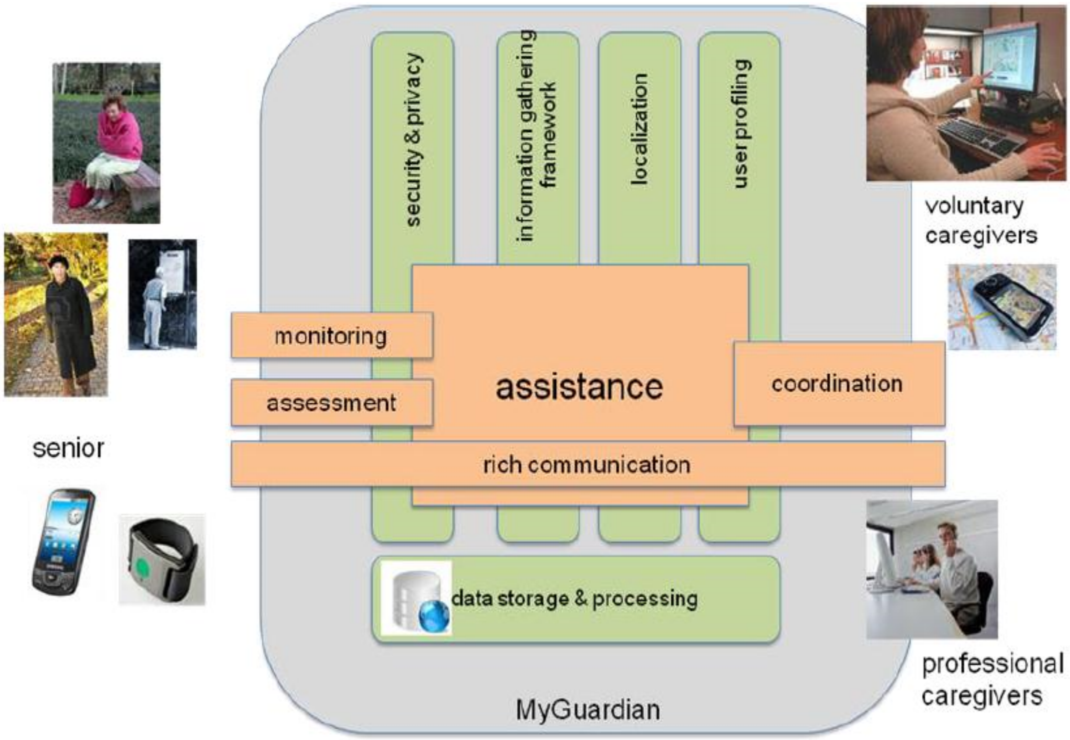


Figure 1.1: An example of an Ambient Intelligence Assisted Environment [1].

to create an environment where the connectivity of devices is embedded in such a way that it minimizes the end user's need to interact with computers. ADL recognition through a pervasive computing process is involved in identifying what an individual is doing e.g. *Sleeping*, *Showering* and *Cooking*.

Accurate recognition of Activities of Daily Living (ADL) plays an important role in providing assistance and support to the elderly and cognitively impaired. Current knowledge-driven and ontology-based techniques model object concepts from assumptions and everyday common knowledge of object use for routine activities. Modelling activities from such information can lead to incorrect recognition of particular routine activities resulting in possible failure to detect abnormal activity trends. As a way forward, it is essential to accommodate object concepts that are specific to the routine activities with regards to the individual and the contextual environment. To provide assistance and support to the elderly and cognitively impaired, the recognition of their ADLs must be accurate and precise with regards to the object use events.

1.2 Motivation

The motivation for this research is to conduct accurate activity recognition which can be used to provide assistance and support for the elderly and the cognitively impaired. Activity recognition can be conducted using data-driven and knowledge-driven techniques. A combination of both these techniques can also be used to carry out activity recognition. Data-driven activity recognition uses machine learning and statistical methods on sensor or vision data, which represents low-level events. This process involves discovering the patterns and the classification of constituents data to make activity inferences. Research efforts in [66, 64, 105, 85] shows the strengths of this approach within the learning process. Although data-driven methods are capable of handling uncertainties, they lack semantic clarity and are either hidden or latent, thus requiring expressivity in an understandable format for the end user. On the other hand, knowledge-driven approaches model activities dependent on the prior knowledge of object usage in the home environment through a knowledge engineering process. The modelling process involves associating low-level sensor data to the relevant activities through knowledge modelling and representation to build a knowledge base. The activity recognition then follows logical inference or subsumption reasoning as the case may be. In comparison with data-driven technique, knowledge-driven techniques are more expressive and semantically rich but weak in handling uncertainties. However, inferences are usually in a format that is easily understood by the end user. Knowledge-driven techniques in most cases depend on the common and everyday knowledge of object usage to model and represent object concepts in association to activity concepts. These common and everyday knowledge of object usage are mostly by assumptions and are generic knowledge of object usage for specific routine activities or sometimes from wiki-know-how [1]. Activity ontologies modelled and represented in this way may not fit into certain situations or capture specific routine activities in home environments. If activity concepts have been developed based on the generic and or assumed knowledge of object usage, the recognition model may fail due to objects fitting which differs with individuals and home environments. This then affects the quality of assistance and support provisions to the elderly and cognitively impaired.

Generic Ontologies models have been designed and developed as in [30, 28] to emphasise reusability and shareability. As a way forward, it is essential to extend these ontologies to accommodate object concepts specific to the routine activities given the individual and the home environment. Individuals differ in their lifestyles, habits and home setting with objects therein also differs. With these, the contexts of activities cannot be generalised

[1] http://www.wikihow.com/Main-Page

with regards to the object usage. For instance *Breakfast* as activity can be performed by an individual using *Kitchen* as a *Location*, *Fridge* and *Microwave*. Another individual in a different home setting could use *Kitchen*, *GroceryCupboard*, *Toaster* and *Cooktop* to perform *Breakfast* as an activity. *Breakfast* is the activity in both cases, but they have been performed using different sets of objects which could be unique to the individual and the home settings. If a recognition model is developed based on the generic knowledge that *Breakfast* has object use *Kitchen*, *Plate* and *Cup*, it would fail to accurately recognise *Breakfast* for both individuals due to wrong object specifications which do not match. The reliance on generic activity ontologies would fail in both cases to recognise *Breakfast* as the activity being performed and eventually affect the assistive and support provisions as the case may be.

To achieve accurate activity recognition, the specifications of the object used for activities can be achieved through an activity context describing the technique. In cases where habits and home settings are changing, it can be used for updates and allow for modifications. The Figure 1.2 illustrates the functional intention of the proposed approach. It extends the generically modelled activity ontology to an activity ontology with the specific activity-context descriptions based on sensor data captured in a home environment. Unlike the generic activity ontology, the activity ontology with specific activity-context descriptions has extended the object contexts such as *Cooktop*, *Toaster*, *GroceryCupboard* from specifications based on the proposed hybrid approach to enhance activity recognition. The idea is to develop an activity recognition technique with consideration to the uniqueness of the user and the home setting. Also as habits, lifestyles and possibly home setting changes, the hybrid approach adapts and allows these changes and modification to be made to the activity ontology through an update service. The eventual hybrid activity recognition technique can be used for accurate activity recognition and integrated into systems for the provision of assistance and support for the elderly and cognitively impaired. In summary, the thesis focuses the development of a novel hybrid activity recognition approach with due consideration to unique and different individuals and home settings. This thesis also with the emphasis on knowledge-driven techniques has identified that the knowledge acquisition process should be extended beyond the generic and everyday knowledge of object usage to build activity ontology. Given these, this thesis is motivated to harness the complementary strengths of the data-driven, and the knowledge-driven techniques to provide activity recognition solutions over the limitations and weaknesses highlighted.

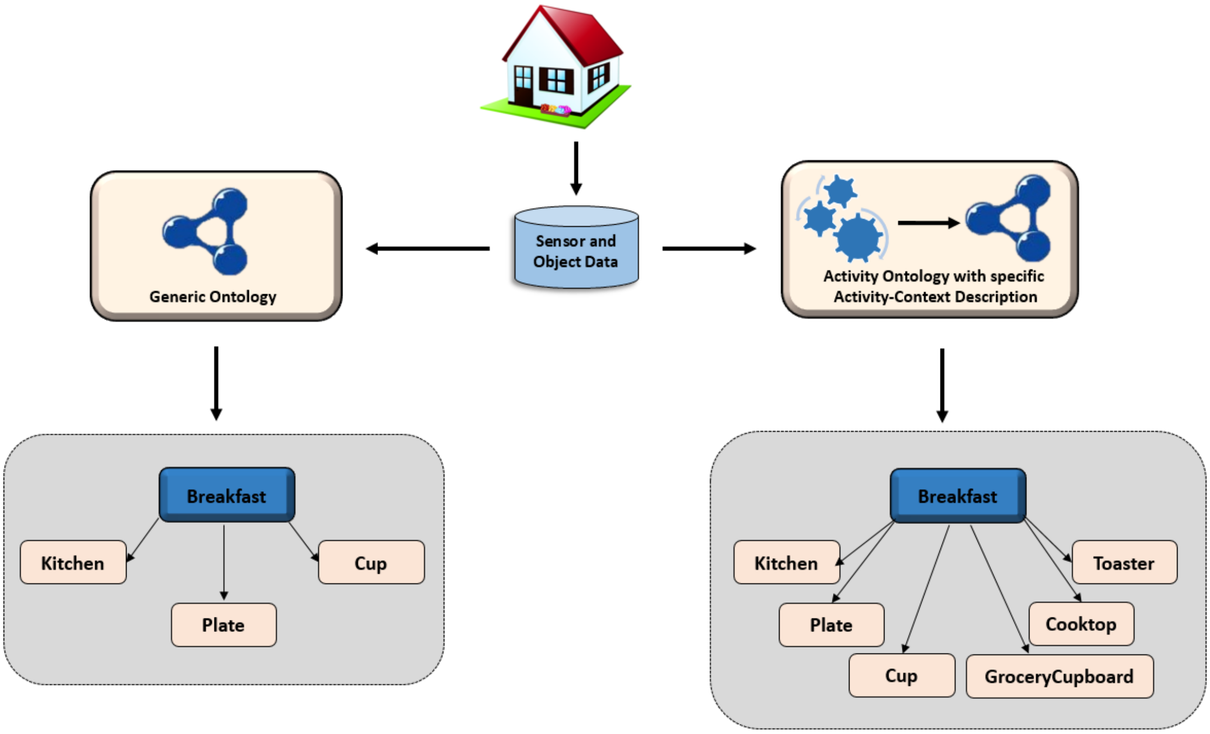


Figure 1.2: Functional Intention of the Approach.

1.3 Thesis Objectives

The recognition of activities relies on accurate representation of activity and object concepts. To achieve this, the knowledge of object contexts used for particular activity situations are essential. With due regards to the *Breakfast* example in the section above, activity ontologies modelled from generic and everyday knowledge of object usage need to be extended to accommodate inferences of specific activity situations from object contexts describing them. In this regard, the principal aim of this thesis is:

To design and implement a hybrid activity recognition approach that recognises routine activity situations as events from sensor datasets by the accurate specification of the object use as the context describing the activities.

This aim can be achieved by addressing the following objectives:

- A review of research efforts in the area of ambient intelligence, context awareness pervasive computing with regards to activity recognition.
- Identification of techniques to discover the object contexts for activities and the

knowledge representation and formalism to enable activity recognition.

- Design and the implementation of context description and activity recognition algorithms on the basis of the specification of the object use as context describing routine activities.
- The selection of an appropriate evaluation methodology to evaluate the proposed hybrid approach against requirements.
- Validate activity recognition results through experiments using sensor datasets representing activity events in home environments.

1.4 Research Methodology

To achieve the goal as stated in the section above, this thesis follows the methodology as illustrated in the Figure1.3 and as given below:

- Review relevant and related literature on ambient intelligence, pervasive computing, assisted living and activity recognition.
- A review of the state of the art in activity recognition with an aim of analysing, evaluating, highlighting the strength and weakness of activity recognition approaches to identify areas for extension and contribution.
- Based on the areas for extension and contribution set out requirements ensuring they become specifications and the basis for measurement.
- Design and develop activity recognition technique and approach based on the areas for extension and contribution identified.
- Carryout incremental tests on the proposed activity recognition technique and approach developed.
- Experimentation and evaluation of the activity recognition approach.
- Selection of an appropriate validation methodology to evaluate the proposed hybrid approach against requirements.
- Presentation of the research results.

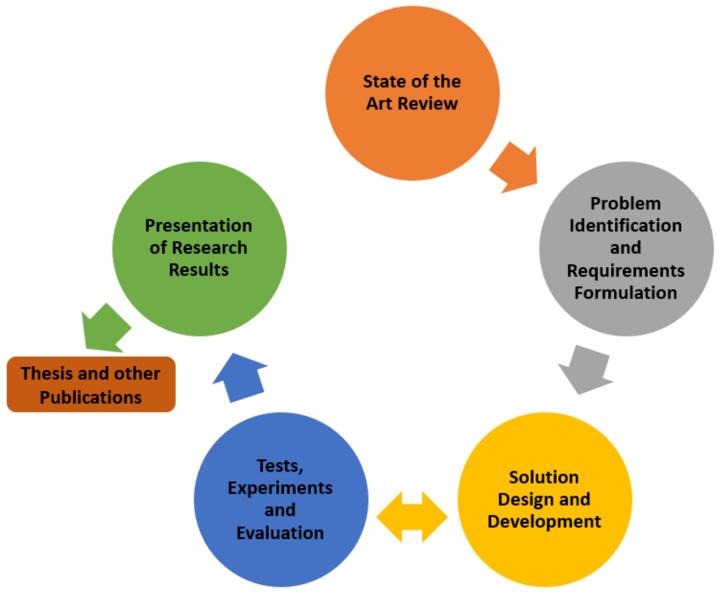


Figure 1.3: Research Methodology

1.5 Thesis Contributions

This thesis presents a novel hybrid approach to sensor and object-based knowledge-driven activity recognition that combines ontology and topic model techniques. The proposed approach offers improvements with regards to the limitations and drawbacks of each of the separate techniques to activity recognition, such as accurate specification of object concepts as context descriptors for particular activities and clear expressivity of recognised activities. With these, this thesis makes the following contributions:

- A review of research efforts in the area of ambient intelligence, context awareness pervasive computing with regards to activity recognition. This thesis also considered a broad overview of these attempts highlighting data, knowledge driven and hybrid approaches. Distinctions are made of the features, methods and the identified emerging approaches towards activity recognition. Limitations thereof are identified and possibly how the approaches can be complementary. The details of the review can be found in in Chapter 2.

- A novel hybrid approach made up of a context description module that augments an activity ontology module as components. A context description algorithm discovers and assigns the specific objects as context descriptors for particular routine activities which are modelled as activity ontology concepts.This context description algorithm algorithm is presented in Chapter 4.

- A methodology to model static and dynamic activity situations by combining ontology concepts formalism and 4D fluent (temporal attributes formalism with ontology). This is especially applicable to activities occurring at specific times of the day and having same or similar object interactions. This methodology to model static and dynamic activity situations is presented in Chapter 5.

- A methodology to model fine grain activity situations with precedence property with the realisation that activities are a result of atomic events occurring in patterns and orders. These patterns and orders differ, and in some cases, the patterns determine the evolving activity situations. This methodology to model fine grain activity situations is presented in Chapter 5.

- A methodology to detect activity boundaries to signal the end of an activity and the beginning of another activity by the introduction of location concept for objects within the same location of a home environment was introduced. This is based on the assumption that if two consecutively observed objects in use belonged to the same location in the home environment, it suggests the continuation and persistence of an activity. The details of the methodology can be found in in Chapter 5.

- An activity ontology update algorithm to update the activity ontology without the process of editing the entire ontology. Ontologies have been known to be static, so modifications and changes with the activities and object usage are achieved through an ontology update process. This activity ontology update algorithm is presented in Chapter 5.

1.6 List of Publications

The work presented in this thesis has been partially published in the following papers.

- Isibor Kennedy Ihianle, Usman Naeem, Syed Islam and Abdel-Rahman Tawil "A Hybrid approach to Recognising Activities of Daily". MDPI Informatics Journal - Special Issue : Sensor-Based Activity Recognition and Interaction, 2018.

- Isibor Kennedy Ihianle, Usman Naeem, Syed Islam and Abdel-Rahman Tawil "Recognising Activities of Daily Living from Patterns of Object Use". International Journal of Hybrid intelligent Systems (IJHIS), 14(3): 2018

- Isibor Kennedy Ihianle, Syed Islam, Usman Naeem, Saeed Sharif, Muhammad Awais Azam and Amin Karami "Intelligent Recognition of Activities of Daily Living from Patterns of Object Use" in Intelligent Systems Conference (IntelliSys) 2018, [Accepted].

- Isibor Kennedy Ihianle, Usman Naeem, and Syed Islam "Ontology-Driven Activity Recognition from Patterns of Object Use". in Proceedings of the 2nd International Workshop on Intelligent and Personal Support of Human Behaviour in conjunction with UbiComp '17, 2017.

- Isibor Kennedy Ihianle, Usman Naeem and Syed Islam "Knowledge Driven Activity Recognition from Patterns of Object Use" 5th Activity Monitoring by Multiple Distributed Sensing Workshop AMMDS 2017 [Accepted]

- Isibor Kennedy Ihianle, Usman Naeem and Abdel-Rahman Tawil, "Recognizing Activities of Daily Living from Patterns and Web Knowledge Extraction", 1st International Workshop on Intelligent Personal Guidance of Human Behaviour Utilizing Anticipatory Models in conjunction with UbiComp '16, 2016.

- Isibor Kennedy Ihianle, Usman Naeem and Abdel-Rahman Tawil, "Recognition of Activities of Daily Living from Topic Model," The 7th International Conference on Emerging Ubiquitous Systems and Pervasive Networks (EUSPN 2016), 2016.

- Isibor Kennedy Ihianle, Usman Naeem and Abdel-Rahman Tawil, "Getting Knowledge from Patterns for Activity Recognition," in Artificial Life and Intelligent Agents Symposium, ALIA 2016 [Poster].

- Isibor Kennedy Ihianle, Usman Naeem and Abdel-Rahman Tawil, "A Dynamic Segmentation Based Activity Discovery through Topic Modelling," in IET International Conference on Technologies for Active and Assisted Living (TechAAL 2015), London, 2015.

1.7 Structure of thesis

This thesis structured into seven chapters and are as follows:

Chapter 2 presents the state of the art techniques on the recognition of Activities of Daily Living (ADL) in the home environment. It reviews related work and existing approaches in activity recognition. It discusses state of the art in the area of ambient intelligence, ubiquitous computing and context awareness. In addition, data and knowledge driven techniques employed in activity recognition have been reviewed.

Chapter 3 focuses on knowledge representation and formalism for activity recognition. In this chapter, the semantic and ontology were discussed as the approach employed in this thesis to represent knowledge for activity recognition. It also explains in detail the process of representing ontological concepts, modelling and inference.

Chapter 4 introduces the proposed approach, methodology and the conceptual architecture of the approach and the comprising components. It also presents knowledge acquisition and context description for activity ontology. This chapter focuses on how knowledge of object usage is acquired for the proposed activity ontology through activity-object discovery. This chapter describes the process of context description for activity situations with an emphasis on the use of Latent Dirichlet Allocation LDA to discover activity-object use. Further, it presents context description from discovered activity-object use distributions for activity ontology.

Chapter 5 presents ontology modelling of activities of daily living. It also presents the modelling and representation of activity and object concepts in the ontology. It describes the implementation and construction static and dynamic activities and their context descriptors in the ontology as a prototype of the approach.

Chapter 6 presents the dataset, experiments and evaluation results of the technique proposed in this thesis.

Chapter 7 concludes the thesis, final remarks, reflections and future work.

Chapter 2

RECOGNITION OF ACTIVITIES WITHIN THE HOME ENVIRONMENT

The recognition of the ADLs of the elderly within an home environment is a major research area due to the ageing population worldwide. Typical recognition systems monitor the ADL of the elderly cognitively impaired to provide them support and assistance. In this process, object usage data is captured using vision and sensor devices wearable sensors. The captured data is classified using machine learning techniques and in some cases modelled into knowledge-driven methods to infer activities. The machine learning techniques work by discovering activities associated with the most likely value with regards to a set of independently observed object data. On the other hand, the knowledge-driven method involves associating low-level object data to the relevant activity through knowledge modelling and representation to build a knowledge base. There are also hybrid models that combine data and knowledge driven techniques. This chapter provides a review of the state of the art techniques in activity recognition. It also discusses the previous related works on activity monitoring, activity discovery and pattern analysis and modelling. While these techniques of activity recognition have made useful and significant advances, they have some weaknesses, limitations and challenges which are presented in the concluding part of this chapter.

2.1 Ambient Intelligence

Ambient Intelligence (AmI) was introduced in 2001 by the European Commission's Information Society Technologies Advisory Group (ISTAG) [51] in a bid to develop systems to help people within their own environments. Over the years, researchers have made contributions in this area of endeavour making it evolve into an emerging field of computing. The authors of [9], defines it as a technological system which proactively provides support to people in their environment by the use of contextual information. A set of technologies that should disappear into the user's environment to make life easy and entertaining [34]. Artificial Intelligence (AI) supports this as the use of sensor technologies [91] and designs which enhance user-friendliness, human assistance and easy interaction. The design of an Ambient Intelligence (AmI) is such that it is user centric to provide assistive services based on its ability to learn and predict the user actions and the environment. Besides, Cook et al. [91] summarised technologies provided by AmI as sensitive, responsive, adaptive, transparent, ubiquitous, and intelligent.

So it builds upon sensors and sensor networks, pervasive computing, and artificial intelligence to have pervasive and ubiquitous computing and context-awareness as contributing technologies as illustrated in Figure 2.1. Hence, AmI systems should have:

- Sensing mechanism to capture information from the user and the environment.
- Reasoning module able to recognise action of the user and make the decision to provide support.
- Actuators that execute actions and affect the system users.

2.1.1 Pervasive Computing

Pervasive computing also known as ubiquitous computing aims to create AmI where technologies are embedded and integrated into everyday objects in the user's environment to be non-intrusive and communicate unobtrusively. In 1988, Mark Weiser coined the term "Ubiquitous Computing" setting the principles of computers to be quite, invisible, calm and by intuition smarter [138]. Weiser [137] described it as the use of computers in the physical environment to be non-intrusively available to enhance everyday life of the home user without being visible. In pervasive computing, devices are interconnected to collect autonomously, process and transport information to improve the human experience unaware of the underlying communications and technologies. Computational techniques through this would be human centred and freely available everywhere and disappear into the environment. To create ubiquity, a typical pervasive environment as illustrated in Figure 2.2 (An example of context attributes encoded to activity ontology in CONtext ONtology (CONON) [136]) would be equipped with monitoring devices such as sensors, cameras, microphones on home items or the user to gather information about their use even without

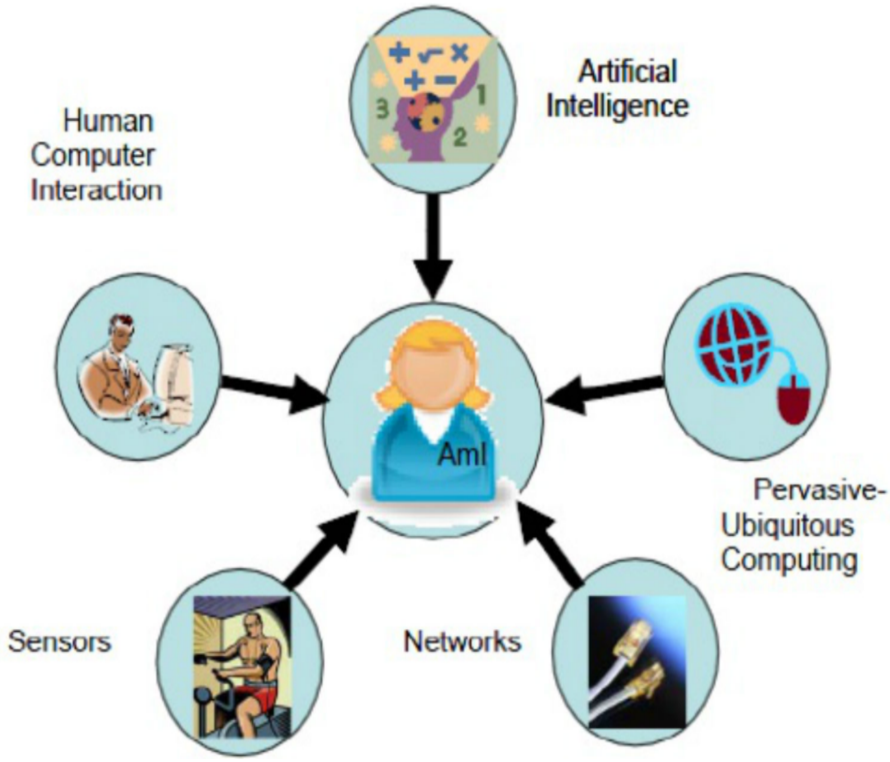


Figure 2.1: Ambient Intelligence building block [91]

the user's intervention [91]. These devices are networked across different layers seamlessly and operated in some cases through applications which support input/output modes [119, 89]. A prominent feature of a pervasive environment is intelligence and its ability to make decisions based on its state and the user with regards to the home object used.

2.1.2 Context Awareness

Context-awareness as a contributing technology to AmI uses contextual information to make human-computer interactions easier without an interfacing medium for engagement. Dey and Abowd [36] defined context as "any information that can be used to characterise the situation of entities (i.e. whether a person, place or object) that are considered relevant to the interaction between a user and an application, including the user and the application themselves. Contexts are typically the location, identity and state of people, groups and computational and physical objects." Which is then define Context-awareness as the ability of devices to sense, detect interpret and respond to aspects of a user's environment and the devices themselves [103, 63]. Salber et al. [120] defined context-awareness as the ability to provide real-time sensing of contexts to provide flexible computational service. Typical characteristics of a context-aware technologies includes: context-sensitive [71],

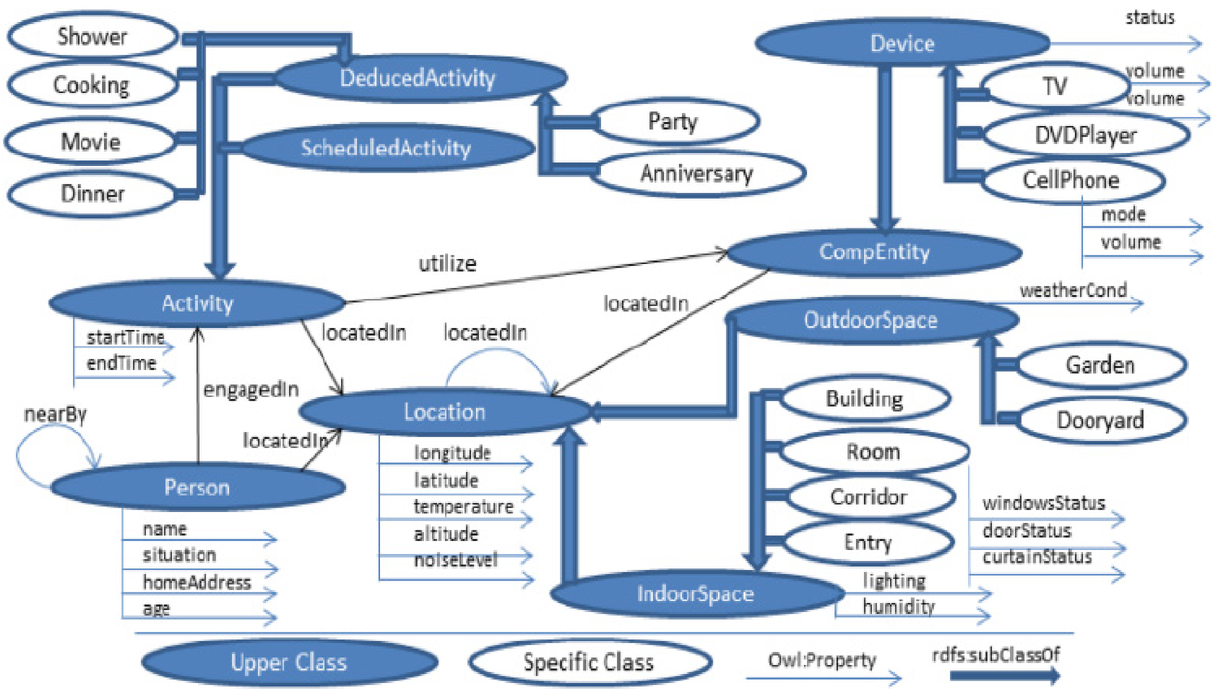


Figure 2.2: An example of context attributes encoded to activity ontology in CONtext ONtology (CONON)[136]

responsive [39], reactive [33] and adaptive [22]. While the features of a typical context-aware application include the above, Dey et al. [37] limits its development to the human-computer interface. Context-aware technologies must also add context which Ryan et al. [117] defined as the user's location, environment, identity and time. Dey and Abowd [36] suggested contexts should include the computing environment, user environment and the physical environment. Important aspects of contexts should include who's, where's, when's and what's (that is, what the user is doing) of entities and these could be used to determine why an activity is taking place [123]. Dey et al. [37] suggested context-aware technologies could use the who's (the user's identity), where's (the user's location) when's (the time of activity), and what's (the user's activity) of entities to determine why the situation is occurring. All these contextual elements add to identify activities or situations involving the user. As a result, context-aware applications can add to provide users with context-aware services. Besides, Lui et al. [87] added that context-aware technologies should have context acquisition and sensing methods, context modelling and representation, context filtering and fusion, context storage and retrieval in addition to the context applications.

With regards to these settings, conventional activity recognition technologies could use accelerometers to measure relative motion, location sensing devices, audio sensing devices, time amongst many. The knowledge of these context attributes as illustrated in Figure 2.2

could help to provide assistive support to the user. The goal is to enable computers through AmI and context-aware technologies to have similar capabilities as humans for recognising people's activities. For example, if a person had dementia, the context-aware application could provide alerts if the person forgets to their take his or her medications. Activity completion support could also be given to the home user. This could be done through activity recognition dependent on contexts like location, time and previous events. Finally, activity recognition dependent on context-aware technologies could reliably recognise user's various activities to improve the way they interact with computers, and through this make a huge impact on behaviour, social, and cognitive sciences.

2.2 Ambient Assisted Living

Ambient Assisted Living (AAL) relies on AmI technologies for its deployment. Typical components of an AAL system includes a monitoring system (such as vision and sensor devices) and a recognition and behaviour analysis model [31]. In the provision of care for the elderly, the AAL system uses vision capturing devices and a variety of sensors to monitor the daily activities of the elderly cognitively impaired. The deployment of these devices in the user's environment are used to pervasively augment [137] as well as capture contexts such as location, time, etc. [37]. Issues such as privacy and acceptability of the monitoring systems are worth considering in the choice of these devices. The analysis of the captured data may focus on recognising the activities of individual user in comparison with daily schedules. It could also be used to recognise abnormal activity trends to identify a decline in health or even dangerous activities. Another could be to provide assistance or support completion of activities of daily living. Irrespective of its use, the deployment of the recognition and analysis components are supported by middleware technologies and software agents. According to Ruijiao et al. [115], AAL services are applied in daily task assistance, rehabilitation and social inclusion, mobility assistance and healthcare as illustrated in Table 2.1.

In addition, prominent projects offering assisted living support and assistance includes; The *CASAS* Smart Home [109, 125] at the Washington State University. This smart home was created to provide comfort, safety, improve productivity and autonomous living of its residents. It has deployed sensors to monitor activities and environmental factors, intelligent agents and a set of actuators to provide mechanisms for control or movement in the provision of assistance and support as the case may be. *iDorm* [91] was established by the Intelligent Environment Group of Digital Lifestyle Centre, University of Essex. This project was upgraded to iSpace composed of physical static computational components

Area	Service Provided
Daily task assistance	Support with daily tasks such cleaning, cooking. Life style management such as when to wake up or got to bed, watching TV and listening to radio. Activity and task reminders.
Rehabilitation and Social Inclusion	Personal Digital Assistant (PDA) enhanced application to read emails, operate washing machine, food items manager, enhanced user dialogue system, access to social network, newspaper reader [45], e-inclusion to stay connected with friends and family [83].
Mobility Assistance	Fall detection, enhanced mobility through smart wheel chairs, Smart route planning and navigation [124], Assistive and robotic limb [47], Obstacle detection for outdoor movements [142], Robotic wheel chair with assistive limbs [32].
Healthcare	Medication and drugs management, Care support and reporting systems [75], Rehabilitation system, Disease management systems [100].

Table 2.1: Services provided by the different areas in Assisted Living.

to learn users' behaviour, a robotic agent and portable computational devices for wireless interactions with the smart home. The *Gator Tech Smart Home* was developed by Helal et al [55] to provide support and assistance to the elderly. This project through deployed technologies which notifies residents when to do laundry, monitor sleep pattern through the smart bed and a cognitive assistant to guide occupants through tasks. The Ambient Lighting Assistance for and Aging Population *ALADIN* [90] project seeks to develop adaptive lighting system which can respond to users lighting needs with respect to the situation and also provide eco-energy management. The *PlaceLab* project is a living laboratory which imitates a real home for routine and everyday activities. Equipped with sensors and vision and audio monitoring devices, it provides context sensitive reminders based on activities recognised [69]. Other projects completed and ongoing includes *iCare* [67], *SYSIASS* [6], *PERSONA* [95], *CareLab* [35], *Aware Home* [76], *RoboCare* [25], and *LsW* [38] amongst many which are not just laboratory based but technologies focusing on routine and everyday activities. To fully understand how these technologies have been deployed, the next section considers approaches to activity recognition.

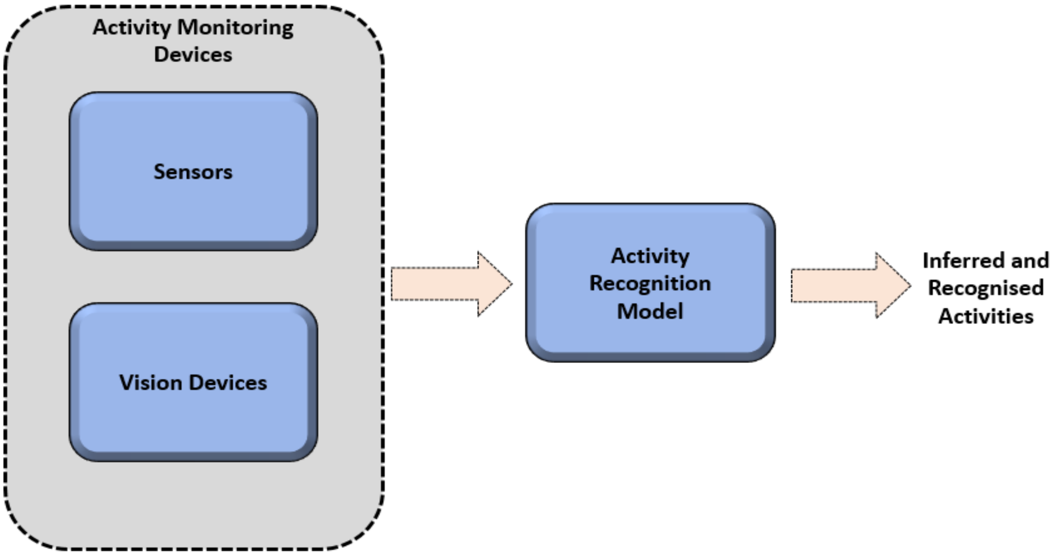


Figure 2.3: A generic activity recognition model

2.3 Activity Recognition

Activity recognition is the process of recognising the actions of an individual through the use one or more agents. According to Chen et al. [27], activity recognition process involves inferring ongoing activities in an environment through the use of technology driven agents which monitor and analyse the activities as they occur. A generic activity model as illustrated in Figure 2.3 tracks user's behaviour and contexts such a location, time, etc. through vision devices, a set of multi-modal sensors or both. This thesis refer to this process of monitoring as activity monitoring process. The activity recognition model processes the incoming monitored data for inference. Activity recognition models can be broadly classified into three; data dependent, knowledge driven and the hybrid models. Data dependent also known as machine learning models, statistically analyse data captured using activity monitoring devices and process them to infer ongoing activities [66, 84]. The knowledge driven methods build activity models by encoding characteristic behaviour of the home user and contexts [21], and the hybrid models combine of data and the knowledge-driven models. The output of the activity monitoring and recognition models are the inferred or recognised activities. In the sections below provide a review of activity monitoring process and recognition models.

2.3.1 Activity Monitoring

Activity recognition models depend on contextual data from the user and the environment. These are captured through activity monitoring devices. Currently, there are two categories of activity monitoring devices include vision and sensor devices.

2.3.1.1 Vision Monitoring Devices

Vision-based activity monitoring use cameras to capture users' activities and environmental changes in the form of digitalized visual data or video sequences. In most cases, a set of multiple cameras is involved to capture context-rich data which may include sound and various image ranges. Captured images are taken through feature extraction, structural modelling, segmentation and pattern recognition processes to recognise the activities which they represent. A major limitation to vision-based activity recognition is that it has been considered to be invasive. Besides, they suffer scalability and reusability issues due to the complexity of real-world setting [27, 28]. Main areas of the application include security, surveillance and capturing biometric data such as fingerprints. Vision-based activity recognition has also been used in smart rooms [27].

2.3.1.2 Sensor Monitoring Devices

A wide range of sensor-based monitoring devices such as accelerometers, Radio Frequency Identification (RFID), motion detectors, Geographical Positioning System (GPS), etc. are used to capture users' behaviour and environmental contexts. Given their variety, use and technical specifications they are deployed where they are needed most. Chen et al. [27], classified sensors into two categories according to how they are deployed: Wearable sensors (Sensors worn by the user) and Object based sensors (Dense sensing or sensors tagged to objects).

Wearable sensors are sensors carried by the users or attached to the body of the user. They can be strapped, clipped to the clothing of the user or even embedded into mobile devices belts, shoe eye glasses, etc. in the process of capturing data. Typical wearable sensors include inertial sensors (accelerometers, gyroscope and Geographical Positioning System GPS Sensors) and body monitors (heart monitors, thermometers, electro-cardiographs, oximetry sensors, etc.). Accelerometers and gyroscope are the most commonly used wearable sensors due to their cost, efficiency and purpose. They are used to measure and monitor motion-based activities such as walking, climbing, running, jogging, falls, etc. Bourke and Lyons [20] used accelerometers to distinguish activities like standing up and lying down, sitting and standing up. Huynh et al. [64] also used accelerometers to discover walking, sitting and eating activities. The use of gyroscopes helps to achieve greater positional results as in Bourke and Lyons [20]. GPS sensors are also another widely used set of wearable sensors. They are used to capture data for location based activities. Patterson et al. [105] used GPS sensors to detect users' behaviour of boarding a bus at a particular bus stop and disembarking. Liao et al. [85] also used GPS in their work to infer user's mode of travel (bus) and taking a wrong bus. Body monitors also known as biosensor are deployed in healthcare to help provide a monitoring system for vital body signs such respiratory, heart, blood pressure, temperature readings, etc. Research works on the use of biosensors include [54] and [41]. In spite of their use and advantages, wearable sensors have some limitations. Users may be reluctant to have wearable sensors worn, tagged, strapped or clipped on them. This acceptability issues raises the question of their use as it may not reflect their acceptance in real life scenarios. Other issues as reported by Chen et al. [27] includes their ease of use, battery life and their size.

Dense and object based sensors consist of passive motion detectors to record human activity. To avoid privacy concerns and reduce cost, dense sensing networks are composed of simple low cost sensors (passive infrared motion detectors) to monitor public spaces and sometimes tagged to objects. These sensors can only detect the presence of a person and cannot identify the person, thus alleviating privacy concerns. Object-based sensors are attached to objects to capture user-object interactions. According to Chen et al. [27], the objects manipulated can determine the activity that is ongoing. We can identify the activities from the sensor data reported because the sensor data is a representation of the objects used that are associated with specific activities. Commonly used sensors are Radio Frequency Identification RFID sensors and binary sensors. Binary sensors are state-change sensors which provide binary reports of ON and OFF, OPEN and CLOSE or 0 and 1 to represent the state of the associated object usage. They are very cheap, easy to install and small. Kasteren et al. [132] used binary state-change sensor tagged to the toilet, doors, fridge, cupboard, etc. to recognise activities like sleeping, toileting, leaving, etc. In the same manner, Ordonez et al. [99] used binary sensors attached to home objects like cooktop and microwave in their work. A major limitation of the use of sensor-based dense sensing is their inability to quantify measurements, hence restricting their use to activities involving the determination of object use. Noisy data could also be associated with their use requiring enormous computational efforts in data pre-processing and cleaning. Irrespective of their drawbacks and limitations, it's hard to say which of visual monitoring devices, wearable sensors and object based sensors is best. The targeted activity and situation determines the choice of the monitoring device to use. According to Chen et al. [27], none of the monitoring devices is mutually exclusive. They could all be used together to achieve better recognition performance.

2.4 Activity Recognition Approaches

Activity recognition approaches can be classified into two broad categories data, knowledge driven approaches and hybrid approaches for activity recognition. These classifications are based on the methodologies adopted, how activities are modelled and represented in the recognition process.

2.4.1 Data Driven

As the name implies, data driven approaches are data dependent. They use of machine learning techniques on datasets to discover and recognise activity models contained therein. The process of activity recognition involves the use of probabilistic and statistical algorithms on existing dataset to perform activity inference. Data-driven approaches can be generative or discriminative.

According to Chen et al. [27], the generative approach builds a complete description of the input (data) space, usually using a probabilistic model. The resulting model induces a classification boundary which can be applied to classify observations during inference. The classification boundary is implicit, and a lot of activity data is required to produce it. Generative classification models includes Dynamic Bayes Networks (DBN) [129, 70], Hidden Markov Model (HMM) [70, 104, 105, 94], Naive Baiyes (NB) [129, 130, 80], LDA [65, 113, 73]. Bayesian networks are graphical models structured as a directed acyclic graph. They are designed for establishing relations between variables. Nodes in the graph represent a random variable with the probability of the corresponding variable. The directed arcs between nodes indicate their dependencies, such that one variable affects the other one directly and this effect can be defined by a conditional probability [116]. Tapia et al. [130] used NB to recognise activities from wearable sensors. Activities recognised includes toileting, bathing, preparing lunch, etc. Also in [129], the authors used both decision trees and NB to recognise gymnasium based activities like walking, running, push ups, sit ups, etc. Bayesian Networks are unable to model temporal entities. DBN were proposed to overcome the temporal limitations of the Bayesian Networks. HMM are a type of DBN with one discrete hidden node and one discrete or continuous observed node per slice. HMM have the advantage of representing spatiotemporal information. Activity recognition approaches using HMM represents an activity as a sequence of hidden states.

A user is assumed to be in one of the states at each time, and each state emits an observation. The user makes a transition to another state following the transition probabilities between states. Once transition and emission probabilities are learned from labelled data, activities are recognised by finding the most likely state sequence in the model that produced the observations [3]. Patterson et al. [104] used interaction information derived from object usage-based sensing to infer ongoing activities using both HMM and DBN. Activities recognised in this process includes cooking and what the home user was cooking. Modayil et al. [70] recognised interleaved activities by the use of the HMM approach. The approach models activities in the context of the last object used. A major drawback associated with HMM their inability to recognise all observation sequences with specific activities. Several HMM extensions have been proposed as the way forward to solving these limitations and other drawbacks not mentioned in this thesis. With the Hidden Semi Markov Model (HSMM), explicit state duration probability distribution is used instead of self-transition probabilities. So, states have variable durations, and some observations are produced in each state according to the duration determined by the distribution [59]. Also, the Factorial Hidden Markov Models (FHMM) enable multiple dynamic processes to interact to produce a single output [78]. The Coupled Hidden Markov Models (CHMM) models dynamic relations between several events by considering as a set of HMM where states at specific times are conditioned by the states at the time for all instances of HMM [101].

Topic models inspired by the text and natural language processing community. The authors of [113] were able to apply the LDA to a collection of documents to show that each author is associated with a multinomial distribution over topics and each topic is associated with a multinomial distribution over words. The LDA topic model has also have been applied to discover and recognise human activity routines in research works by Katayoun and Gatica-Perez [73] and Huynh et al. [65]. Huynh et al. [65] applied the bag of words model of the LDA to discover activities like dinner, commuting, office work, etc. The process involved activity discovery of partitioned sensor segments of time windows. While Katayoun and Gatica-Perez [73] discovered activity routines from mobile phone data, Huynh et al. [65] used wearable sensors attached to the body parts of the user. Although activities discovered (topics) were mostly latent and hidden as the word "latent" in LDA implies, it has the compelling feature of been able to assign words and documents to the discovered topics. The approach proposed significantly with LDA approaches by [73] and [65] with the inclusion of a activity ontology to make object and activity concepts more expressive for the end user.

On the other hand, the discriminative approach, also called conditional models works by modelling the dependence of an unobserved variable on an observed variable such as sensor data to produce activity inference as outputs. Discriminative models, as opposed to generative models, do not allow generating samples from the joint distribution of the models [27]. Discriminative Clasification tries to model by just depending on the observed data. It makes fewer assumptions on the distributions as opposed to generative models. Discriminative models also do not make classification from the joint distribution of observed and target variables. Discriminative models includes Nearest Neighbour (NN)[82, 64] , Decision Trees (DT) [13], Support Vector Machines (SVM) [64, 24, 107], Conditional Random Fields (CRF) [86], multiple eigenspaces [66], and k-means [64]. Artificial Neural Networks (ANN) are modelled to imitate information processing of a biological neural system whose components are composed of neurones and links. In the artificial intelligence systems, each neurone is responsible for an arithmetic operation the output of which will be served as input to the successor neurones through links [116]. A perceptron represents a basic system which consists of some input neurones connected to an output node. Yang et al. [143] proposed an approach using neural classifiers based on signals received from a triaxial accelerometer. The pre-classifier discriminates static activities from dynamic ones by using body acceleration feature. With the distinctions is made, classifiers for static/dynamic activities (standing, sitting, walking, etc.) are constructed using a particular set originated from the acceleration data. SVM functions to locate a hyperplane separating classes from each other with a maximum margin that is the distance between two data points in each class where their distance from the hyperplane is minimum. The closest points to the hyperplane are called support vectors (SV). Qian et al. [107]. used SVM decision trees to recognise activities in a surveillance system. In this approach, differences between activities are learned by identifying boundaries between activity classes in a hierarchically using decision tree where an SVM binary classifier represents each node. By integrating all SVM in the nodes, a multi-class SVM is generated Support Vector Machines with Binary Tree Architecture (SVM-BTA). Cao et al. [24] used SVM to recognise activities from a video system. The captured video data were represented by a set of filtered images which were fed into a classification module. While the major limitation with SVM is the inability to model temporal interactions, ANN are criticised for being a “black-box”, i.e. relations between inputs and outputs are hidden within the network structure, which makes the interpretation of the calculated results difficult. Conditional Random Fields CRF are graphical models which represent conditional probability of a sequence of hidden variables, e.g. activity labels, given a sequence of observations. CRF considers only labels in conditional probabilities, instead of joint probabilities of labels and observations. The authors of [86] used CRF on GPS data to recognise activities. This discriminative approach was formulated as a hierarchical structure of the GPS data capable of handling temporal data. Huynh and Schiele [66] applied eigenspaces to recognise activities from wearable sensors. Theoretically, if a scalar λ is defined as an eigenvalue of the n x n matrix A so that $Ax = \lambda x$. x is called an eigenvector corresponding to the eigenvalue λ. The eigenspace of the n x n matrix A corresponding to the eigenvalue λ of A is the set of all eigenvectors of A corresponding to λ. In their work, they considered data from a linear subspace referred to as eigenspace, in the process the eigenvectors were found. The optimal eigenspace is further determined to represent the data without using any prior annotation or user intervention. To model complex structure inherent in data sets, they extended this to multiple eigenspaces, which are then used to classify activities. With the use of CRF, it is possible to model conditional probabilities without specifying the probability distribution of the observations, which is the most daunting phase. The major weakness of this approach is the complexity associated with the training process which is computationally expensive with many features involved.

2.4.2 Knowledge Driven

The knowledge driven approach builds activity models by exploiting rich prior knowledge in the domain of interest [27]. Rich domain knowledge is used instead of the learning process in data-driven approaches. Knowledge-driven approaches are semantically clear, logically elegant and easy to get started and do not require to be trained. Various methods, in particular, knowledge engineering methodologies and techniques, are used to model domain knowledge. This domain knowledge can then be encoded in various reusable knowledge structures, including activity models for holding heuristics and prior knowledge of performing activities. Domain knowledge can also be encoded contexts for maintaining relationships between the activities, objects, temporal and spatial contexts [4, 26, 43, 48, 57]. With regards to the knowledge model and structures, knowledge-driven techniques can are grouped into three categories logic, mining and ontology technologies.

The logic-based approach uses the specification knowledge representation to represent knowledge models. Knowledge engineering techniques are used to set up and acquire domain knowledge. Logic-based Knowledge representations of activities and sensor data use logical reasoning for activity inference. The process of recognition would involve using observed sensor data against the knowledge-driven model to recognise an activity by way of logical induction, abduction, and or deduction. Given these, domain knowledge can be easily added to allow data fusion. The logic based approach was used by the authors of Kautz et al. [74]. They built activity plans from first-order axioms. Bouchard et al. [19] used lattice theory and action description logic (DL) to identify activities. This work captures the subsumption relationships among activities and activity structures that are modelled as action sequences. With these structures, transitions from initial states to final states in the performance of activities are presented. The classification performs activity recognition through a lattice structure. Lymberopoulos et al. [88] investigated the use of spatiotemporal reasoning in activity recognition. Data is gathered from a sensor network that monitors a person over space and time and generates spatiotemporal sequences of characteristic information. Chen et al. [29] proposed event calculus modelled from events and object states which they used as properties. In this process, they were able to use logical constructs to model compound and parallel activities. An activity trace is simply a sequence of activities that happen at different time points. Activity recognition is mapped to deductive reasoning tasks, e.g., temporal projection or explanation, and activity assistance or hazard prevention is mapped to abductive reasoning tasks. Although, these allowed context-rich concepts, logic-based methods were not flexible enough to be adapted for other users. Similar to other knowledge-driven methods they lacked the ability to handle uncertainties.

The mining-based approach creates activity model by mining existing activity knowledge from a publicly available source. The activities are identified first and described from relevant sources. Information retrieval techniques are then used to determine the definitions of the activities from specific sources and extract examples of phrases or statement which describe the object usage. Afterwards, an algorithm is then used to determine to estimate the object-usage association. Wyatt et al. [140] mined the web to create activity model of object use. Applying discriminative approach they built a genre classifier which when exploited the Viterbi Algorithm and Maximum Likelihood to learn customised activity parameters from unsegmented, unlabelled sensor data. Palmes et al. [102] proposed a method for activity segmentation and recognition. They applied the frequency of object usage for different activities from web pages. This method created object weights which were used to recognise relevant activities and create a segmentation of activity trace. The major limitation with the mining based approach is the lack of reusability of the genre classifier built from this model.

Ontology-based activity recognition is an emerging area in knowledge-driven approach. It involves the use of ontology modelling and representation to support activity recognition, support and assistance. Ontology uses the formal and explicit specification of a shared conceptualization of a problem domain [52]. Vocabulary for modelling a domain are provided by specifying the activity and object concepts, properties, and their relationships. It then uses domain and prior knowledge to pre-define activity concept in the ontology [27]. Latfi et al. [81] proposed an ontology framework for a telehealth smart home aimed at providing support for elderly persons suffering from loss of cognitive autonomy. Chen et al. [30, 28] proposed an ontology-based approach to activity recognition in which they represented activity and object concepts and contexts for explicit domain modelling. Sensor

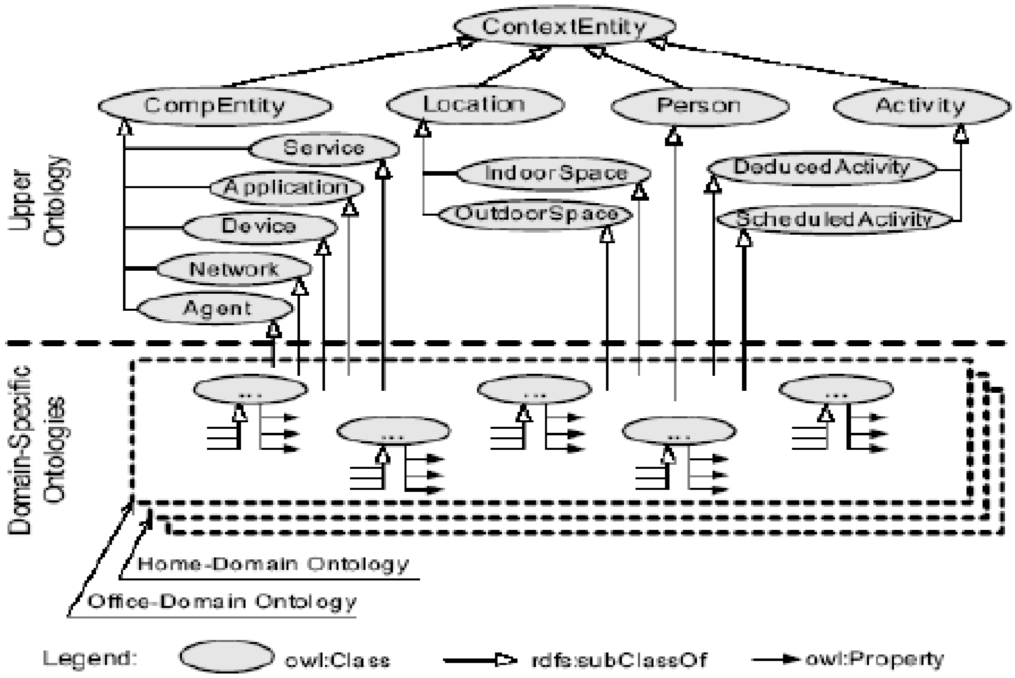


Figure 2.4: An activity ontology with specification of conceptual structures [136]

activations over a period are mapped to individual contextual information and then fused to build a context at any particular time point. Subsumption reasoning was used in the classification process, thus inferring the ongoing activity. Knowledge-driven ontology model as shown in Figure 2.4, follow OWL theories for the specification of conceptual structures and their relationships [141]. OWL has been widely used for modelling human activities for recognition, which most times involve the description of events by their specifications using their object and data properties [28]. In ontology modelling, domain knowledge is required to encode activity scenarios, but it also allows the use of assumptions and common sense domain knowledge to build the activity scenarios that describe the conditions that drive the derivation of the activities [97]. Recognising the activity then requires the modelled data to be fed to the ontology reasoner for classification. The authors of [30, 96] and [28] followed generic activity knowledge to develop an ontology model for the smart home users. While these approaches to model activities following common sense, everyday domain knowledge and its associated heuristics are commendable, they may lead to faulty activity recognition due to lack of specificity of the contexts describing the activity situations. Specific considerations to object use for routine activities could be put into accounts just as home settings and individual object usage differ which may not apply to ontologies developed from generic knowledge of object usage. They also do not follow evidenced patterns of object usage and activity evolution as they rely on generic know-hows to model activity and object concepts.

2.4.3 Hybrid Approaches for Activity Recognition

Hybrid approaches combine data-driven and knowledge-driven approaches. In most cases, data-driven approaches are applied to augment the knowledge-driven approach. Research efforts by [118, 111, 97, 96, 53] show that hybrid approaches are promising. An example of a hybrid based activity recognition is the Context-aware Activity Recognition (COSAR) by Riboni and Bettini [111]. The COSAR system recognises activities using ontology and statistical reasoners which are based on ontology and multiclass logistic regression method respectively. It relies on the statistical pre-classification of considered activities which are integrated to the ontology reasoner. Although this work shows how recognition rate improves, it failed to consider detail specification of object use and interactions for particular activities which might impact on the recognition of fine grain activities. Kun Gu et al [53] proposed an activity recognition assistance algorithm based on a hybrid semantic model. The hybrid semantic model proposed was based on the HMM and an activity ontology model. The system infrastructure included a search engine module which performs internet information retrieval for its ontology module for activity recognition which is connected to a 3 layer HMM module to provide assistance. The strength of the work is in its ability to provide assistance based on recognised activities, but it failed to mention how retrieved internet information is filtered since object search return multiple results leading to ambiguous activity recognition. Okeyo et al. [97] proposed a hybrid approach which combined ontology and temporal formalism and a dynamic sensor data segmentation technique [96], based on shrinking and expanding the time windows. Despite exploiting the 4D approach for their temporal formalism, they modelled object and activity concepts from generic knowledge of object use and wiki-know-how rather than accurate object use discovery which we propose in this thesis. Okeyo et al. [97], this thesis relies of the discovery of the objects use for activities to model object and activity concepts which proves to be more accurate for the specification of activities and the eventual recognition of activities.

Hybrid approaches have not been purely limited to the combination of data and knowledge-driven approaches. A hybrid generative and discriminative model was proposed by Ordonez et al [99]. This pure data-driven hybrid approach used Artificial Neural Network (ANN) and Support Vector Machine (SVM) within a Hidden Markov Model (HMM) framework. Although the approach provided a unifying data-driven hybrid framework for the recognised set of activities, it lacked semantic expressivity and knowledge inference as with the ontology-based hybrid approaches. A major drawback found with most existing hybrid approaches is their inability to consider accurate knowledge of object use and interactions for the specification of routine activity concepts [99, 97, 96, 53, 111], which should be an inherent feature of activity recognition, and that can serve as contexts describing activity situations in the home environment. In cases where such prior knowledge is not available, such techniques become virtually unemployable. A significant step in the recognition of activities is the accurate discovery of the objects used for specific routine activities.

2.5 Discussion

As discussed in the sections above, data-driven approaches and knowledge-driven approaches follow different methodologies and procedures. They also have with them their various limitations and drawbacks. In spite of this, the strengths of these approaches have led to different commendable research efforts. In relation to this, a summary of the discussed works with emphasis on their strengths and weaknesses is presented in Table 2.2.

Data-driven approaches, in general, have the advantage of handling incomplete data and managing noisy data. Also, they can handle uncertainty much more than knowledge-driven approaches. The review above also highlighted the weaknesses of Bayesian Networks at managing temporal information. DBN, HMM and CRF support modelling temporal information and so could be used for modelling interleaved and concurrent activities. With efforts made with the HMM as in Modayil et al. [70], interleaved activities could be modelled with time take into consideration. The dependence of the HMM on limited observation sequence led to extensions like HSMM, FHMM, and CHMM. The LDA also has the compelling feature of being able to allocate words in a corpus of documents to topics. This feature was put to use in Katayoun and Gatica-Perez [73] and Huynh et al. [65] where the sensor features corresponded to words and the activity topics corresponded to the topics of the LDA. However, the data-driven models are very reliant on data. Activity recognition results in most cases lack semantic expressiveness understood by the end user. Features for classification are very limited due to the inability of context-rich features to be incorporated in these models. With these, it is hard to adequately reflect real life situations as features to be classified in the process of activity recognition using data-driven models.

On the other hand, knowledge driven approaches build on some of the weaknesses of the data-driven approaches. As discussed, they rely on logic, ontology theories which enhance their expressiveness [28]. Domain knowledge can then be encoded in different re-usable knowledge structures for maintaining relationships between activities, objects and temporal and spatial contexts. In most cases, Knowledge-driven approaches can be reusable ontologies which support the common use and sharing across applications. But then, knowledge driven approaches are purely dependent on expert's domain knowledge [97, 30]. These characteristics limit it opportunities amongst models. They are static and unable to handle fuzzy and uncertain situations unless they are extended. Unlike data-driven knowledge, they do not automatically learn the most optimal activity models to characterise activities and sensor data from a set of possible models. Some knowledge-driven methods, e.g. those using ontological, rule-based and case-based reasoning, do not provide inherent support for handling temporal information. Their dependence on knowledge and domain expert's results to generic models being built from the assumptions, and common everyday knowledge of activity situations which may not adequately represent specific settings or home user.

Activity recognition can benefit from both data and knowledge driven approaches together – hybrid models which can utilise data-driven methods to discover object usage patterns and these to be used as knowledge concepts and contexts by knowledge-driven methods to recognise activities. Existing hybrid models are unable to consider accurate knowledge of object use and interactions for the specification of routine activity concepts. Home environments and settings have been known to be varied, and user habits are equally different. Individuals perform activities differently; hence, it is difficult to use generically built knowledge-driven models or models developed from assumptions and common everyday knowledge of object usage for activity situations. It can be argued that models should progress to evolve methods that are unique to home settings and user specific taking advantage of activity patterns and the home settings. The object usage for specific activity situations can be acquired using a data-driven approach which is then encoded as activity and object concepts using a knowledge driven technique to build more adaptable recognition system. This also provides the domain expert with the needed knowledge to encode onto the knowledge base. With this combination, the strength abilities to handle uncertainties by data-driven techniques and semantic expressiveness and clarity of knowledge-driven techniques can be harnessed to recognise activities in a hybrid model proposed in this thesis. More specifically, the compelling word-topic allocation feature of the LDA can also be utilised to discover object usage for specific routine activities. Given these, the work in this thesis harnesses the complementary strengths of data and the knowledge driven techniques to overcome the limitations and challenges highlighted above as a hybrid activity recognition approach built on the activity ontology augmented by LDA topic model.

	Classification Methods and Techniques			
Approaches	**Methods**	**Techniques**	**Strengths**	**Weaknesses**
Data-Driven	Generative	(DBN: Tapia et al. [129]), (HMM: Modayil et al. [70], Patterson et al. [104] Patterson et al., [105]), (NB: Tapia et al. [130], Langley et al. [80]), (LDA: Huynh et al. [65], Rosen-Zvi and Griffiths [113], Katayoun and Gatica-Perez [73]), (HSMM: Hongeng and Nevatia [59]), (FHMM: Kulic et al. [78]), (CHMM: Ou et al. [101])	Flexible models allowing classifications from joint probabilistic distributions of observed samples. Can handle noisy data and uncertainties.	Unable to incorporate some context rich features for activity recognition. Traditional HMM cannot handle temporal information in some cases. Recognition results lack expressiveness.
	Discriminative	(NN: Lee and Mase [82], Huynh et al. [64]), (DT: Boa et al. [13]), (SVM: Cao et al. [24], Qain et al. [107]), (CRF: Liao et al [86], Huynh et al. [66]), (ANN: Russell [116], Yang et al. [143])	Relies on limited amount of observed data to focus on boundaries for classification. Can handle noisy data and uncertainties.	
Knowledge-Driven	Logic-Based	Kautz et al. [74], Bouchard et al. [19], Lymberopoulos et al. [88], Chen et al. [29]	Semantically expressive and logically clear	Weak at handling uncertainties.
	Mining-Based	Wyatt et al. [140], Palmes et al. [102]	Information retrieval from relevant sources.	Static in some cases.
	Ontology-Based	Gruber [52], Akdemir et al, [4], Fancois et al. [43], Hobbs et al. [57], Chen et al. [30, 28], Latfi et al. [81], Yamada et al. [141], Okeyo et al. [96, 97]	Semantically expressive and logically clear.	Generic and built from assumptions and everyday common knowledge. Weak with temporal attributes.
Hybrid Models	Data and Ontology	Riboni and Bettini [111], Okeyo et al. [96, 97], Kun et al [53]	Semantically expressive and logically clear	Poor specification on accurate object and activity concepts.

Table 2.2: Summary of Activity Recognition Approaches

2.6 Conclusion

This chapter reviewed research areas related to smart environments such as pervasive computing, ambient intelligence, smart homes, ambient assisted living, and context awareness. This chapter also reviewed current activity recognition approaches. A discussion covering the strengths and the weaknesses was provided with a way of signalling a framework which combines data-driven and knowledge driven approaches towards a hybrid knowledge driven approach.

Chapter 3

KNOWLEDGE REPRESENTATION AND FORMALISM FOR ACTIVITY RECOGNITION

The previous chapter presented a review of the state of the art techniques in activity recognition. This thesis proposes a hybrid activity recognition approach built on knowledge-driven technique augmented by the Latent Dirichlet Allocation LDA topic model. Knowledge-driven activity recognition is an emerging area of research which uses knowledge engineering methodologies to represent ADLs through ontology modelling. Ontology methodology was followed to implement the approach, which includes all of the use case ontologies developed in this chapter. The Ontology model described in this chapter allows the formalisation and semantic expressiveness of activity and object concepts for activity recognition which the data-driven model lacks. This chapter presents knowledge representation, formalism as aspects of ontology engineering methodology highlighting its implementation for activity recognition, types of ontology and languages and representations.

3.1 Semantic Web

The semantic web was introduced was introduced as a standard for computers to read data on the internet. This involves the contents of the web to have precise semantics and meaning for web contents and resources [56]. The aim of this is to enable people and machines to cooperate by equipping the device with semantic tools and resources. Semantics as a branch of linguistics is concerned with the meaning of words and analysis of these words [56] with the sense in it, references, implications and relations between the words. So, machine processing of web contents equipped with knowledge representations provides the information for conclusions to be drawn. “The Semantic web” as coined by Tim Berner-

Lee, describes how the internet could evolve so that web contents could have meanings and knowledge representations based on information to enhance the use of computers by humans [17]. The introduction of machine enabled semantics supports data and information integration. Through this platform supported by semantic web, information is shared on the web. The semantic web is based on knowledge representation, modelling and features for reasoning accessible on the internet [93]. The vision of the network permits models which are re-useable and easily shared across the platform. Specifications and conceptualisation which allow integration are also another feature which helps to enhance interlinking of information and data [56]. Over the years, the semantic web has evolved to technologies like Resource Description Framework (RDF) [134], OWL [134] [49], DL [12], Semantic Web Rule Language (SWRL) [61], SPARQL Protocol and RDF Query Language [135], JENA [92], Java Expert System and Shell (JESS) [44] and many more.

3.2 Ontology

Gruber [52] defined ontology as "a formal specification of a shared conceptualisation". By this, it provides the formalism for modelling concepts which allows humans and computers to interact especially in specific domains. The vocabularies used in ontology enhances its expressiveness and its utilisation in the modelling of relationships between concepts. The interoperability, information integration and reusability are key strengths of ontology. With the integration of semantics to ontology, data processed and analysed by computers tend to be human readable from which conclusions can be drawn. This has led to the development of the ontology in e-government [121], bioinformatics [108] amongst many. The complex nature of the problems to which ontology are applied has led to ontologies regarded as knowledge bases or repositories of information. These knowledge bases link specific information through data for which contextualised interpretations can be made. This process involves the knowledge content of a domain to be encoded onto the ontology through data or word of choice as concepts. The concepts are then interlinked in the ontology based on the relationship established or enforced in the ontology. By way of inferential reasonings, results are achieved depending on what is being sought. Authors have had different classifications for ontologies depending on the implementation, but this thesis recognises the two classifications by Roussey et al [23] i.e. the first classification is based on language expressivity and formality and the second classification relies on the scope of the objects described in the ontology. The language expressivity and formality classification focus on the concepts, instances and properties referenced to one or more symbols which could be entities e.g. "Martin Luther King" as an instance of a concept "person". The symbols are terms humans can easily interpret and understand by simply reading them. The method could be seen as a declarative method of implementing the concept hence providing a "know that" sort of expressive interpretation of the data. The second classification of ontologies based on the scope of the objects described in domain ontologies. They are modelled to carry the fundamental concepts of a domain e.g. Hydrontology developed specifically for hydrographic representations. Domain-based ontologies can be shared and re-used by different applications belonging to the specified domain. In a way, these domain based ontologies could be said to be generic and provide "know how" sort of expressive interpretation of the information sought to make them procedural.

3.3 Knowledge Representation and Formalism

This thesis follows knowledge representation and its formalism for the development of a hybrid activity recognition approach. Knowledge representation and reasoning aim to design and implement technology solutions that reason about machine interpretable representations of real world problems similar to human reasoning [77]. A knowledge base stores the computational model of the knowledge representations in the form of symbols and statements and then performs reasoning by the manipulation of the symbols and statements. We illustrate knowledge representation using an example from an activity recognition scenario.

Scenario: In the home environment having *"Rooms"* as different locations in it. The *"Objects"* in a *"Room"* are used to performed *"Activities"* which are *"ADL"*.

The activity recognition scenario as illustrated in Figure 3.1 is a semantic network with nodes representing the concepts while the arcs represent the relations between the concepts. The network of nodes then provides the means to abstract from natural language to represent knowledge in the form of text suitable for computation. This is exemplified with concepts like *"action"*, *"activity"* or *"object"*" would be linked with relations `hasUse` or `isUsedfor`. This knowledge and its formalism can be represented in different languages, forms and rules which we consider next.

3.3.1 Description Logic

Description logics (DL) are a family of knowledge representation languages. They are decidable fragments of first-order logic and at the same time expressive enough such that

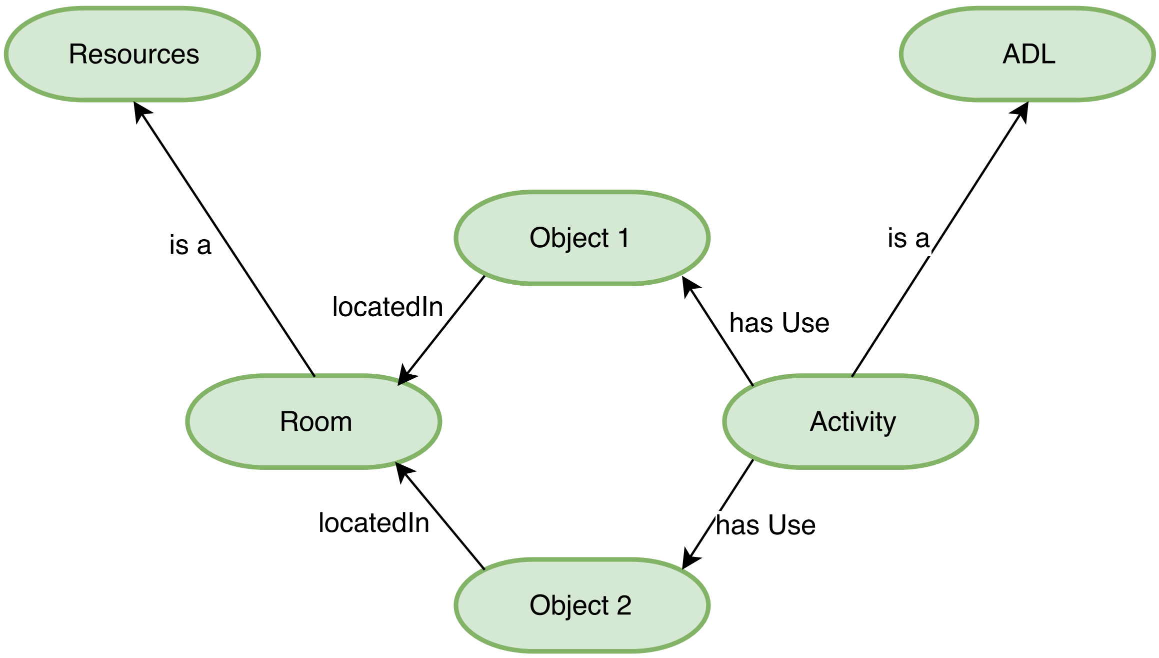


Figure 3.1: Knowledge representation of activity from an object use

they have become a major knowledge representation paradigm. They are equipped with a formal semantics which allows humans and computer systems to exchange DL ontologies without ambiguity as to their meaning [77, 7, 11]. This makes it possible to use logical deduction to infer additional information from the facts stated explicitly in an ontology which is an important feature that distinguishes DLs from other modelling languages. The capability of inferring additional knowledge increases the modelling power of DLs but it also requires some understanding on the side of the modeller. DL ontology consists axioms which must be a true reflection of a situation being described. The axioms can represent only partial knowledge of any situation of the ontology which can be expressed in different ways as long as it is consistent with the ontology. While this may be true, the axioms are categorised into three groups based on terminology: assertional ***ABox*** axioms, terminological ***TBox*** axioms and relational ***RBox*** axioms [127, 7]. The ***ABox*** axioms are used to capture knowledge about name individuals. They specify an instance of a concept as an individual for example asserting *"Breakfast"* as an individual of the concept *"Activity"*. The ***TBox*** axioms describes the relationships between concepts. Unlike ***ABox***, ***TBox*** can be used for concept inclusion and subsumption as in Object $\sqsubseteq$ Resource. This implies the inclusion of the concept of Object as Resource. If the $\sqsubseteq$ in the example given is replaced with $\equiv$, it the asserts equivalence between the two concepts. The ***RBox*** axioms makes reference to properties and could be used in expressing the assertion of inclusion. In the ***RBox*** axiom `timesliceOf` $\sqsubseteq$ `hasBeginning` refers to `timesliceOf` is a sub-time of `hasBeginning`. In DL, the range features are determined by the power of expressivity of the language used. Most common of these language are the Attributive Concept Language with Complements (***ALC***) logic representation scheme. ***ALC*** implies the AL with full existential support and union. Other versions of DL has been derived by the use different letters such ***SHOIN***, ***SROIQ*** etc. [77].

3.3.1.1 The Description Logic *ALC*

The ***ALC*** constructors allows for concept negation ¬ C, the intersection and union of concepts C and D are expressed C ⊓ D and C ⊓ D respectively. A brief summary of the concept contructs for the Description Logic ***ALC*** and first order logic symbols are given in Table 3.1.

Translation of concepts with Boolean construct is possible to first-order logic like C ⊓ ¬ D can be translated to C(x) ∧ ¬ D(x). Also C ⊑ D ⊔ ¬ E can be translates to ∀ x : (C(x) → (D(x) ∨ ¬ E(x))). In ***ALC*** a role r is allowed as an entity which could be used relations between entities. The composite concept ∀r.C translates to ∀ y : (r(x, y) → C(y)) in first-order logic, while ∃ r.C translates to ∃ y : (R(x, y) ∧ C(y)). With reference to the activity recognition scenario above ***TBox*** statements like,

$$Breakfast \sqsubseteq Activity \tag{3.1}$$

This statement would encode the knowledge that Breakfast is an Activity.

$$Breakfast \sqsubseteq \exists \quad hasUse.(Microwave \sqcup PansCupboard). \tag{3.2}$$

This implies that Breakfast can be made using Microwave or Pans Cupboard. The ***ALC*** allows to state that some individuals belong to (named or composite) concepts, e.g. C(a) states that the individual a belongs to concept C. An ALC knowledge base consists of an ***ALC ABox*** and an ***ALC TBox*** [77, 127, 7].

3.3.1.2 The Description Logic *SROIQ*

The DL ***SROIQ*** is one of the most expressive DLs. Every DL ontology is based on three finite sets of signature symbols i.e. a set NI of individual names, a set NC of concept names and a set NR of role names. A ***SROIQ*** role expression over this signature is a role name, the inverse of a role name, or the special symbol U (universal role). It also considers every

Syntax	Meaning
$\sqcup$	**Union**: $X \sqcup Y$ is the set of all objects which are in X, in Y, or in both X and Y.
$\sqcap$	**Intersection**: $X \sqcap Y$ is the set of all objects that are members of both the sets X and Y.
$\equiv$	**Equivalent**: $X \equiv Y$ is true only if both X and Y are false, or both X and Y are true.
$\sqsubseteq$	**Inclusion**: $X \sqsubseteq Y$ all the contents of X object are also contained within Y
$\neg$ $\sim$	**Negation**: $\neg X$ is true if and only if X is false.
$\rightarrow$ $\Rightarrow$	**Implication**: $X \Rightarrow Y$ Y is true only in the case that either X is false or Y is true.
$\leftarrow$	**Inverse Implication**: $X \leftarrow Y$ implies if not X then not Y
$\longleftrightarrow$ $\Longleftrightarrow$	**Equivalence**:$X \Leftrightarrow Y$ is true only if both X and Y are false, or both X and Y are true.
$\wedge$	**Conjunction**: $X \wedge Y$ is true if X and Y are both true; else it is false.
$\vee$	**Disjunction**: $X \vee Y$ is true if X or Y (or both) are true; if both are false, the statement is false.
$\forall$	**For All**: $\forall\ x$: P(x) or (x) P(x) means P(x) is true for all x.
$\top$	**Concept with Everything**: The statement $\top$ is unconditionally true.
$\perp$	**Empty Concept**: The statement $\perp$ is unconditionally false.
$\exists$	**Exists**: $\exists\ x$: P(x) means there is at least one x such that P(x) is true.

Table 3.1: Construct for Description Logic ***ALC*** and First Order Logic

concept name, $\perp$ as a concept expression [60, 77] . A set of ***SROIQ*** concept expressions is defined as:

$$C ::= NC|(C \sqcup C)|(C \sqcup C)|\neg C|\top| \perp |\exists R.C|\forall R.C| \geqq nR.C| \leqq nR.C|\exists R.Self|\{NI\}, \tag{3.3}$$

Where in C represents concepts, R is a set of roles, and n is a non-negative integer. Axioms are built from concept expressions, role expressions and individual names. ***ABox*** axioms are of the form C(a), R(a, b), a $\approx$ b, or a 0 b; ***TBox*** axioms are of the form C $\vee$ D or C $\equiv$ D; ***RBox*** axioms are of the form R $\vee$ T, R $\equiv$ T, R $\circ$ S $\vee$ T, or Disjoint (R, S). Table 3.2 shows a summary of the syntax and semantic of ***SROIQ*** constructors [77] .

	Syntax	Semantics
Individuals		
Individual Name	a	a^I
Atomic Role	R	R^I
Inverse Role	R^-	{ (x, y)| (y, x) $\in R^I$ }
Universal Role	U	Δ^IxΔ^I
Concepts		
Atomic Concept	A	A^I
Intersection	$C \sqcap D$	$C^I \cap D^I$
Union	$C \sqcup D$	$C^I \cup D^I$
Complement	$\neg C$	$\Delta \backslash C^I$
Top Concept	$\top$	Δ^I
Bottom Concept	$\bot$	$\oslash$
Existential restriction	$\exists R.C$	{ x|some R^I -successors of x is in C^I}
Universal restriction	$\forall R.C$	{ x| all R^I -successors of x are in C^I}
At-least restriction	$\geqq nR.C$	{ x| at least nR^I -successors of x are in C^I}
At-most restriction	$\leqq nR.C$	{ x| at most nR^I -successors of x are in C^I}
Local Reflexivity	$\exists R.Self$	{ x|($x, x) \in R^I$ }
Nominal	$\{a\}$	$\{a^I\}$

where a, b $\in$ NI are individual names, A $\in$ NC is a concept name, C, D $\in$ C are concepts, R $\in$ R is a role

Table 3.2: Construct for ***SROIQ***

3.3.2 Resource Description Framework

Resource Description Framework RDF as a major component of the semantic web and knowledge representation and formalism allows the specification of the semantics of data based on XML in a standardized, interoperable manner. It also provides mechanisms to explicitly describe resources using a graphical data model [7, 11]. The RDF and RDFS graphical data model represent properties or relations between entities in the form of triplets to describe resource of interest. In addition, the graphical model represents information as a labelled, directed multigraph with vertices and labelled edges (multiple edges with different labels between the same nodes are allowed). The vertices consist of Internationalized Resource Identifier IRI representing abstract "things", literals of concrete data values and nodes. Typically, a graphical can be expressed as a set of <subject, predicate, object> triples, each interpreted as an edge labelled with "predicate" going from the "subject" node to the "object" node for example *Breakfast* `hasUse` *Microwave*. Individuals are instances which belong to classes as *Breakfast* is an *Activity*. Properties such as `hasUse` can relate individuals of specific classes, for example one can specify that the object of the property `hasUse` belongs to class Activity and the subject belongs to the class Resource. Classes of the object and the subject of a property are abbreviated as domain and range respectively. Relationship between classes and properties can be also specified, for example is can be stated that *Microwave* is a subclass of *Resource*. The resultant graphical model and the associated semantics allows for sub graphs and inferential instances.

3.3.3 Web Ontology Language OWL

In realisation of the objectives of the Semantic Web standards for formal machine understandable semantics, Web Ontology Language was conceptualised. It build upon DARPA Agent Markup Language (DAML) [5] and Ontology Interchange Language (OIL) [40] and compatible with RDF for describing concepts and properties of objects. OWL extends RDF/RDFS and offers increased expressiveness over the RDFS description for RDF. OWL exists with three variants OWL-Full, OWL-DL and OWL-Lite [16].

OWL-Full is the fully compliant with RDF and has been the most routinely used version of OWL. It is the most expressive variant of OWL but is not supported by OWL reasoners and retain decidability. OWL-DL is based on the ***SHOIN-D*** description logic. Its constructs are Concept negation, union, intersection, value and existential restrictions and transitive properties (S), subproperties (H), nominals (O), inverse properties (I), unqualified cardinality restrictions (N) and Datatypes from the acronym ***SHOIN-D*** was formed. OWL-Lite is a subset of the OWL-DL. It is less expressive than OWL-DL but allows for the definition of class hierarchies and simple constraint features. It is based on ***SHIF-D*** description logic which supports the concepts of negation, union, intersection, value and existential restrictions and transitive properties (S), subproperties (H), and inverse properties (I) and functional properties (F). Reasoning over OWL-DL is non deterministic exponential in time (NExpTIME) although, in practice, optimised tableaux based reasoners offer tractable average case running times. Unilke OWL-DL, it is deterministic exponential in time although, average case complexity is lower [11].

OWL 2 extends OWL-DL with additional constructs. OWL 2 is based on ***SROIQ-D*** description logic which offers all the constructs of ***SHOIN-D*** and in addition qualified number restrictions (Q) with complex role inclusion axioms (R) [111]. These properties also offers that is decidable with additional expressiveness whilst retaining the computational properties of OWL-DL.

This thesis follows the OWL specification which are similar to those of the Description Logic which includes abstract syntax. The names specifically used for the classes represent the concepts for example, if C_1, $C_2...C_n$ are implemented as classes in an OWL, they represent concepts and keywords in that OWL. The `intersectionOf`(C_1, $C_2...C_n$) and `unionOf`(C_1, $C_2...C_n$) represent the intersection and disjunction of classes C_1, $C_2...C_n$.

The complements of the respective classes C_1, C_2...C_n are defined using keywords `complementOf`(C_1), `complementOf`(C_2)...`complementOf`(C_n) for each of the classes. For each of the classes, there could a set of individuals (O_1, O_2...O_n) is defined using the declaration `oneOf`(O_1, O_2...O_n). If P represents property, restrictions (expressed using the restriction keyword followed by the property P over which the restriction applies and the restriction keyword) can be any of: `someValuesFrom`(C), `allValuesFrom`(C), `hasValue`(O), `minCardinality`(n) and `maxCardinality`(n) representing qualified existential restrictions, value restrictions, exact value restriction, min and max cardinality restrictions respectively, where C is a class name, O an individual (or datatype value), and n an integer. An enumeration using the `oneOf` keyword can be used instead of a class name C in the above definitions. `SubClassOf`(C_1, C_2...C_n) asserts that C_1 to C_n are sub classes of the parent class of reference. `EquivalentClasses`(C_1, C_2...C_n) and `DisjointClasses`(C_1, C_2...C_n) represent respective class equivalence and disjointness for the list of classes named. `SubPropertyOf`(P_1, P_2...P_n) represents the subproperty relation between properties P_1 and P_n, while property equivalence is defined using `EquivalentProperties` keyword. domain and range corresponds to domains and ranges arguments respectively for the classes to which they point, while keywords `inverseOf, Symmetric, Asymmetric, Functional, InverseFunctional, Transitive, DisjointProperties, Reflexive` and `Irreflexive` are used to indicate the properties on which they are applicable. Also, `owl : Thing` and `owl : Nothing` represents the top $\top$ and bottom $\perp$ concepts respectively.

3.3.4 Semantic Web Rule Language SWRL

The SWRL is an acronym for Semantic Web Rule Language. It is the rule language applicable to semantic web ontologies. The rule are expressed in terms of OWL concepts (classes, properties, individuals) using Horn clauses which are disjunction of classes with at most one positive literal. For example, first order notation like $\neg$ A $\vee$ $\neg$ B...$\vee$ C can be written as A $\wedge$ B $\wedge$... $\implies$ C (See Table 3.1 above). SWRL extends OWL expressiveness whilst retaining decidability. The intersection of properties over named individuals can be expressed using SWRL which is not part of OWL [61, 62]. With this, it has an edge over OWL as an important tool for embedding rules into an ontology. In this thesis, we use SWRL rules which we present in both the first order notation and in some cases the corresponding SWRL notation. This is because Horn clauses have reasoning efficiency using forward chaining rules engine based available data from the constructs, notations and inference rules to extract more data until a goal is reached. The antecedent (body) of the rule is regarded as a conjunction of clauses and the consequence (head) is one positive. The conjunction of clauses in the consequence part of the rule can be expressed indirectly by a set of rule which have identical antecedents. A class with C concept and P property can have clause in the rule expressed as C(`?x`)), property names P (in the form P(`?x, ?y`) where x, y are variables). In cases where the antecedent of the rule holds for a given set of variable instantiations, the consequence is asserted into the knowledge base. Disjunction and negation of clauses are not supported in the body and they cannot appear as a consequence of the rule. SWRL rule also support specific build ins numerical operators and datatypes.

3.3.5 SPARQL

SPARQL Protocol and RDF Query Language (SPARQL) is a query language for RDF. SPARQL works by graph matching of the query criteria in the form of RDF triples specified as the subject predicate and object discussed in the sections above. This set of triple patterns, called the basic graph pattern, defines the graph patterns that has to be matched to a target dataset [106]. With this, SPARQL queries can be used on a diverse set of data as long as they are stored or saved as RDF. The capabilities SPARQL includes considerations of RDF triples with their conjunctions, disjoints, negations as well as support for aggregation. SPARQL queries are similar to SQL which allows results to be sorted and filtered from duplicates. An extension of SPARQL is the Temporal Ontology Querying Language (TOQL) which has the expressive power for handling time queries [14]. In addition to the strengths of SPARQL, TOQL supports queries on time evolving information instantiations to an ontology using allen operators that allow comparisons between time intervals, and the operator **AT** that allows comparisons between time points or time intervals. Typical TOQL query structure would involve syntax like:

- **PREFIX**: This declares the namespaces used in the query relative to the target dataset.
- **SELECT**: A select clause in the query defines the variables to which the query is bound.
- **FROM**: This clause is an optional clause. It is used for the specification of the target dataset.
- **WHERE**: The **WHERE** clause specifies the graph pattern to match against the data graph. This pattern in the format of the RDF triple like the subject-predicate-object searches for the look alike match of the query in the dataset e.g. *Breakfast* `hasUse` *Microwave* which implements *Microwave* is used for *Breakfast*. An optional **WHILE** clause included implements cases when the **WHERE** clause is true

and present and a **UNION** clause returns results for multiple graphs matches. **FILTER**: The **FILTER** clause is used to add constraints and restrictions which may entail the use of variables.

- Other clauses and constructs applicable to TOQL query includes **DISTINCT** used to distinguish results, **ORDER BY** used as a sequence comparator of query results, **LIMIT** construct puts an upper or a lower bound in terms of number to the result and **OFFSET** used for the specification of the results after generating a number of other results.

The approach proposed in this thesis adopts TOQL query language as the preferred means of retrieving activity results from the knowledge base of activity ontology. It build TOQL queries to capture observe object usage data to retrieve activity results from an RDF based activity ontology by matching the query to the RDF activity ontology. A simple TOQL query can look like this:

$$\begin{array}{l} \textbf{SELECT} \;\; \texttt{Activity Name} \\ \textbf{FROM} \;\; \texttt{Activities} \\ \textbf{WHERE} \;\; \texttt{Activities hasUse Object} \\ \textbf{AT} \;\; \texttt{Time} \end{array} \tag{3.4}$$

This TOQL query retrieves the *Activity* which is an *ADL* at a particular time which could be instantiated with specific values using the clause **AT**, directly filtering the results to those matching the query.

In addition to the SPARQL and TOQL query, is the recently developed SPARQL update language. The SPARQL update is used for the specifications and execution of updates, changes and modifications to Ontologies and RDF graphs [46]. Typically, SPARQL updates work oppositely to the SPARQL query. While the query performs retrieval operations, the update adds and in some cases deletes to modify the ontology. To make modifications and changes to ontologies, the SPARQL update uses the update operations **DELETE** and **INSERT** in addition to **WHERE** and the **PREFIX** similar to the SPARQL query. The

- **INSERT** data to insert data in the form of new triples into an ontology or RDF graph
- **DELETE** data seeks to remove data in the form of known triples from an ontology or RDF graph.

Both operations are applied to a part of the ontology or RDF graph with the clause **WHERE** similar to the query where the resulting RDF triples get removed from and added to the data or both. With these update operations, minor changes and modifications can easily be carried out on the ontology graph thus saving time and bypassing the process of editing the entire ontology. This thesis applies SPARQL update operations similar in expressiveness to the TOQL to make changes to the choice of object usage which typically would have required editing the whole ontology using Protégé which could be cumbersome and time-consuming.

3.4 Conclusion

This chapter presented knowledge representation and formalism for the development of activity ontology. It described semantic web, web ontology language, description logic and ontology rules. In the process, the need to acquire knowledge of object use through activity-object discovery to aid information fusion and the development of activity and object concepts in the activity ontology was highlighted. It also presented the TOQL as an extension of SPARQL and the preferred language of object use retrieval to perform activity recognition and the SPARQL update language for making changes to the ontology in the event of object use change to bypass the process of editing the entire activity ontology. The next chapter shall focus on presenting knowledge acquisition and context description for activity ontology.

Chapter 4

THE PROPOSED HYBRID APPROACH AND KNOWLEDGE ACQUISITION FOR CONTEXT DESCRIPTION

This chapter presents the proposed hybrid approach to recognising activities of daily living from patterns of object use which extends activity ontology to include a context description component for enhanced activity recognition. This chapter also presents the knowledge acquisition of the context descriptors for activity ontology. The conventional techniques for the acquisition of knowledge for context describing activity situations and concepts formation to build activity ontology relies on the use of predefined templates, rules, static background knowledge and assumptions from everyday use of home objects. These conventional techniques are not only difficult to scale between different individuals and home settings given their uniqueness. Their activity recognition results also could be far from desirable especially when object use for specific activities have not been predefined of are lacking. This chapter presents a novel technique for acquiring knowledge of object concepts which describes activity situations. This technique involve activity-object use discovery as part of an ontology knowledge acquisition for context description of activity situations. The rationale behind this technique is that routine activities should correspond to separate sets of objects as contexts describing them.

4.1 Proposed Approach

This thesis proposes a hybrid approach to recognise activities conducted in an home environment. The functional intention as illustrated in Figure 1.2 is to give an overview of possible outcomes and outputs with regards to the individual performing activities in the home environment. The system architecture allows for a continuous activity recognition process in the home environment and made available to the family, physicians and care-givers to provide the needed support and assistance that may be required based on the activities recognised. To achieve the goal of activity recognition, the proposed approach supports acquiring knowledge of object use as contexts of the activity situations through activity-object use discovery, information fusion of activity and object concepts, activity ontology design, development and modelling, followed by activity recognition. Figure 4.1 illustrates a conceptual overview of the proposed hybrid activity recognition approach, which proposes a complementary topic model approach through a context description module as an extension to the traditional ontology-driven activity recognition. The architecture is made of two component modules the context description and ontology modules. As a unified approach, the functions of these components modules are integrated to provide a seamless activity recognition platform which takes in inputs of sensor and object observations captured in the home environment representing atomic events of object interactions. The resultant outputs are activities and activity situations. The subsections below describes in detail the component modules.

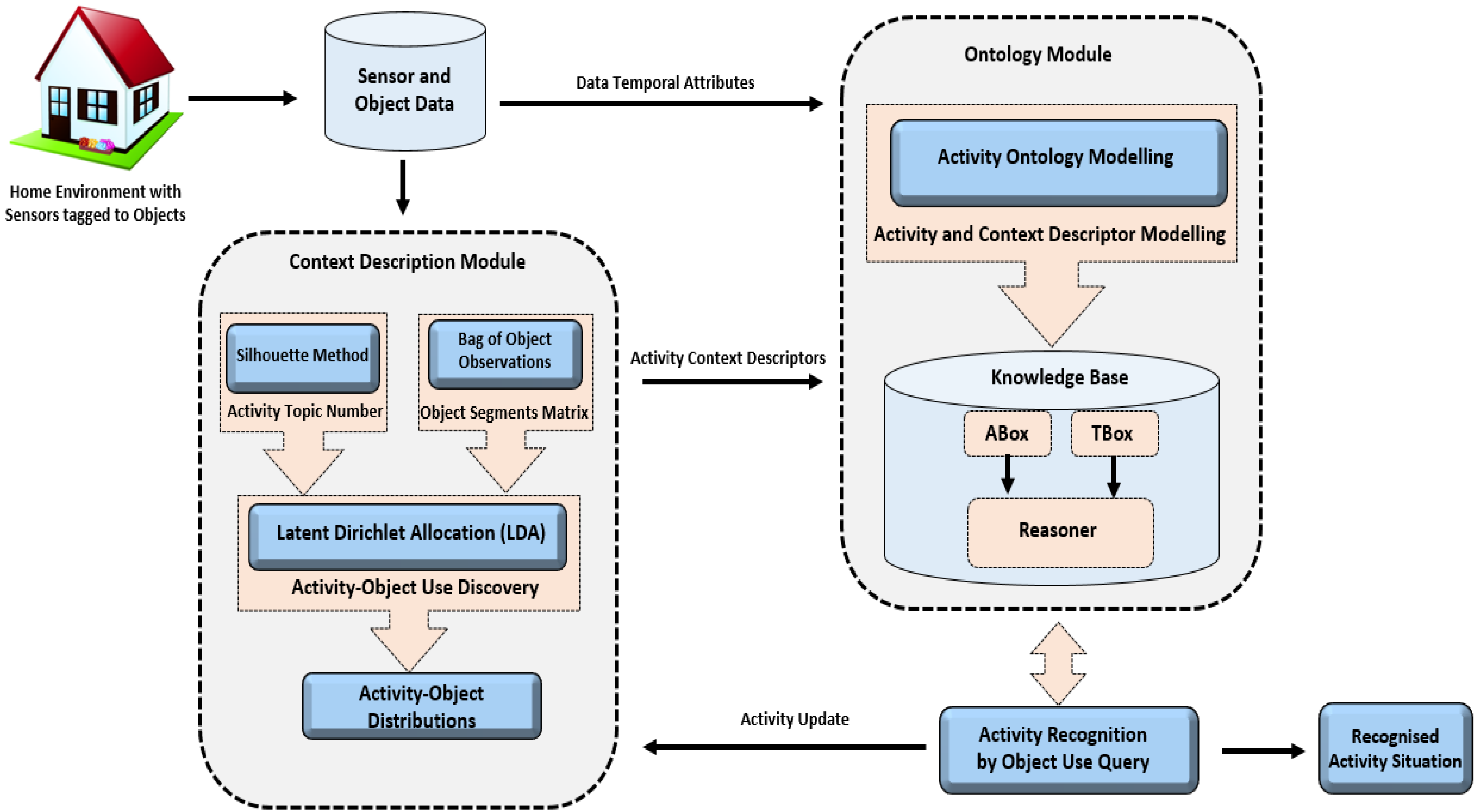


Figure 4.1: An Overview of the Proposed Hybrid Activity Recognition Approach.

4.1.1 Context Description Module

The context description module extends the traditional knowledge-driven activity recognition technique. Its principal function is to provide the knowledge of object usage for specific routine activities. The object usage for specific routine activities are the contexts describing the specific routine activity situations, hence the name context description module. To provide the basis for a hybrid knowledge-driven activity recognition, knowledge of object and activity concepts are required. The dependency on this knowledge is such that activities as high-level events are a result of low-level tasks or atomic events of object interactions. Traditional knowledge-driven activity recognition techniques [30, 96] model ontologies from object use assumptions or common everyday knowledge of object use. This may not be the case in every home setting or the environment as this may lead to faulty object descriptions or low-level tasks and eventually, wrong activities recognised. So to design and model for accurate recognition of these activity situations, there must be a process of acquiring the accurate knowledge of objects or context descriptors that describe activities or results to the activities as higher level events. The context description module performs its function by a process of activity context description which uses the output from an activity-object use discovery process. The modular process is described briefly below:

- **Activity-Object Use Discovery**: The activity-object use discovery uses the generative Latent Dirichlet Allocation LDA topic model to find a correlation between objects and corresponding activities. It does this by generating activity-object distributions in a probabilistic manner using a bag of object observations as inputs and activity topic number as a parameter. The main idea of the LDA of topic models is that documents in a corpus of texts are a mixture of latent topics while the latent topics are probability distributions over words therein. The input are documents represented as "bag of words" and topic number as a parameter so that the output has for the topics probability distribution over the unique words [18]. In the context of activity recognition, observed sensor/object in a dataset, bag of object observations and the activity topic number corresponds to the corpus of texts or words, "bag of words" and topic number of the LDA respectively. To determine the activity topic number which is a necessary parameter for the LDA, an unsupervised silhouette method through K-Means clustering is used. The bag of object observations is constructed by partitioning the sensor/object dataset into sensor/object segments using time window interval. The sensor/object segments are then formed into segment-object-frequency matrix referred to as "bag of object observations". The result using the activity topic number as a parameter and the bag of object observations as input on the LDA are allocations or assignments of objects to specific activities.

- **Activity Context Description**: To satisfactorily assign and allocate the objects as context descriptors, a context description algorithm is used on the objects to activity topics allocations from the step above. This algorithm uses the number of times an object is assigned to an activity topic and a threshold (a measure of the mean and standard deviation of the number times from the collective distributions). An object becomes a context describing an activity situation if it is assigned to that activity by the number of times greater than the threshold. Finally, the activity topics are labelled and annotated in line with the activities carried out with the objects and model these as concepts in the activity ontology.

4.1.2 Ontology Module

The knowledge base is the component and repository of information consisting of the modelled activity ontology concepts, data, rules used to support activity recognition. Just like other knowledge bases, it functions as a repository where information can be collected, saved, organized, shared and searched. The activities and the context descriptors from the context description module are designed, developed following description logic knowledge representation and formalism and added to the knowledge base. The knowledge base is made of the TBox, ABox and the reasoner. The TBox is the terminological box made of the activities and the relevant context descriptors as defined and encoded ontological concepts. The ontological design and development process gradually populates the TBox by encoding the activities and context descriptors from the context description module as ontological concepts. The ABox is the assertional box made of the instances and individuals of the concepts encoded in the TBox asserted through properties which may be object or data properties. For all the concepts in our TBox, instances and individuals are created out of them and asserted through different properties to populate our ABox. In addition to the activity and context descriptor concepts and instances, temporal concepts and and with their instances are added following the 4D fluent approach which is an ontology temporal formalism to allow for a realistic reflection of activity evolution and transition. The resultant activity ontology created with the fusion of likely object use and behavioural information from the activity-discovery component makes it possible for activity inference. The reasoner checks the relationships between the concepts in the TBox and the consistencies in the ABox for the individuals and instances to perform activity recognition by information retrieval. The eventual results from the information retrieval are the activities or activities situations.

4.1.3 Activity Recognition

Activity recognition is performed by querying the knowledge base. With sensor or objects observed along a timeline, a query is set up which reasons this observed sensor information against the activity model in the knowledge base to make activity inference. The activity inference is a result of retrievals from the knowledge base through a process of subsumption and equivalence reasoning of the TBox and ABox. The approach is eventually evaluated for the proof of concept for which this thesis claims based on the activity recognition performance using publicly available datasets. The major objective of this approach is activity recognition through the modules and components described above. For the proposed hybrid approach to achieve this, a subset of sensor/object dataset (training subset) is used for the context description module to get the context descriptors and the corresponding activities. The resulting context descriptors and activities are modelled ontologically to populate the knowledge base (see Figure 4.2 for the methodology and data flow). Activity recognition is carried out using the test subset to retrieve activity situations which the streams of sensors/objects represents. Full detail of the process and experiments are provided in Chapter 6.

4.2 Knowledge Acquisition of Contexts

The use of ontologies and inter-operable semantics are becoming significantly important in the area of activity recognition. The conventional technique of acquiring knowledge of contexts describing activity situations and concept formation to build ontology relies on the use of predefined templates, rules, static background knowledge and assumptions of everyday use of home objects. The problem exacerbates this that recognition of ADL should result from the ways activities can be carried out from the interactions of home objects. To positively impact on this existing problem of knowledge acquisition is to construct and maintain activity ontologies through a complementary semi-supervised or unsupervised learning technique of object usage in the home environment. Knowledge acquisition and context description from object use in the home environment is the process of identifying the object usage for specific routine activities. The knowledge of the object usage for the activities then become the contexts and concepts used to construct and maintain the activity ontology. The task of determining the object used for the routine activities in the home environment then become the fundamental step in the process of knowledge acquisition and

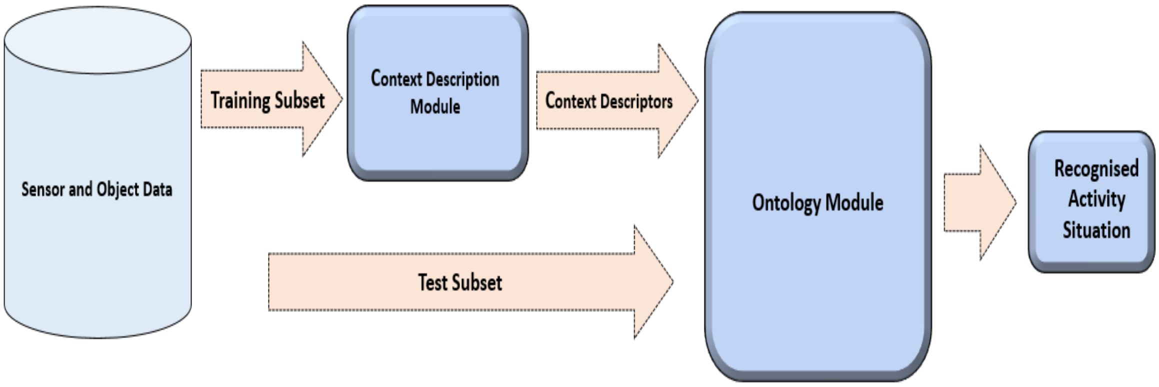


Figure 4.2: Methodology and Data flow

context description for activity situations. The names of the objects as used by the authors [30, 28, 97, 110] in their ontologies are terms turned contexts and concepts which do not have established feature analysis based on their usage for activities. So to say, these object names and the resulting concepts have been created dependent on background knowledge acquired through everyday and common knowledge over the years. For example, the activity *Make coffee* has over the years been known to be composed of the objects use *Coffee*, *Milk*, *Sugar* and *Mug*.

On the other hand, knowledge learning techniques for ontologies apply text and natural language based clustering techniques which rely on contextual cues of the terms as features [50]. Lagus et al [79] noted that, a document could be encoded as a histogram of its word which might not retain information of relatedness. Oliveira et al [98] demonstrated certain properties which K-Means clustering has such as cluster sizes and ability to identify the number of clusters. Also, the generative probabilistic topic models Latent Dirichlet Allocation LDA by Blei et al [18] and the Probabilistic Latent Semantic Analysis (PLSA) by Hofmann [58], learn a set of latent variables called topics from a set of words in the form of documents as inputs. The central assumption of topic models is that documents are generated by a mixture of topics while topics are probability distributions over words. Output are a classification of topic assignments to documents and words. Despite these, the potentials of K-Means clustering and LDA remains relatively unexplored in the process of acquiring knowledge of context describing activity situations for activity ontology.

This chapter presents the context description by explaining the activity-object use discovery process to learn and acquire the knowledge of object usage and the contexts for describing activities for the formation of ontology concepts. The steps to activity-object use discovery and context description presented in this thesis is as illustrated in the Figure 4.3. Towards this, activity-object use discovery as a part of the process involved is presented in section below.

4.3 Activity-Object Use Discovery

One of the primary tasks of activity recognition applications is the discovery of the objects used and the contexts describing the different activities regardless how they are performed. This is very key because specific objects used for routine activities are determined and cases where activities share similar object usage (e.g. *Making Breakfast*, *Lunch* and *Dinner*) could also be discovered. Also, behavioural tendencies and habits may be uncovered in the process. Activity object use discovery aims to identify possible objects used for routine activities. The rationale behind this process is that routine activities should correspond to the use of a separate set of objects. To achieve this, a 3 step approach is proposed which includes:

- **Determining the number of activities**: A key parameter needed by the LDA process is the topic number. In the context of the activity-object discovery, the number of activities corresponds to the topic number of the LDA. The number of distinct activities in a dataset is determined by applying the silhouette method through K-Mean clustering.

- **Bag of Objects Observations**: In this step, the observed sensor or object data stream are partitioned into segments of time intervals i.e. each segment corresponds to sequences of observed sensor or objects within suitable time windows. A segment-object frequency matrix is formed or constructed from the resulting sensor segments.

- **LDA Process**: The LDA process uses the topic number from the first above and the segment-object frequency matrix formed from the "bag of object observations" as inputs to determine the object distributions for the specific activities.

The steps are presented in detail below:

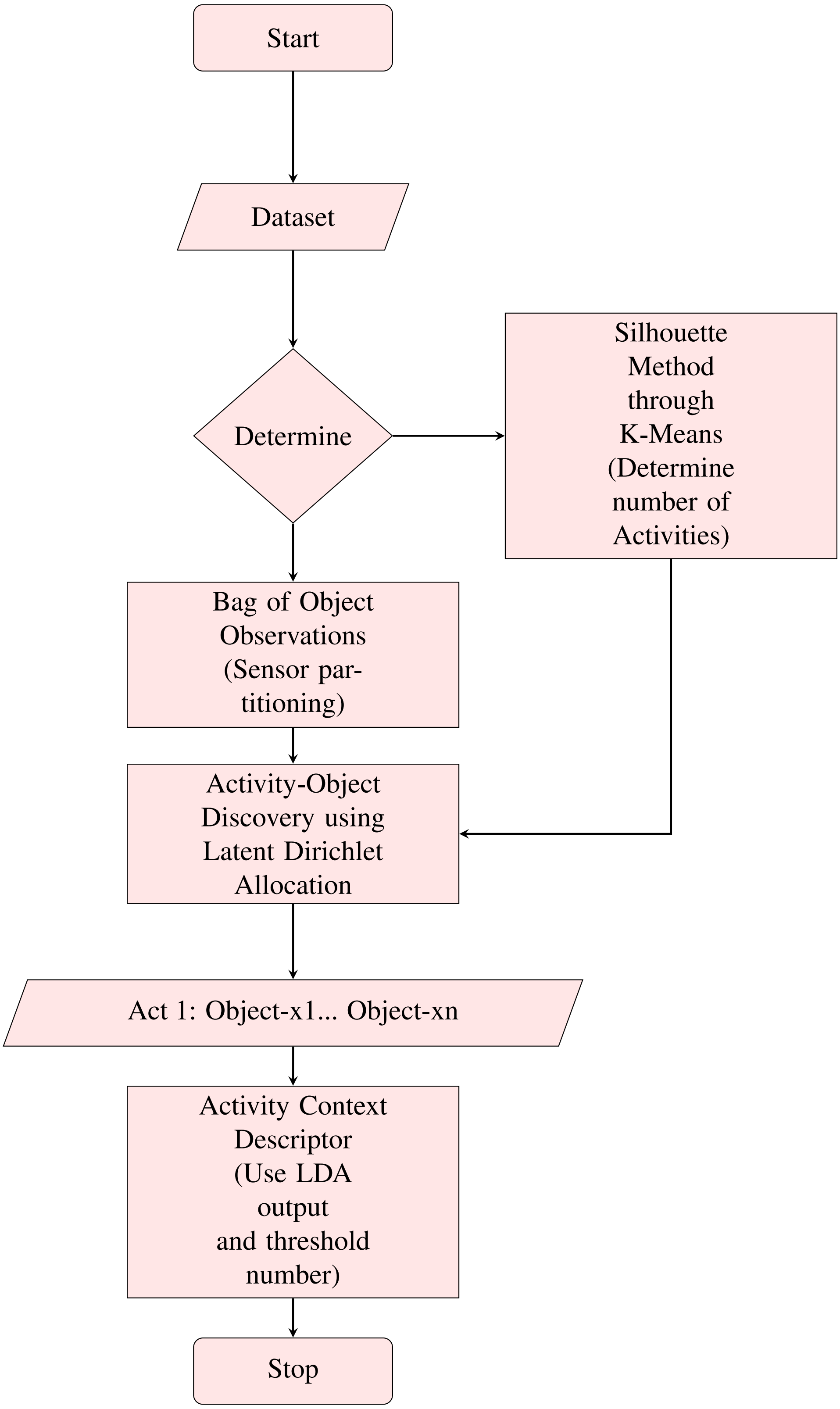


Figure 4.3: Activity-Object discovery and Context Description flowchart

4.3.1 Determining Activity Topic Number

Observed objects in a dataset are a representation of activities which implies that activities can be performed by a set or group of objects. These object groups which represent the activities can be determined by clustering the dataset so that each cluster represent a candidate activity. The major challenge is the estimation of the number of groups or clusters of objects in the dataset, especially where the number of possible activities has not been specified. The number of activities which is the same as the activity topic number is determined by a silhouette method through K-Means clustering. This method searches for the optimal number of clusters in a given dataset. So, in the context of the activity discovery this optimal number of clusters in the sensor or object data is analogous number of activities hence determining the topic number needed as a parameter by the LDA process. But first, the K-Means clustering is explained and how silhouette method can be applied through it. K-means is an unsupervised clustering technique used in partitioning data into clusters. It can be applied to a dataset of N unique observations with the aim to partition into k clusters ($k <= N$), since the number of possible clusters cannot be more than the number of N unique observations. For x_1 ...x_n where each observation is a d-dimensional real vector and are entities as contained in N, the clustering process produces sets of S = S_1... S_k partitions so that $S_k \in S$ is a cluster having centroid c_k calculated by minimizing the within-cluster sum of squares as in equation 4.1. This process is iterated until clusters S_k stabilise.

$$W(S, C) = \sum_{i=1}^{K} \sum_{i \in S_k} \parallel x_i - c_k \parallel^2 \tag{4.1}$$

In the context of activity recognition, K-Means clustering can be applied to partition sensor or object data stream. Huynh et al [65] applied K-Means clustering to partition sensor dataset. They have used the results from the clustering process to construct document of different weights. In a slightly different way, this thesis applies the K-Means clustering to determine the topic number. The clustering process partitions any given dataset into clusters and in this case, each cluster represents a candidate activity which resultant from the object interactions therein. This approached also aims to use this process to determine the number of activities which is a measure of the optimal number of clusters. An optimal number of clusters/activities is an important parameter that will maximize the recognition accuracy of the whole approach. To achieve this, the concept of silhouette width which involves the difference between the within-cluster tightness and separation from the rest is applied. Theoretically, it is a measure of the quality of clusters [114]. The silhouette width of x_i from N as w_{x_i} can be computed from:

$$w_{x_i} = \frac{z_{x_i} - y_{x_i}}{max(y_{x_i}, z_{x_i})} \tag{4.2}$$

where y_{x_i} is the average distance between x_i and all other entities belonging to the cluster and z_{x_i} as the minimum of the averages distances between x_i and entities in other clusters. Normally, the measure of silhouette width values ranges -1 and 1. If all the silhouette width values are close to 1, then the entities are well clustered. The highest mean silhouette width over different values of k then suggests the optimal number of clusters.

4.3.2 Bag of Object Observations

The "bag of objects observation" proposed is analogous to the "bag-of-words" used text and document analysis. In text and document analysis, a document (bag) in a corpus of texts can be represented as a set of words with their associated frequencies independent of their order of occurrence [50]. Thus, disregarding the order of word occurrence, the "bag of words" is a representation of the words in the documents with their frequencies. This "bag of words" approach can be followed to represent discrete observations of objects or sensors of specific time windows generated as events in the manipulation of the home objects. In this regard, it is referred to as "bag of object observations". To satisfactorily achieve bagging of the objects accordingly, the stream of observed sensors or objects data are partitioned into segments of suitable time intervals. By this, the objects and the partitioned segments then respectively corresponds to the words and documents of the "bag of words". If a dataset is given by D made of $x_1...x_n$ objects, D can be partitioned using suitable sliding time window intervals into $d_1...d_D$ segments. Observed objects in each of the segments are then represented with their associated frequencies to form a segment-object frequency matrix. In this thesis, the Kasteren et al [132] and the Ordonez et al [99] datasets has been used. Similar to the documents of texts, these datasets are collections of objects observations which represents activities through object use in the home environment. For the "bag of object observations", the intention is to partition the datasets into segments using suitable time intervals similar to schema 4.3. A sensor-segment frequency matrix is formed from the object counts from each of the segments. Observed objects $x_1...x_n$ in each of the segments $d_1...d_D$ are then represented with their associated frequencies f to form a segment-object-frequency matrix similar to the schema given in Equation 4.4. The objects are then represented as their aliases as in *Seat* (S), *Basin* (B), *Bed* (A), *Microwave* (M), *Cupboard* (C), *Fridge* (F), *Cabinet* (N), *Toilet* (T), *Shower* (Sh) etc. to be encoded onto the partitioned segments and "bag of objects observations". This is further described with the bag of object observation using the scenario below.

$$D = \begin{bmatrix} \{x_1..., x_{n1}\} & objects & in & segment & d_1 \\ \{x_2..., x_{n2}\} & objects & in & segment & d_2 \\ ...... & & & & \\ \{x_{N1}..., x_N\} & objects & in & segment & d_D \end{bmatrix} \tag{4.3}$$

$$BagofObjectObservations = \begin{bmatrix} d_1 & x_1 & f_1 \\ d_2 & x_2 & f_2 \\ & ...... & \\ d_D & x_N & F \end{bmatrix} \tag{4.4}$$

Scenario: The construction of the 'bag of objects observations' is described using a part of the Kasteren House A dataset[1], as illustrated in Figure 4.4. The observed object data are partitioned into segments using a sliding window of 60 s intervals so that objects: *Hall-Bedroom Door* belongs to Segments 1–3; *Hall-Toilet Door*, *Hall-Bathroom Door* and *ToiletFlush* belong to Segment 4; *Hall-Bathroom* belongs to Segment 5; and *Plates Cupboard* and *Fridge* belong to Segment 6. The objects in each of the segments with their associated frequencies form a segment-object-frequency matrix representing the bag of objects observations. Further, the objects are represented as aliases, e.g., *Hall-Bedroom Door* (BE), *Hall-Toilet Door* (TO), *Hall-Bathroom Door* (BA), *ToiletFlush* (TF), *Fridge* (FR), *Plates Cupboard* (PC), etc, to be encoded onto the bag of sensor observation as given in Equation (4.5).

$$BagofObjectObservations = \begin{bmatrix} 1 & BE & 1 \\ 2 & BE & 1 \\ 3 & BE & 1 \\ 4 & BA & 1 \\ 4 & TO & 1 \\ 4 & TF & 2 \\ 5 & TO & 1 \\ 6 & PC & 2 \\ 6 & FR & 2 \end{bmatrix} \tag{4.5}$$

[1]https://sites.google.com/site/tim0306/datasets

Start time	End time	ID
25-Feb-2008 00:20:14	25-Feb-2008 00:22:57	24
25-Feb-2008 09:33:41	25-Feb-2008 09:33:42	24
25-Feb-2008 09:33:47	25-Feb-2008 17:21:12	24
25-Feb-2008 09:36:43	25-Feb-2008 09:37:04	5
25-Feb-2008 09:37:20	25-Feb-2008 09:37:23	6
25-Feb-2008 09:37:51	25-Feb-2008 09:37:52	14
25-Feb-2008 09:37:55	25-Feb-2008 09:37:56	14
25-Feb-2008 09:37:58	25-Feb-2008 09:38:01	6
25-Feb-2008 09:49:27	25-Feb-2008 09:49:28	9
25-Feb-2008 09:49:31	25-Feb-2008 09:49:38	9
25-Feb-2008 09:49:39	25-Feb-2008 09:49:44	8
25-Feb-2008 09:49:53	25-Feb-2008 09:49:56	8

Figure 4.4: A part of the Kasteren House A dataset with Objects/Sensors ID represented as: 24 = *Hall-Bedroom Door*; 5 = *Hall-Toilet Door*; 6 = *Hall-Bathroom Door*; 14 = *Toilet-Flush*; 9 = *Plates Cupboard*; and 8 = *Fridge*.

4.3.3 Latent Dirichlet Allocation (LDA)

The Latent Dirichlet Allocation is a generative classification topic model widely used in the text mining and analysis. The Probabilistic Latent Semantic Analysis PLSA [58] and the Latent Dirichlet Allocation LDA [18] are two topic models extensively used in text mining natural language processing. They work in a similar way requiring inputs of documents represented as "bag of words". The outputs are latent topics and topic assignments for each of the input documents. In addition to this output, topic model have the ability to assign the individual words in the documents to topics according to the frequency of how the words appear in the documents. Due to this appealing characteristics, the LDA has been used to assign objects used to activities in the activity recognition context. The LDA was introduced by Blei et al. [18], in which the documents from the "bag of words" are modelled as a multinomial distribution of topics. It takes advantage of the assumption that there are hidden themes or latent topics which have associations with the words contained in a corpus of documents. It extends the PLSA by the introduction of Dirichlet priors on the distribution over topics for the particular document, θ, and the distribution over words for a specific topic ϕ. A graphical model of the LDA is illustrated in Figure 4.5 and Table 4.1 for symbols used. The generative process follows:

1. For each each topic z_i in Z = $z_1 ... z_k$:
    - choose a distribution over words $\phi \sim$ Dirichlet(β)
2. For each document d_i in D = $d_1 ... d_D$:
    - choose $\theta \sim$ Dirichlet(α)

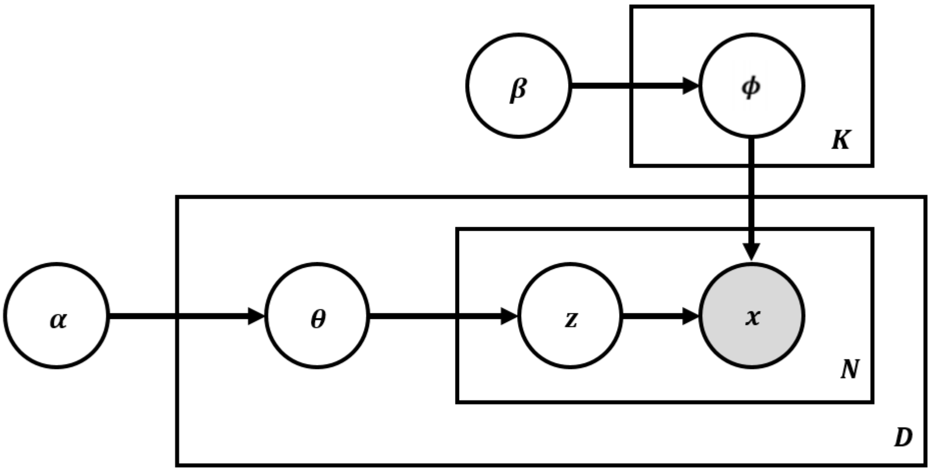


Figure 4.5: Graphical model of the Latent Dirichlet Allocation.

3. For each word x_i in X = $x_1...x_n$:
   - choose a topic $z_k \sim$ Multinomial(θ)
   - choose a word $x_i \sim$ Multinomial(ϕ)

The documents are presented in the form of documents of words $d_1...d_D$. With D composed of words independent of the order, $d_1...d_D$, d_i would be made of words represented as $x_{i_1}...x_{i_n}$ from X words of $x_1...x_n$. The LDA places a dirichlet prior P($\theta_d|\alpha$) with parameter α on the document-topic distributions P($z|\theta_d$). It assumes a Dirichlet prior distribution on the topic mixture parameters θ and ϕ, to provide a complete generative model for documents D. θ describes D x Z matrix of document-specific mixture weights for the Z topics, each drawn from a Dirichlet(α) prior, with hyperparameter α. ϕ is an X x Z matrix of word-specific mixture weights over X words for the Z topics, drawn from β which is a Dirichlet prior. The probability of a corpus, is equivalent to finding parameter α for the dirichlet distribution and parameter β for the topic-word distributions P($x|z,\beta$) that maximize the likelihood $\mathbb{L}$ of the data for documents d = $d_1...d_D$ [18]

In the context of the activity-object use discovery, this approach is applied to discover the activity-object distribution. In this process, the topic number from the silhouette method

X	The object observed with x_i as a unique object.
Z	The latent activity topic assigned to the objects in X.
$X^{\neg i}$	The object excluding x_i.
$Z^{\neg i}$	The object excluding z_i.
K	The number of activity topics determined from the silhouette method above.
U	The number of unique objects.
D	Denotes the number of Segments.
α	Topic dirichlet prior
β	Word dirichlet prior
$\Omega_{d,k}$	The number of object counts in each segment assigned to activity topic k
$\Psi_{k,u}$	The number of object counts in the entire data assigned to activity topic k

Table 4.1: Symbols Used

above as K and the sensor-segment frequency matrix from the "bag of object observations" are required. In this case, the "bag of objects observations" constructed from dataset is analogous to the corpus of documents which contain words. The LDA iterative process simply as illustrated with the schema 4.6 (see Table 4.1 for key symbols). The Gibbs sampling [50] is applied so that P($X|Z, \beta$) and P($Z|\alpha$) depend on Φ and Θ respectively and used to derive them.

$$\begin{bmatrix} d \\ ... \\ x \end{bmatrix} \rightarrow Initialize \rightarrow \begin{bmatrix} z_0 \\ ... \\ x \end{bmatrix} \rightarrow P(z_i|z_{-i}, d, w) \rightarrow \begin{bmatrix} z_i \\ ... \\ x \end{bmatrix} \rightarrow \Phi, \Theta \tag{4.6}$$

To sample from P($Z|X$) the joint distribution is expressed as:

$$P(Z, X|\alpha, \beta) = P(X|Z, \beta)P(Z|\alpha) \tag{4.7}$$

$$P(X|Z, \beta) = \int P(X|Z, \Phi)P(\Phi|\beta)d\Phi \tag{4.8}$$

$P(\Phi|\beta)$ has a dirichlet distribution as:

$$\begin{aligned} P(\Phi|\beta) &= \prod_{k-1}^{K} P(\phi_k|\beta) \\ &= \prod_{k=1}^{K} \prod_{u=1}^{U} \frac{1}{B(\beta)} \phi_{k,u}^{\beta_u - 1} \end{aligned} \tag{4.9}$$

But P($X|Z,\Phi$) has a multinomial distribution given as:

$$\begin{aligned} P(X|Z,\Phi) &= \prod_{i=1}^{N} \phi_{z_i,x_i} \\ &= \prod_{k=1}^{K} \prod_{u=1}^{U} \phi_{k,u}^{\Psi_{k,u}} \end{aligned} \tag{4.10}$$

Ψ is the activity topic number K x the unique objects matrix. From the expression above $\Psi_{k,u}$ is the number of times that an activity topic is assigned to an object. With Φ in 5.7 and 5.8, 5.6 is resolved to be:

$$P(X|Z,\beta) = \int \prod_{k=1}^{K} \prod_{u=1}^{U} \frac{1}{B(\beta)} \phi_{k,u}^{\Psi_{k,u}+\beta_u-1} d\phi_k \tag{4.11}$$

Simplifying and integrating out, the next two expressions will be:

$$P(X|Z,\beta) = \prod_{k=1}^{K} \left(\int \frac{1}{B(\beta)} \prod_{u=1}^{U} \phi_{k,u}^{\Psi_{k,u}+\beta_u-1} d\phi_k \right) \tag{4.12}$$

$$P(X|Z,\beta) = \prod_{k=1}^{K} B \frac{(\Psi_k + \beta)}{B(\beta)} \tag{4.13}$$

Ψ_k is the kth row in the matrix Ψ. Recall that P($Z|\alpha$) is dependent of Θ, so that it becomes:

$$\begin{aligned} P(\Theta|\alpha) &= \prod_{d=1}^{D} P(\theta|\alpha) \\ &= \prod_{d=1}^{D} \frac{1}{B(\alpha)} \prod_{k=1}^{K} \theta_{d,\alpha_k - 1} \end{aligned} \tag{4.14}$$

$$\begin{aligned} P(Z|\Theta) &= \prod_{i=1}^{N} \theta_{d_i,z_1} \\ &= \prod_{d=1}^{D} \prod_{k=1}^{K} \theta_{d,k}^{\Omega_{d,k}+\alpha_k-1} \end{aligned} \tag{4.15}$$

$$
\begin{aligned}
P(Z|\alpha) &= \int P(Z|\Theta)P(\Theta|\alpha)d\Theta \\
&= \prod_{d=1}^{D}\left(\int \frac{1}{B(\alpha}\prod_{k=1}^{K}\theta_{d,k}^{\Omega_{d,k}+\alpha_k-1}d\theta_d\right) \\
&= \prod_{d=1}^{D}\frac{B(\Omega_d+\alpha)}{B(\alpha)}
\end{aligned}
\tag{4.16}
$$

$\Omega_{d,k}$ is the number of times activity topic k is assigned to the object in the object segment d. The joint distribution using eq 5.5 becomes eq 5.15 using 5.11 and 5.14.

$$
\begin{aligned}
P(Z,X|\alpha,\beta) =& P(X|Z,\beta)P(Z|\alpha) \\
&= \prod_{k=1}^{K}\frac{B(\Psi_k+\beta)}{B(\beta)}.\prod_{d=1}^{D}\frac{B(\Omega_d+\alpha)}{B(\alpha)}
\end{aligned}
\tag{4.17}
$$

For each object, it determines the estimate of the probability of assigning it to a topic given the assignment of the other words in the entire data set of objects. It then calculated by:

$$
P(Z^{\neg i},X^{\neg i}|\alpha,\beta) = \prod_{k=1}^{K}\frac{B(\Psi_k^{\neg i}+\beta)}{B(\beta)}.\prod_{d=1}^{D}\frac{B(\Omega_d^{\neg i}+\beta)}{B(\beta)}
\tag{4.18}
$$

The elements of matrices Φ = $\{\phi_{k,u}\}$ and Θ = $\{\theta_{d,k}\}$ containing the specific activity-object distributions and the specific segment-activity topic, are determined by:

$$
\phi_{k,u} = \prod_{k=1}^{K}\frac{B(\Psi_k^{\neg i}+\beta)}{B(\beta)}
\tag{4.19}
$$

$$
\theta_{d,k} = \prod_{d=1}^{D}\frac{B(\Omega_d^{\neg i}+\beta)}{B(\beta)}
\tag{4.20}
$$

$$
\begin{aligned}
\phi_{k,u} &= P(x=u|z=k) \\
\phi_k &= P(x|z=k) \\
\theta_{d,k} &= P(z=k|d) \\
\theta_k &= P(z|d)
\end{aligned}
\tag{4.21}
$$

It then conversely applies the LDA assumptions to that of the segments of objects in the dataset, that latent activity topics would have associations with the features of objects data in the partitioned segments of the "bag of object observations". If the number of suggested clusters are used as topic numbers from the silhouette method above in section 4.3.1, then the object use distributions specific to the routine activities can be determined result similar to the matrix as schema 4.22. The unique objects and activities are expressed as $x_1...x_N$ and $k_1...K$ respectively. The process would have discovered the activity-object use. The process ends with the LDA process assigning objects to specific activity topics.

$$P(x|z) = \begin{bmatrix} \{x_{1k1}..., x_{Nk1}\} & number & of & unique & objects & are & assigned & to & k1 \\ \{x_{1k2}..., x_{Nk2}\} & number & of & unique & objects & are & assigned & to & k2 \\ ...... & & & & .... & & & & \\ \{x_{1K}..., x_{NK}\} & number & of & unique & objects & are & assigned & to & K \end{bmatrix} \tag{4.22}$$

4.4 Context Descriptors for Routine Activities

The knowledge base of a knowledge-driven activity recognition technique is dependent on a set of activity and object concepts carefully encoded ontologically to represent the activity descriptions. The knowledge representations are such that for a particular activity as a concept, there are object concepts which are used to describe the activity. In essence, the activity is specified by linking and associating it with objects as context descriptors. The activity concepts are structured in some cases hierarchically allowing more and general contextual properties in addition to the main associating object concepts to encode the activity descriptions. Ideally, activities are performed generating sensor events resulting from object interactions and object usage in the home environment. Understanding the activities and how they are performed relies on the knowledge breakdown of the respective object usage for the specific routine activities. This thesis bases this knowledge on the activity-object use discovery described in the previous sections. The activity-object use discovery process provides the knowledge of objects used for the routine activities. The aim is to use the activity-object use discovery to provide the conceptual model for annotating the routine activities with their context descriptors. So the object usage discovered then becomes the context descriptors for the activity concepts in the activity ontology. The context descriptors specifications for the routine activities provides the link and relationship between the objects and the activities. This link and relationship are carried onto the ontology layer to help provide class relationship and property assertion among the domain concepts. This process of context description utilises the object assignments to a specific activity topic. It uses the number of times an object has been assigned to an activity and applies a threshold to imply the object as a context describing the activity conveniently.If $x_{1ki}..., x_{Nki}$ are the number of times unique objects has been assigned to an activity topic k_i, then it becomes a context descriptor if x_i for the activity topic k_i is greater than a threshold value. The idea is that for an object to be a context describing an activity topic, it must have been assigned to an activity topic by a number of times greater than the threshold μ. μ is determined by computing the mean M of the number of objects assignment to K topics and standard deviation SD of the number of times an object has been allocated to an activity topic (see expression 4.23). The threshold values vary as $\mu_1...\mu_K$ for the unique activities $k_1...K$ since the unique objects $x_{1ki}..., x_{Nki}$ have different numbers of occurrences in the dataset. Finally, the context descriptors for the specific routine activities are achieved using the Algorithm 1 with dependency on the activity-object distributions from the LDA and μ.

$$\mu = M + SD \tag{4.23}$$

Algorithm 1: Algorithm for Context Descriptors of Activities.

Input: Observed Sensors in Partition of segment-sensor frequency matrix D = $d_1...d_D$, Probability distribution P($x|z$), μ ;
Result: Most plausible Activity z_i, Set of z_i descriptors.
Begin:;
while *data stream is active* **do**
    Extract observed objects from segment d_i, X = $x_1...x_n$ **for each** $d_i \in D$;
    Perform LDA topic model P($x|z$) ;
    for each P($x_i|z_i$) $\in$ P($x|z$) do;
    Y = $\{x_j|\forall x_j \in X\ \}$;
    Z = $\{z_j|\forall z_j \in Z\ \}$;
    if ($\exists\ x_i \in Y$, $z_i \in Z$) $> \mu$ **then**
        z_i most plausible activity;
        $x_i \rightarrow x_i \cup$ {Set of z_i descriptors} ;
        for all Repeat process for next d.
    end
end

These context descriptors for the routine activities become the knowledge acquired from the object usage and the needed information of object and activity concepts that will be encoded onto the activity ontology.

4.5 Conclusion

The Chapter presented the proposed hybrid approach and knowledge acquisition of object use and context description for activity ontology. It described the method for acquiring knowledge of object usage for specific routine activities in the home environment. It also presented the process used to automatically determine the number of activity topics, "bag of object observations" and the generative topic models. As a means to represent develop and represent activity and object concepts in the development of the knowledge base for activity recognition, the need for context descriptors was highlighted. In addition, this chapter presented how to use these three steps to determine the context descriptors for routine activities. The algorithm for context description was introduced to form the basis for knowledge representation and the activity ontology development in the next chapter.

Chapter 5

ACTIVITY ONTOLOGY MODELLING FOR ACTIVITY RECOGNITION.

The previous chapter considered knowledge acquisition of object use for activity ontology. In particular, it focused on how to determine the likely object usage for routine activities using topic models. As discussed in Chapter 4, this process of knowledge acquisition provides the context descriptions of object concepts to be linked or associated with the activity concepts needed to be modelled in the proposed activity ontology. Unlike the hybrid approach proposed in this thesis, traditional and existing knowledge-driven activity recognition techniques develop or model activity ontologies of concepts (activities and objects) based on the general everyday knowledge or assumptions of object usage for activities which most times are not fitting or cannot be adapted to home setting and users. This chapter focuses on developing and modelling the activity ontology from object and activity concepts. The resulting activity ontology of concepts relies on the context descriptors from Chapter 4. Sections 5.1 and 5.2 discusses the ontology development from concepts and contexts. Temporal representations of the concepts are presented in 5.3. Sections 5.4 and 5.5 presents ontology activity model and modelling activities as class concepts respectively. Activity recognition by object use query is presented in section 5.6 and finally concludes the chapter.

5.1 Modelling Activities of Daily Living (ADL) Concepts

The process of modelling ontology begins with the identification of the key concepts and the relationships between them. The concepts in any domain are defined by the information space produced by a set of known procedures [112]. In ADL, the collection of devices in the

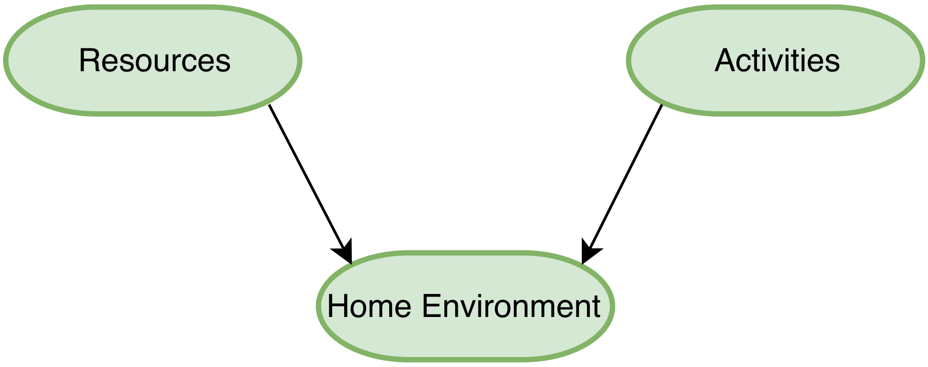


Figure 5.1: Key concepts identified in the home environment.

home environment and the activities resulting from the interaction of these devices form the set of key concepts needed to build an activity ontology. For the sake of their significance in ADL, activities and resources (devices) are regarded as key ontology concepts. The devices or the resources serve as the common basis for which activities are carried out hence defining the relationship or the shared semantics which characterises these two concepts. In this information space, activity situations are produced by the use of the resources in the home environment as illustrated in Figure 5.1.

Activities that are performed are identified and *Resources* are used in the process. The formal modelling process would involve the specification of the *Activities* and *Resources* as classes in the activity ontology domain to encode the information they represent. A common feature of the concepts in any domain are the relationships between them. This relationship establishes the links which facilitate the connections between the key concepts. Depending on the domain, relationship concepts carry evidence of commonly shared semantic properties which may convey similarities or differences between the key concepts. Typical relationship concepts include equal, subsume, overlap, adjacent and disjoint [29, 28, 128]. For example, if in a home environment, *Make Food*, *Make Breakfast*, *Make Dinner*, *Make Drink*, *Use Shower*, *Use Toilet*, *Sleeping* and *Go Out* are *Activities* concepts set for specification in the activity ontology, the following relationship concepts could be specified to encode the relationship information as:

- *Make Breakfast* is equal to *Make Food* exhibiting reflexive, symmetric and transitive properties.
- *Make Food* subsumes *Make Breakfast* and *Make Dinner* exhibiting reflexive, anti-symmetric and transitive properties.
- *Make Food* overlaps *Make Breakfast*, *Make Dinner* and *Make Drink* exhibiting irreflexive and symmetric properties.

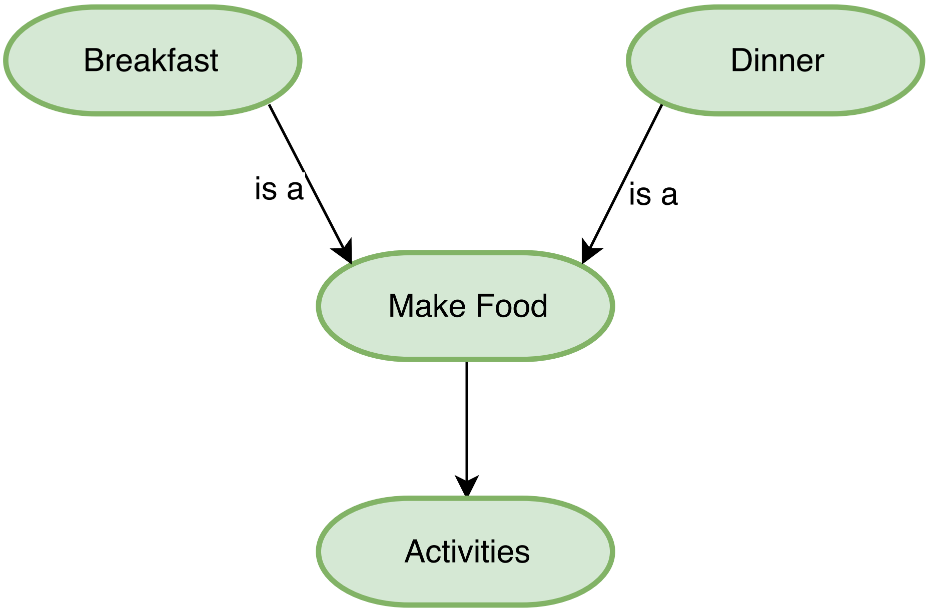


Figure 5.2: Make Food subsuming Breakfast and Dinner

- *Use Shower* is adjacent to *Use Toilet* exhibiting irreflexive and symmetric properties.
- *Make Drink* is disjoint with *Make Breakfast* and *Make Dinner* exhibiting irreflexive and symmetric properties.

The process of ontology modelling continues with the identification of other concepts which may be defined as members of the key concepts. They are then specified in a hierarchical structure to represent relationships in terms of commonly shared semantic characteristics or properties. *Make Food* from the example of *Activities* concepts above could be specified to have *Make Breakfast* and *Make Dinner* as member concepts as illustrated in Figure 5.2. *Make Food* the subsumes *Make Breakfast* and *Make Dinner* so that *Make Food* becomes a superclass of with respect to *Make Breakfast* and *Make Dinner*. *Make Breakfast* and *Make Dinner* the inherit properties of *Make Food* exhibiting reflexive, symmetric and transitive properties. *Make Breakfast* and *Make Dinner* may have distinct properties which makes them disjoint and irreflexive.

5.2 Modelling ADL Concepts with Contexts

To this point, the concepts in the home environment based on the abstract information and relationship they can exhibit in this information space have been described. But these, in reality, restricts and excludes detail information about the situational contexts which may convey details which characterises the dimensional space in context. By these, reference is made to any contextual information which can be used to characterise the concepts and contexts encoded with them to provide a true reflection of the concept in the information space. In the words of Yau and Karim [144], contexts are "any information acquired from a system or an environment". According to Dey et al. [36], situational contexts could be adapted from who's (the user's identity), where's (the user's location), when's (the time of activity), and what's (the user's activity) of concepts to determine why the situation is occurring. The process of modelling concepts with contextual information involves establishing contexts which reflect the reality of the situation. They are specified appropriately by encoding them as properties of the concepts which may further be used to enforce relationships between concepts. For example, *Microwave* and *Fridge* are *Resources* located in the *Kitchen*. They are used to for the activity *Make Breakfast* in the morning. Obvious contexts here are location, time and the usage requirement of the *Microwave* and *Fridge* concepts to perform the activity concept *Make Breakfast*. Enforcing these contexts requires that they are encoded as predicate properties so that they can be used to represent the relationship between the concepts as illustrated in Figure 5.3. In ontology, predicate properties could be expressed as object properties or data properties to accommodate contextual information to be encoded. In the example, above `locatedIn` and `hasUse` are encoded as object properties corresponding to the location and the usage requirement contexts, whilst `hasTime` has been encoded as a data property to corresponding to time as will be further discussed in the subsection below.

5.2.1 Activity Concept with Context

Considering the initial discussion on activity contexts in section 5.2, activities as events are performed dependent on the resources in the home environment. So, the process of modelling the activities as ontology concepts requires that the specific resources as concepts are encoded as vocabularies to represent and reflect how the activity events are performed. In addition to these resources concepts, contextual information can be specified to reflect the situation in the home environment. This thesis, the conceptual activity model with context information as illustrated in Figure 5.4. An *Activity* is represented as an instance of an ADL performed in the home environment similar to the *Make Breakfast* example above. Depending on the type and description of the *activity*, it could be a super-class or sub class of the instantiated ADL. This super and sub-class relationship defines the hierarchy of relationship of the *Activity* concepts. The super-class *Activity* subsumes the subclass *Activities* such

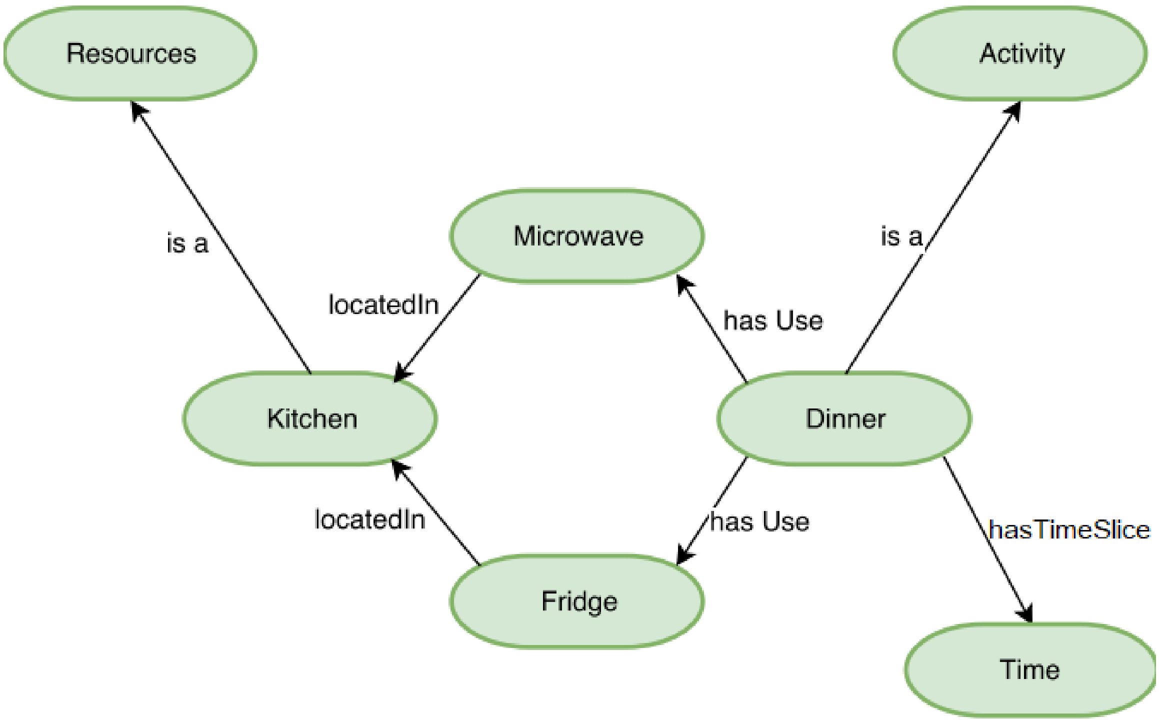


Figure 5.3: Modelling concepts with contexts

that the subclasses inherits all the properties of the super-class. The sub-classes may in addition have distinguishing properties which disjoints them just as in the example of *Make Breakfast* and *Make Dinner* which are subclasses of *Make Food* considered in sections above. The subclasses then exhibit irreflexive properties to disjoint them as distinct activities though inheriting the properties of *Make Food*. An individual referred to as an *Actor* performs the instantiated activity of the ADL. This *Actor* may be specified as an individual in the ontology to have its own specific properties. The properties `hasDescription` and `hasLocation` encodes the context information of the *Description* and *Location* of where the *Activity* is performed. The *Activity* requires the use of objects in the home environment to perform them. These *Objects* are of different types and are placed in the different locations of the home environment. In dense sensing, these *Objects* or *Resources* usually have sensors attached to them. Sensor observations and reports indicate the object usage or resource interactions which are translated as activity events. The *Objects* used for specific *Activities* are encoded as *Resources* for the respective *Activity* concepts with the property `hasUse`. The resources as ontology concepts are further modelled with the properties `hasLocation`, `hasType` and `hasSensorAttached` to represent *Location*, *Type* and *Sensor* respectively. The *Location* of an *activity* in most cases is the same location of the *Resources* used in performing it. Additionally, the temporal information of the *Activity* is represented using the property `hasTime` to encode *Time* as a concept. Using

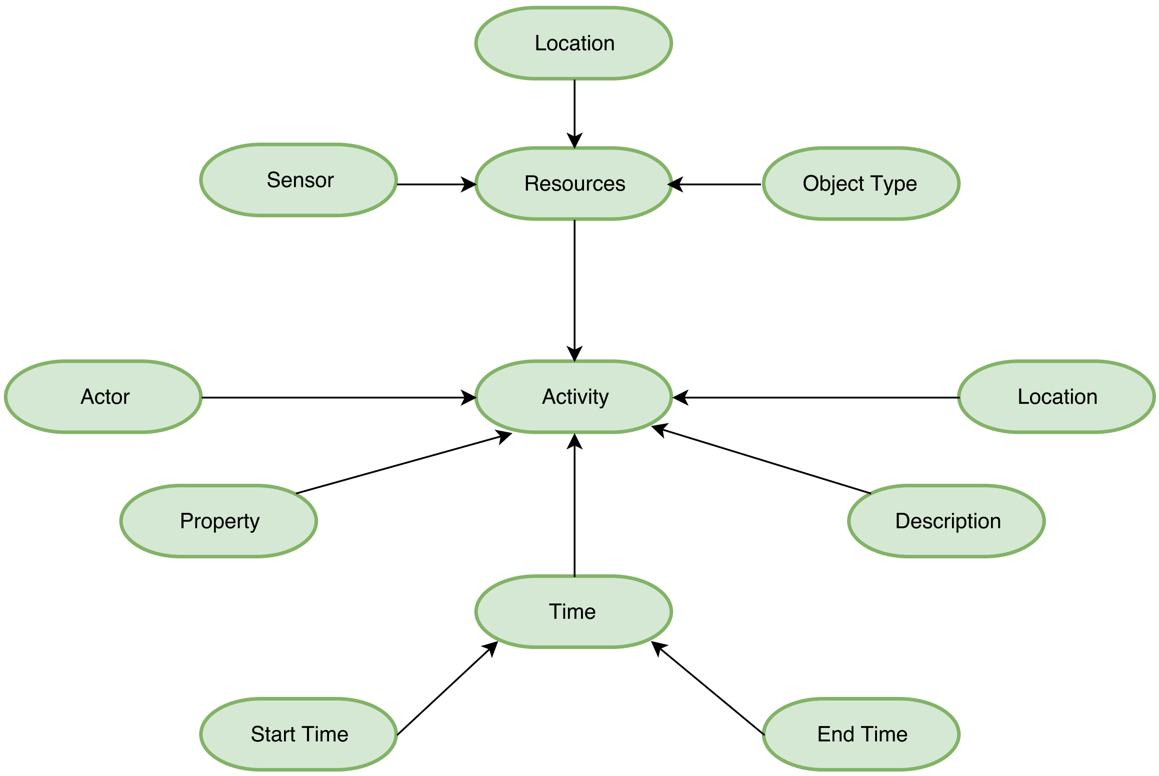


Figure 5.4: A conceptual activity model with contexts

`hasStartTime` and `hasEndtime` further gives more granular temporal information of the instantiated *Activity*. More details on temporal representation are presented in the next section.

5.3 Temporal Representation

To model ADL concepts satisfactorily to reflect activities as they are performed, temporal properties are required. However, temporal properties are difficult to be encoded because OWL specifications allows only unary and binary relationships between concepts [15]. With OWL-Time, it is now possible to encode temporal concepts and properties. As a way demonstrating the representation of temporal properties and relation, Batsakis and Petrakis [15] used 4D-fluents approach. According to Welty and Fikes [139], fluents are relational concepts holding within certain intervals. They also are relational properties used to define time instants and time interval as part of the temporal attributes of concepts. In practical terms, fluent properties holds between two time instants which can be implemented to represent start time and end time of either an activity or the use of an object. In this thesis, we follow the 4D-fluents approach to encode temporal properties to the activity ontology model. Representing 4D-fluents requires the use of the core vocabularies time slices and fluents. The time slices represents specific time instants for example the instant an object is used or an activity event occurred. A time interval can also be encoded to represent a range of times characterizing the time instances. The intervals then aggregates all the time instants as time slices. Additionally, *Time slice* and *Time intervals* can be included as class concepts with the relational properties `hasTimeSlice` and `hasTimeInterval` to extend the activity concepts with contexts. The *Time interval* encodes a range of time intervals within the day path for example a day path can be encoded to have four 6 hourly intervals 00:00 to 06:00, 06:00 to 12:00 etc. The *Time slice* then encodes time instants within these Time intervals to capture specific temporal attributes of events in the home environment.

5.4 Ontology Activity Model

The activity ontology is developed following the conceptual activity model with contexts. The process start with a generic activity ontology as illustrated in Figure 5.5. The *ADL* class is the parent of all activity class concepts. The *Resource* class is the parent of all artefacts or object concepts in the home environment. The *Resource* class concept has as *Doors*, *Fixtures*, *Furniture* and *Devices* as subclasses so as to allow the different objects used in the home environment to be encoded. Other class concepts included in the generic ontology implementation includes *Symbolic Location*, *Sensors* and *Persons* as exemplified as contexts in the conceptual activity model. *TimeSlice* and *TimeInterval* has been included to encode temporal properties. Contextual relationships between the class concepts has been enforced through the different specified properties. The respective properties are specified to have domain class concept to which it intersects and a range of class concept to which it intersects. As set of properties used for the activity ontology are provided in Table 5.1. With the specification of the properties given their domain and ranges a generically modelled simple activity would be defined as:

$$SimpleActivity \sqsubseteq ADL \sqcap \exists property1.Range1 \sqcap \exists property2.Range2. \tag{5.1}$$

Where *Ranges1* and *Range2* are resource concepts used as objects to perform the simple activity. The property has been specified such that the domain intersects the *Simple activity* or in broader terms *ADL* and ranges on to the objects as they are used to perform

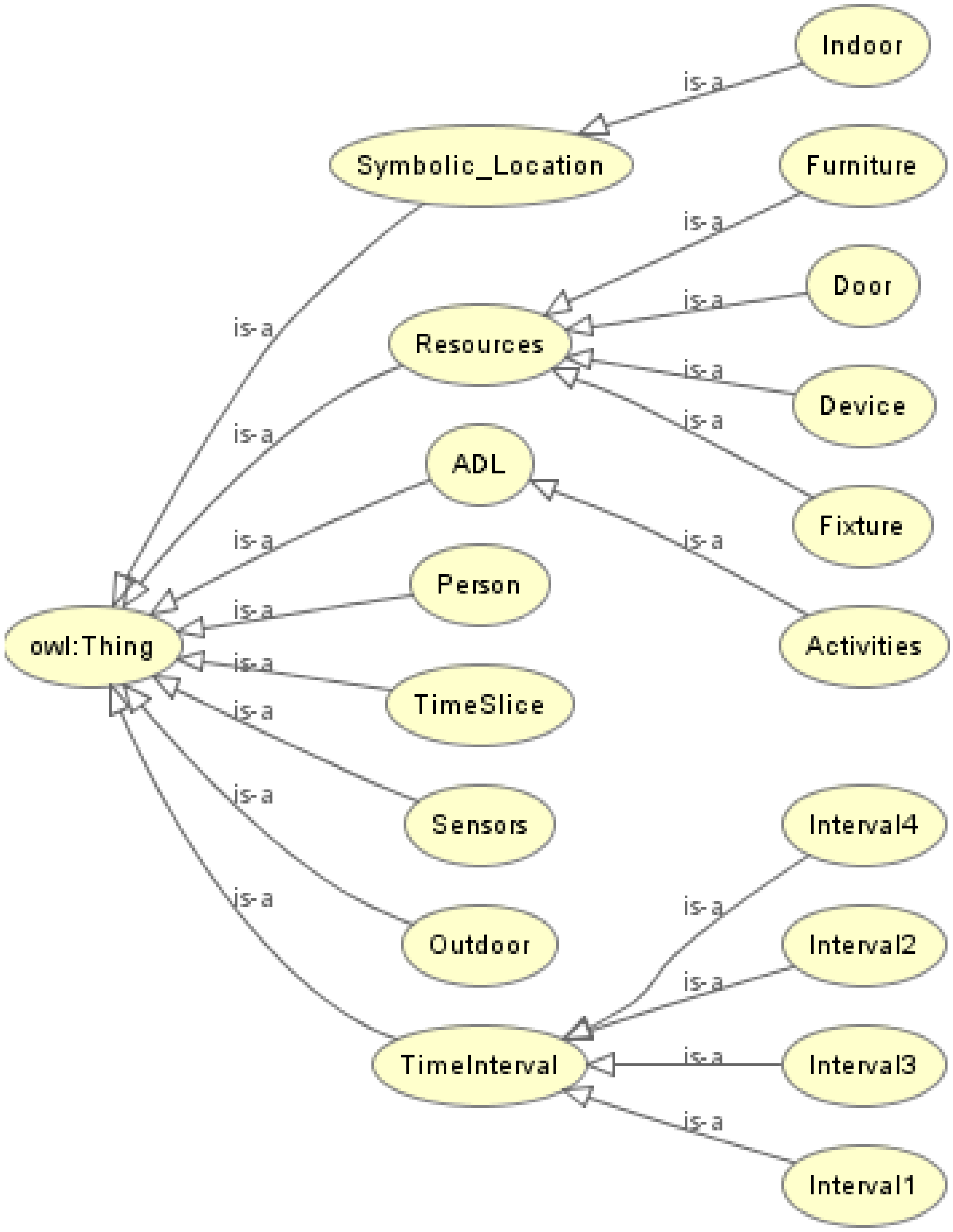


Figure 5.5: A generic activity ontology model

Property	Domain	Range
`hasUse`	Activities	Resources
`isTaggedto`	Sensors	Resources
`hasActor`	Activities	Person
`hasLocation`	Activities	SymbolicLocation
`tsTimeSliceOf`	TimeSlice	Activities
`tsTimeInterval`	TimeSlice	TimeInterval

Table 5.1: Basic Properties for the Activity ontology

the *Simple Activity*.
While these are at the generic level of development, it is pertinent that the activity ontology is developed to reflect the reality of how activities are carried out in the home environment. This means the activity ontology designed and developed for recognition of activity situations should encode sensor and object data output to describe activity events. To achieve this, the activity ontology proposed in this thesis is composed of a Terminological Box (TBox) and an Assertional Box (ABox) [110, 122]. The ontology modelling process populates the TBox with the sensors, objects usage for specific activities as context descriptors and the activities. For example, if *Microwave* is example of a *Resource*, it can be modelled as a subclass of *Resource*. With this, *Resource* and *Microwave* are ontology terms or concepts making up the TBox. Further, *Microwave* can be modelled as an individual resource or an instance of the class *Resource* which then makes up the ABox. The ABox can also populated by including the instances of the states of sensors and objects as an extension of the sensors and objects classes to reflect their use and further describe in reality activity situations. With this, sensor outputs from object interactions can be integrated into the activity ontology to recognise an ongoing activity. El^{++} being a lightweight description logic [12] which supports quantification and data types are used to describe possible sensor and object states in the Protégé [1] editing environment.

5.4.1 Modelling Sensors and Objects

In dense sensing, the interactions of the objects in the home environment produce sensor outputs. In some cases, these outputs may be Boolean and discrete values which determine the state of use of the object to which the sensor is attached. Modelling activity ontology to recognise activities from sensor data requires extending the sensor and object classes to include the state of the sensors which reflects possible sensor data outputs from object usage. With this, sensor data resulting from object interactions can be used to infer ongoing

[1] http://protege.stanford.edu/

activities. A binary sensor state can be ON or OFF, then the concept as represented in expression (5.2) would encode the possible states for a Passive Infra-Red (PIR) sensor attached to a *Microwave*.

$$\begin{aligned} &PIR \sqsubseteq Sensor \\ &Microwave\_On \sqsubseteq PIR \\ &Microwave\_Off \sqsubseteq PIR \end{aligned} \tag{5.2}$$

With the `hasState` data property, discrete values returned as outputs of actual sensor states can be captured in the ontology through data types like string or integer as express in (5.3) below.

$$Microwave\_On \sqsubseteq Microwave \sqcap \exists \texttt{hasState}.(=, On) \tag{5.3}$$

Further, an individual can asserted to instantiate the state of the sensor in the activity ontology. If an activity *Breakfast* results from the use of the *Microwave*, then the expression (5.4) would encode the activity through the object property concept `hasUse`. These assertions are then added to the ABox.

$$Breakfast \sqsubseteq ADL \sqcap \exists \texttt{hasUse}.Microwave\_On \tag{5.4}$$

Every sensor output has specific times associated to them. In most cases, they are the start times and end times to denote the time interval when the sensor reading was observed. These temporal attributes and time instants associated sensor data output can be included e.g. `hasStartTime` data property to represent the instant when the sensor state was captured. The `hasStartTime` data property of a *Microwave* turned on at time t can be captured through data types like and expressed as (5.5)

$$Microwave\_On \sqsubseteq Microwave \sqcap \exists \texttt{hasStartTime}.(=, t) \tag{5.5}$$

The activity *Breakfast* resulting from the use of *Microwave* turned on at time *t* would be expressed as:

$$Breakfast \sqsubseteq ADL \sqcap \exists \texttt{hasUse}.\Big(Microwave\_On \sqcap \exists \texttt{hasStartTime}.(=, t)\Big) \tag{5.6}$$

`hasStartTime` and `hasEndTime` properties are included in the activity ontology to have domain intersection TimeInterval so that resulting activities are modelled reflecting the intervals of their occurrence. The inferential process through instance checking returns

Breakfast in the example above by checking the states of all the sensors and their asserted times and then it returns the most relevant results.

5.4.2 Modelling Activity Situations with Context Descriptors.

The previous section described modelling of a simple activity situation from a sensor with state On and observed at time t. But in reality, an activity situation can be a result of a sequence of sensors triggered by objects used. Modelling an activity situation resulting from a set of sensors requires asserting all the sensors and with their times to be encoded to represent the activity situation. If an activity situation *Breakfast* is the result of *Microwave_On* and *Fridge_On* at times *t1* and *t2* respectively, then *Breakfast* can be asserted with the properties `hasUse` and `hasStartTime` as `hasUse`(*Microwave_On*, `hasStartTime` *t1*), `hasUse`(*Fridge_On*, `hasStartTime` *t2*). The activity situation is therefore modelled as a list of the sensors and with their times ordered temporally. The example of *Breakfast* from *Microwave_On* and *Fridge_On* at *t1* and *t2* can then be encoded by the expression below.

$$\begin{aligned} Breakfast \quad & \sqsubseteq ADL \sqcap \exists \texttt{hasUse}.\Big((Microwave\_On \sqcap \exists \texttt{hasStartTime}.(=, t1)) \\ & \qquad \sqcap \exists (Fridge\_On \sqcap \exists \texttt{hasStartTime}.(=, t2))\Big) \end{aligned} \tag{5.7}$$

Typically, activity situations or activities in the home environment are a result of specific objects and resources usage. To model activity situations accurately, it is significantly important to extend the present activity ontology modelling to include specific resources and or objects used for routine activity situations. This can be achieved through activity-object discovery for context descriptors. Recall in Chapter 4 of this thesis discussed activity-object use discovery enabled by the LDA topic model. This is a layer in the proposed approach to determine the likely object use for the routine activity situations rather the dependence on generic and or everyday knowledge of object use. The context descriptors resulting from the activity-object use discovery component forms the resources and objects concept and knowledge to be modelled onto the activity ontology for the specific routine activities such that if:

Activities: The activity topics determined from the activity-objects use discovery and are annotated as activities analogous to the activity situations in the home environment. This represents a class collection all types of activities set as $z_1...z_k$.

Objects: These represents class collection of all objects in the home environment set as $x_1...x_n$.

Contextual Elements: They represent the context attributes and states of the observed objects associated to the activities and environmental conditions. They are set as $e_1...e_m$ entities describing the the states of the observed objects.

Recall P($x|z$) in equation 4.21 and the expression 4.22 computes the likely objects for each of the activity topics, then z_i from $z_1...z_k$ would have a subset of likely objects defined as $x_{1z}...x_{iz}$. The function f is the context descriptor of $x_{1z}...x_{iz}$ objects for z_i as the likely objects for this activity such that:

$$f: \quad z_i \rightarrow x_{1z}...x_{iz} \tag{5.8}$$

If the context description function is replaced with the object property function in ontology object property `hasUse`, then expression (5.8) becomes:

$$z_i \quad \texttt{hasUse} \quad x_{1z}...x_{iz} \tag{5.9}$$

Considering that objects in the home environment are sensor tagged with outputs as discussed in section 5.4.1, then expression (5.9) with temporal attributes will be expressed as:

$$z_i \quad \texttt{hasUse} \quad \{(x_{1z}\_On, \texttt{hasStartTime}t_{1z})...(x_{iz}\_On, \texttt{hasStartTime}t_{iz})\} \tag{5.10}$$

The context descriptors are then modelled as resources and objects class concepts accordingly and added to the ABox so that:

$$\begin{aligned} Z_i \quad \sqsubseteq ADL \sqcap \exists \texttt{hasUse}.\Big(((x_{1z}\_On \sqcap \exists \texttt{hasStartTime}.(=, t_{1z})) \\ ... \sqcap \exists (x_{iz}\_On \sqcap \exists \texttt{hasStartTime}.(=, t_{iz}))\Big) \end{aligned} \tag{5.11}$$

5.5 Modelling Activities as Class Concepts

The success of modelling activities does not only depend on the context descriptors but also on the patterns and situations of occurrence. Without considerations to activity-object use or context descriptors, some activities are traditionally performed at specific times of the day whilst others are not. This gives rise to the relevance of time in the modelling of activity concepts. The concepts *Time Interval* and *Time Slices* discussed in section 5.3 above becomes relevant. Recall that the *Time Slices* represents specific time instants, for example the instant an object is used or a sensor is triggered. The *Time Interval* represents the range of times characterizing the time instances. But an activity is a result of at least an object use or a sequence of sensor outputs. The activities are considered to be performed in *Time Interval* whilst object are used in time slices or instants. Then, the *Time Interval* activities are performed aggregates all the time instants as *Time Slices* in which the objects are used. The *Time Interval* can be modelled in ontology as part of the day path so that the activities are considered to be performed at particular times of the day. But some of these activities are constrained to specific *Time Intervals* making them "static activities". The other category of activities not constrained to be performed at specific times of the day are regarded as "dynamic activities". Further, as these activities are performed, they can generate a sequence of situations or patterns in which an activity may be interleaved, non interleaved or concurrent with other activities. These activity situations and patterns are independent of whether they are classed as static or dynamic activities. In the sections below considers how to model static and dynamics activities and then modelling activity situations.

5.5.1 Static and Dynamic Activities

In the home environment, activities are performed differently, in different ways and times within the 24 hour day. These activities can be static or dynamic, and in some cases, same and or similar objects may be used to perform these different activities. *Make Breakfast*, *Make Lunch*, *Make Dinner*, *Use Toilet* and *Showering* are used as examples of activities in the home environment to make the following analogies. *Make Breakfast*, *Make Lunch* and *Make Dinner* are examples of different activity concepts which can be performed with same or similar object interactions, but they are all performed at different times of the day. As sub class activities of the activity class *Make Food*, they differ with regards to their respective temporal properties. While they inherit all the properties of *Make Food* by subsumption, they can be easily confused in the recognition process if modelled in the ontology without consideration to their usual times of performance. Semantically, they are different, but they share the same properties by subsumption. Distinction can be achieved for them by the specification of the time intervals they are usually performed. On the other hand, activities like *Use Toilet* and *Showering* can be performed at any time of the day making the process of distinguishing them less dependent on their temporal properties . This thesis refers to activities that are known traditionally to be performed at specific times of the day as static activities whilst activities that can be performed at any time of the day as dynamic activities. The Figure 5.6 illustrates an example of static and dynamic activities. The static activities in this example is with reference to *Make Breakfast* which is specified to be

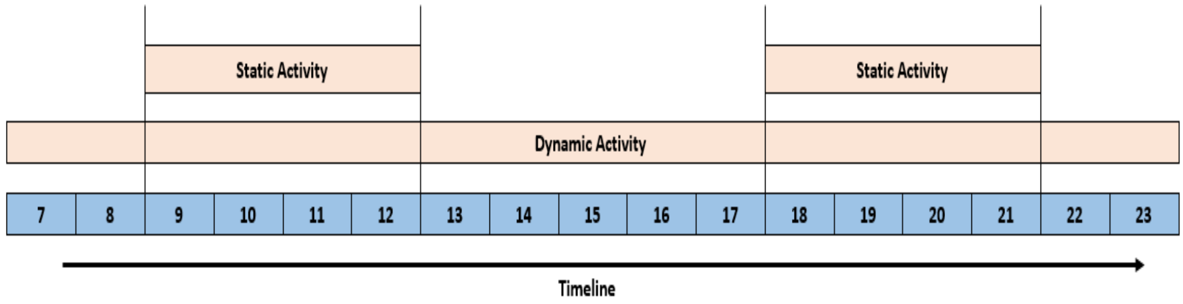


Figure 5.6: An example static and dynamic activities based on the analogy

possibly performed within the time interval 09.00 and 12.00 hours and *Make Dinner* specified for performance within 18.00 and 21.00 hours. Dynamic activities are not constrained within any time interval, so they have time interval ranging from 0.00 to 23.59 hour of the day. The extension of the ontology modelling of activity situations described in section 5.4.2 to include static and dynamic activities using the 4D-fluent approach [15], requiring the temporal class concepts *Timeslice* and *TimeInterval* to be specified using the relational properties `tsTimesliceOf` and `tsTimeIntervalOf` respectively (see Table 5.1). As illustrated in Figure 5.7, the time intervals *Interval1* and *Interval2* holds the temporal information of the time slices for the static and dynamic activities. An instance of a *TimeSlice* of an activity whether static or dynamic is linked by the property `tsTimeSliceOf` and property `tsTimeInterval` and then links this instance of the class *TimeSlice* with an instance of class *TimeInterval* which may be *Interval1* or *Interval2*.

Modelling a Static Activity: A static activity is modelled by requiring the specification of the *TimeSlice*, *TimeInterval* class concepts and with the context descriptors for that activity. The activity situations described above are extended so that the `hasUse` object property encodes the usage of the objects for the static activity by specifying the static activity as the domain class concept and ranges to all the object classes which describes the context descriptors. This is extended further with the temporal properties which requires `tsTimeInterval` to have domain *TimeSlice* and *Resources* and it ranges *TimeInterval* to capture specific time interval of the day through *Interval* (a sub class of *TimeInterval*). The time instants of the activities are captured through the `tsTimeSliceOf` with domain *TimeSlice* and *Resources* and it ranges to *TimeSlice*. The DL expression 5.12 encodes a static activity so that *Interval1* asserts the time interval of the day the static activity is performed using the object x_{1z}. *Interval1* with regards to Figure 5.6 can be modelled in the

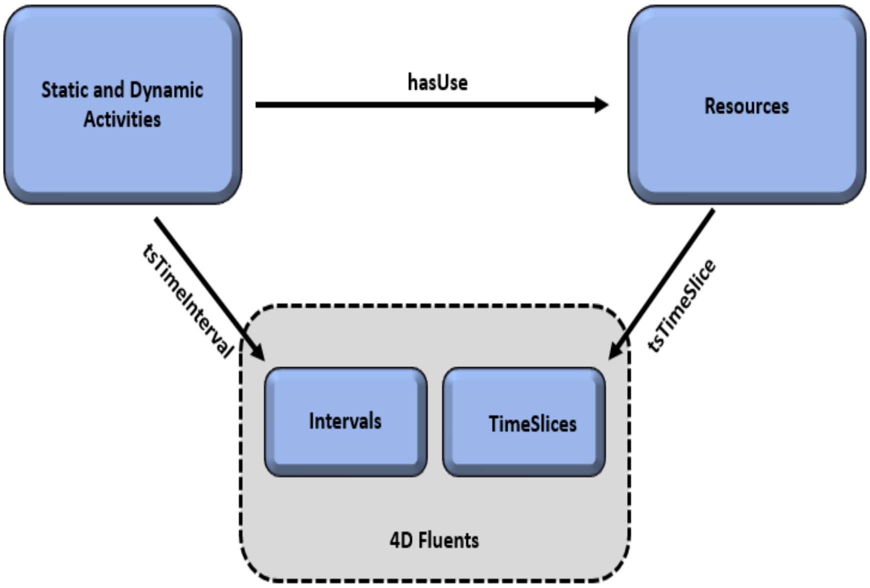


Figure 5.7: 4D-Fluents with Activities and Resources

activity ontology to have a range of times between 9 am and 12 noon using data property `starttime` 9.00 and `endtime` 12.00 in the format date time format. Activities performed within these ranges of time e.g. *Breakfast* can be modelled with this *Interval1* as an extension. An *Interval3* is also included ranging between 6 pm (18.00) and 9 pm (21.00) for static activities performed at this time of the day.

$$StaticActivity \sqsubseteq ADL \sqcap \exists \mathtt{hasUse}.\Big(x_{1_z}\_On \sqcap \exists \mathtt{tsTimeInterval}.Interval1\Big) \tag{5.12}$$

The expression 5.13 then asserts *Interval1* to cover the time instant t_{1_z} through the property `tsTimeSliceOf`.

$$Interval1 \sqsubseteq Interval \sqcap \exists \mathtt{tsTimeSliceOf}.\Big(TimeSlice \sqcap \exists \mathtt{hasStartTime}.(=, t_{1_z})\Big) \tag{5.13}$$

Modelling a Dynamic Activity: Similar to static activities, dynamic activities are modelled by requiring the specification of the *TimeSlice*, *TimeInterval* class concepts and with the context descriptors for that activity. An instance of a *TimeSlice* of a dynamic activity is linked by the property `tsTimeSliceOf` and property `tsTimeInterval` and then links this instance of the class *TimeSlice* with an instance of class *TimeInterval* which may be *Interval2*. *Interval2* ranges to cover the full 24 hour cycle of the day as asserted by 5.15 and modelled using data property `starttime` 0.00 and `endtime` 23.00 in the format date time format.

$$DynamicActivity \sqsubseteq ADL \sqcap \exists \texttt{hasUse}.\Big(x_{2z}\_On \sqcap \exists \texttt{tsTimeInterval}.Interval2\Big) \tag{5.14}$$

$$Interval2 \sqsubseteq Interval \sqcap \exists \texttt{tsTimeSliceOf}.\Big(TimeSlice \sqcap \exists \texttt{hasStartTime}.(=, t_{2z})\Big) \tag{5.15}$$

5.5.2 Modelling Fine Grain Activity Situations

In reality, activities are a result of multiple stepwise atomic tasks or sensor events. So when modelling the concepts in ontology, it is important to put this into consideration given that order of activity or precedence in some cases may lead to different types of activities. The role value expressed by $r1 \circ r2 \sqsubseteq r3$ holds the relationship of transitivity between $r1$, $r2$ and $r3$ [122, 11]. Therefore transitivity can be applied to a list ontology concepts in expressing precedence. Given this, the property relationship `hasLastObject` is introduced to specify precedence relationship between objects in use. With the `hasLastObject`, the ontology can be extended to include the order of list of context descriptors more especially in the of modelling fine grain activity situations. Considering a stepwise activity situation *Make Breakfast* from *GroceryCupboard_On*, *Microwave_On* and then *Fridge_On* afterwards. This would be expressed as:

$$\begin{aligned} MakeBreakfast \sqsubseteq\ & GroceryCupboard\_On \sqcap \exists \texttt{hasLastObject}.\\ & \Big(Microwave\_On \sqcap \exists \texttt{hasLastObject}.Fridge\_On\Big) \end{aligned} \tag{5.16}$$

The precedence property `hasLastObject` has been used encode the order as *Grocery Cupboard_On, Microwave_On* and *Fridge_On*. This adds granularity to the activity specification and refinement. This assertion can then be added to the ABox to describe *Make Breakfast* with the specifications of the order of object use. The power transitivity expressed in the example above can be fully utilised in the design and development of ontology to model fine grain activity to specification specific. If an activity A can be performed

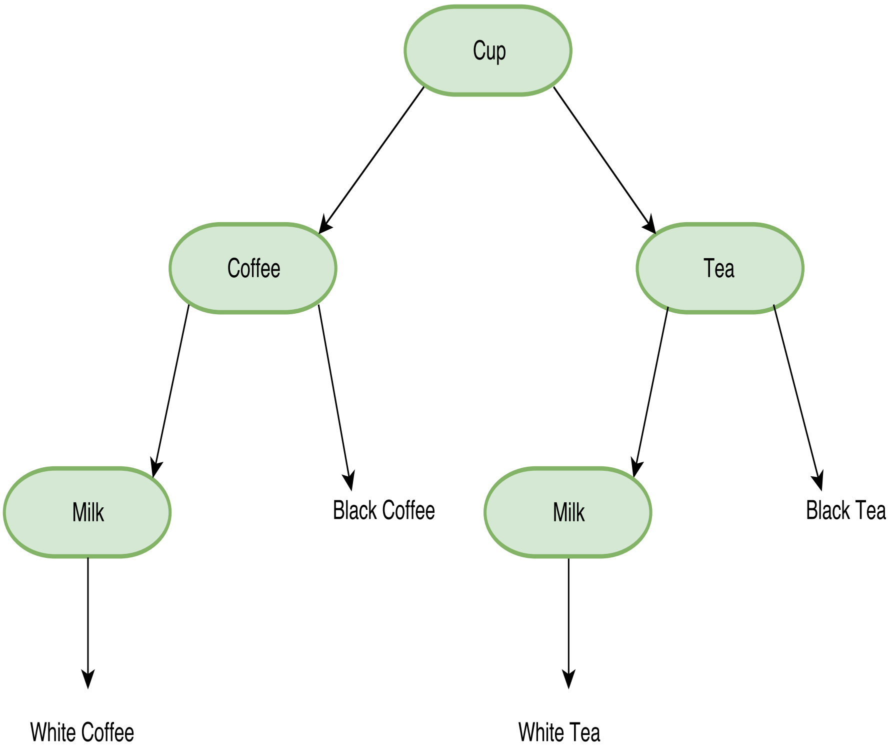


Figure 5.8: The four kitchen object and the possible resulting activity situations.

specifically with the objects $x1$, $x2$ and $x3$, the expression 5.17 can be followed to assert and recognise this activity situation given the order of precedence and object evolution.

$$\begin{aligned} x1 \quad \texttt{hasLastObject} \quad x2, \\ x2 \quad \texttt{hasLastObject} \quad x3 \quad then; \\ ActivityA = x1 \quad \texttt{hasLastObject}(x2 \quad \texttt{hasLastObject} \quad x3), \end{aligned} \tag{5.17}$$

The significance of this ordering in precedence is such that each of these objects as events can lead to multiple activity situations, the specifications of precedence refines the activity based on the context which describes the activity. The scenarios below explains further.

Scenario 1: Suppose a kitchen environment has the objects *Cup*, *Coffee*, *Milk* and *Tea*. *White Coffee*, *Black Coffee*, *Black Tea* and *White Tea* are four likely activity situations that could result from the combinations of these object based on preference. Basically from the tree illustrated in Figure 5.8 shows the four activity situations. The activity situations following object use order could be encoded as expressions 5.18, 5.19, 5.20 and 5.21. The Figures 5.9, 5.10, 5.11 and 5.12 illustrates possible orders of object use with *Cup* as first object use leading to *White Coffee*, *Black Coffee*, *White Tea* and *Black Tea* respectively.

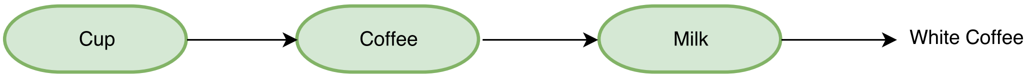


Figure 5.9: White Coffee

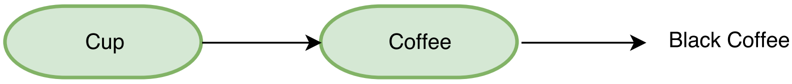


Figure 5.10: Black Coffee

White Coffee :Cup, Coffee and Milk.

$$WhiteCoffee \sqsubseteq Cup \sqcap \exists \texttt{hasLastObject}. \Big(Coffee \sqcap \exists \texttt{hasLastObject}.Milk\Big) \tag{5.18}$$

Black Coffee :Cup and Coffee.

$$BlackCoffee \sqsubseteq Cup \sqcap \exists \texttt{hasLastObject}.Coffee \tag{5.19}$$

White Tea :Cup, Tea and Milk.

$$WhiteTea \sqsubseteq Cup \sqcap \exists \texttt{hasLastObject}. \Big(Tea \sqcap \exists \texttt{hasLastObject}.Milk\Big) \tag{5.20}$$

Black Tea :Cup and Tea.

$$BlackTea \sqsubseteq Cup \sqcap \exists \texttt{hasLastObject}.Tea \tag{5.21}$$

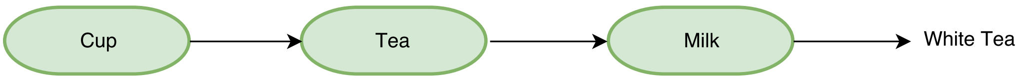


Figure 5.11: White Tea

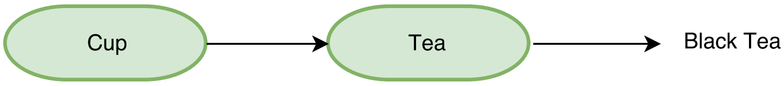


Figure 5.12: Black Tea

With order and precedence in a stepwise fashion, the four activities are bound if *Cup* is used first. Including *Coffee*, refines the situations to *White* and *Black Coffee*. A step further including *Milk* further refines the activity situation to *White Coffee*. With the transitivity property of `hasLastObject` the activity situations can be achieved so that all four activities depending on the path.

The modelling of activity situations proposed in this thesis are described by their contexts. The order according to precedence can be used to define and refine the activity situations especially when similar or same object interactions leads to different activity situations. Activity situations from the Kasteren Home environment are considered next[132].

Scenario 2: Consider the discovered context descriptors (following the context description process described in Chapter 4) for *Drink* includes *Fridge* and *Cups Cupboard*; *Breakfast* includes *Fridge*, *Cups Cupboard*, *Microwave* and *Toaster*; *Snack Fridge*, *Cups Cupboard* and *Microwave* (see Figure 5.13 for the activity situations tree). Although, *Breakfast* is an example static activity, *Snack* and *Drink* dynamic activity as earlier explained, they share similar or same objects for the situations. They are encoded them so that their context descriptors reflect the activity situations they could result to. The `hasUse` is used so that they are asserted as expressions 5.22, 5.23 and 5.24 and can be added to the ABox:

$$Drink \sqsubseteq ADL \sqcap \exists \texttt{hasUse}.(Fridge \sqcup \texttt{hasUse}.Cupboard) \tag{5.22}$$

$$Snack \sqsubseteq ADL \sqcap \exists \texttt{hasUse}.\Big(Fridge \sqcup Cupboard \sqcup Microwave\Big) \tag{5.23}$$

$$Breakfast \sqsubseteq ADL \sqcap \exists \texttt{hasUse}.\Big(Fridge \sqcup Cupboard \sqcup Microwave \sqcup Toaster\Big) \tag{5.24}$$

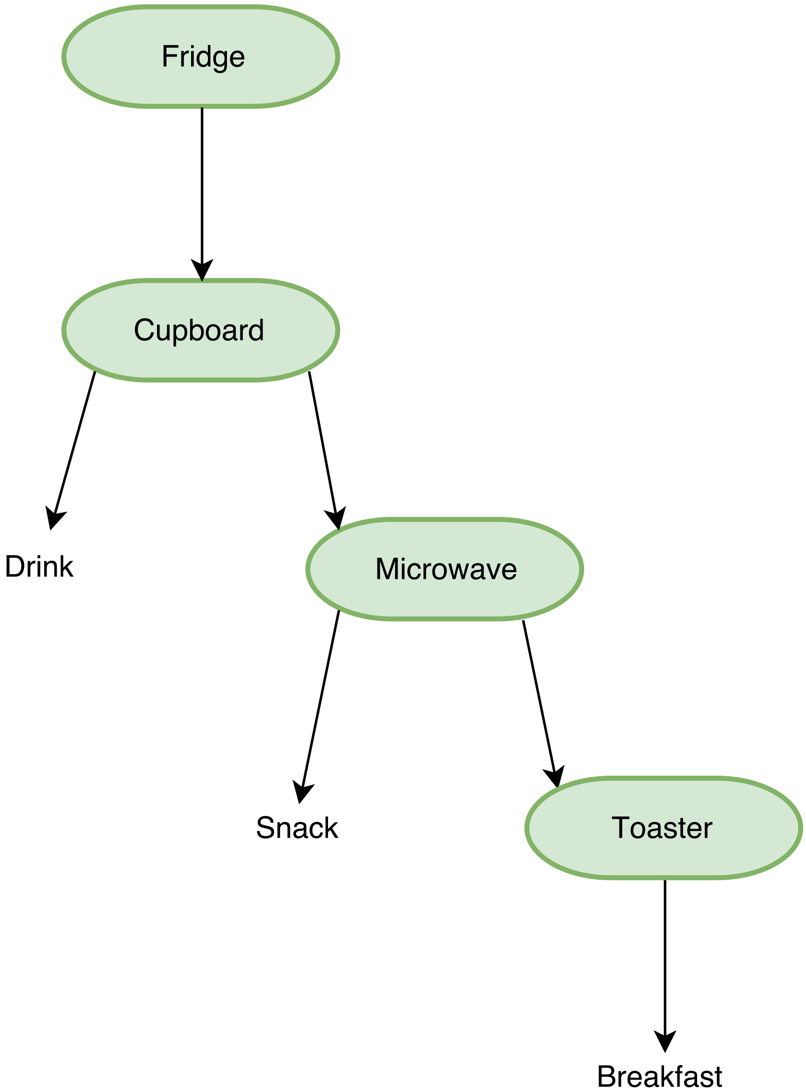


Figure 5.13: Tree structure of object contexts for activity

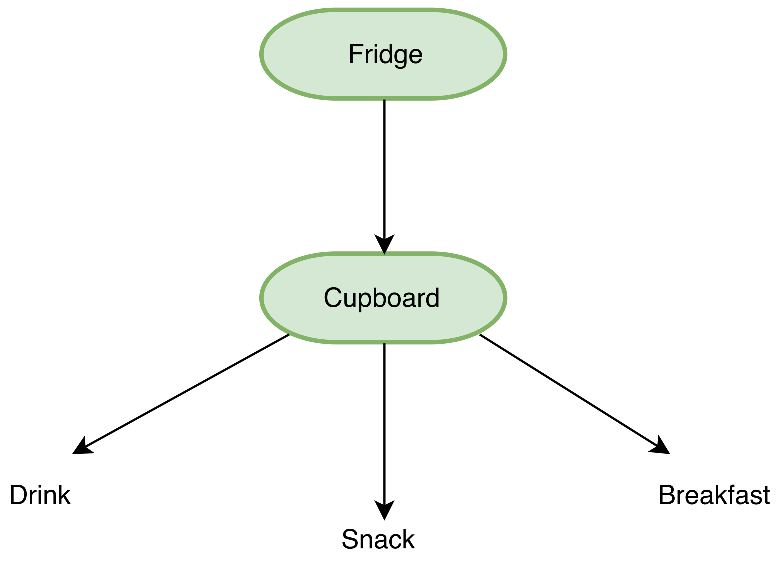


Figure 5.14: *Fridge* and *Cupboard* resulting to *Drink*, *Snack* and *Breakfast*

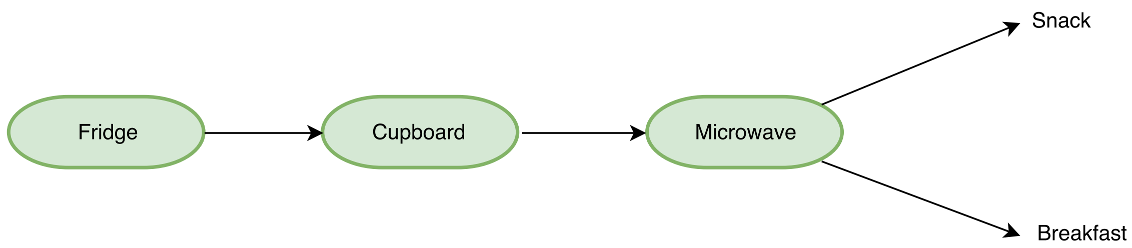


Figure 5.15: *Snack* and *Breakfast* from *Fridge*, *Cupboard* and *Microwave*

If all three activity situations are assumed to take place within the same time interval, `hasLastObject` can be used to further enhance the expressions to include precedence. With the order of the objects observed downward with regards to Figure 5.13, you notice that *Fridge* and *Cupboard* are common for all three activities. If *Fridge* is activated first as being used, *Drink*, *Breakfast* and *Snack* are the suggested activities asserted as expression 5.25. Activating *Fridge* and then *Cupboard* still results to the three activities asserted as eq 5.26 and illustrated as Figure 5.14.

$$Drink, Snack, Breakfast \sqsubseteq ADL \sqcap \exists Fridge \tag{5.25}$$

A step further activating *Microwave* excludes *Drink* so that the path now leads to *Snack* and *Breakfast* as expression 5.26 and illustrated as fig 5.15.

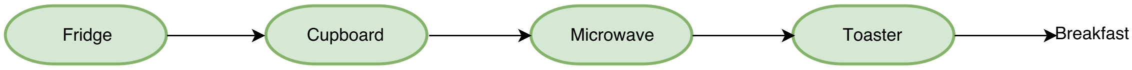


Figure 5.16: *Breakfast* from *Fridge*, *Cupboard*, *Microwave* and *Toaster*

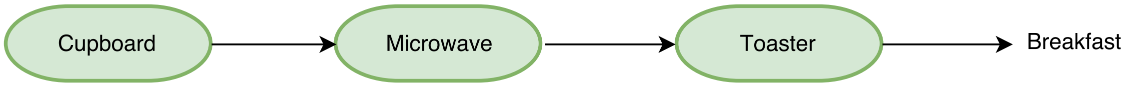


Figure 5.17: *Breakfast* with a changed of object activation

$$Snack, Breakfast \sqsubseteq ADL \sqcap \exists\ \texttt{hasLastObject}.\Big(Fridge \sqcap \exists\texttt{hasLastObject}.\\ (Cupboard \sqcap \exists\texttt{hasLastObject}.Microwave)\Big) \tag{5.26}$$

Further activating *Toaster* in addition, eliminates *Snack* from the path to result to *Breakfast* as in expression 5.27 and Figure 5.16.

$$Breakfast \sqsubseteq ADL \sqcap \exists\texttt{hasLastObject}.\Big(Fridge \sqcap \exists\texttt{hasLastObject}.\\ (Cupboard \sqcap \exists\texttt{hasLastObject}.(Microwave \sqcap \exists\texttt{hasLastObject}.Toaster))\Big) \tag{5.27}$$

This order of object observations may change to allow flexibility. However, transitivity rule still applies. If the order of observation becomes *Cupboard*, *Microwave* and *Toaster*. the progressive assertion becomes as expression 5.28 and Figure 5.17 for only *Breakfast* eliminating *Drink* and *Snack*:

$$Breakfast \sqsubseteq ADL \sqcap \exists hasLastObject.\Big(Cupboard \sqcap \exists\texttt{hasLastObject}.\\ (Microwave \sqcap \exists\texttt{hasLastObject}.Toaster)\Big) \tag{5.28}$$

Notice that the precedence property did not only incorporate the context descriptors for the activities but also is used to return finer grain of activity through the `hasLastObject` property. Modelling activity situations with the context descriptors and asserting them with transitivity property can create situations where an activity stops or ends creating an activity boundary to signal end or the beginning of an activity. Locations in the home environment creates location groups for objects and thus location basis for activity situations i.e *Toileting* in the *Toilet* using Toilet objects. Having a *Shower* in the *Bathroom* with *Bathroom* based object. If a user moves from the *Toilet* location to the *Bathroom* it could signal the end of an activity and the beginning of another. Introducing location grouping for objects could be used to create disimilarity amongst objects and then used to signal the end of an activity or the beginning. Disimilarity when applied with the transitivity property of `hasLastObject` enhances the process. Disimilarity of location object grouping can be determined using the Jaccard and Dice Coefficient such that if the similarity index of objects from different locations falls below a threshold then it can suggest an activity boundary.

At this point, β is introduced as the Jaccard similarity index [72] for A and B. If A is a set of Kitchen based objects x_1, x_2, x_3, and x_4 and B is a set of Toilet based objects x_5, x_6, and x_7 the β for location base objects in A and B is as:

$$\beta = \frac{|A \cap B|}{|A \cup B|} \tag{5.29}$$

$$\beta = \frac{|(x_1, x_2, x_3, x_4) \cap (x_5, x_6, x_7)\,|}{|(x_1, x_2, x_3, x_4) \cup (x_5, x_6, x_7)\,|} \tag{5.30}$$

β should be below a threshold for the set of A and B to be disimilar. Since objects as

context descriptors are likely resultant to activity situations, β can be used in sequence of objects observations to determine the persistence of an activity. Activity termination is discovered when the β value for consecutively observed objects falls below the threshold value for the location set they belong. The algorithm 2. implements the boundary detection for activities with dependency on β. Notice that the precedence of object observations is also enabled in this case by the use of `hasLastObject` property since they are consecutively observed. A third scenario is considered applying location based β similarity index.

Algorithm 2: Algorithm for Activities Boundary Detection.

Input: Observed Sensors X = $\{x_1,...x_n\}$ in Partition of Sensor Segment S = $\{s_1...s_n\}$, ADL ontology (ADL)
Result: Activity A, Activity Boundary.
Begin:;
while *data stream is active* **do**
  Extract observed object $x_i \in X$ = $\{x_1...\ x_n\}$, from $s_i \in s_1...s_n$ **for each** $s_i \in S$. ;
  Create Activities $Z \equiv Z(x_1)\sqcup...Z(x_n)$;
  for each $x \in A$ do;
  Map x_{i1} to an Activity A ;
  So that ;
  $Z \equiv ADL \sqcap \exists\ \texttt{hasUse.}\Big(x_{i1}\_On \sqcap \exists\ \texttt{hasStartTime.}(=, t)\Big)$;
  Activity Inference
  if *an activity Z_i is returned* **then**
    Report Activity A;
    Repeat process for next x_{i2}.
    if *an activity Z_i is not returned for next x_{i2}.* **then**
      Calculate β for x_{i1} and x_{i2} **if** β < *0.5* **then**
        Report Activity Boundary
      end
    end
  end
end

Scenario 3: The scenario where object observations are from different locations in the home environment. Activities are assumed to be location-based for example *Fridge*, *Cupboard* and *Microwave* are used for *Breakfast*, *Dinner* and *Snack* in the *Kitchen* with regards to the Kasteren home environment. In addition are toilet based objects like *Toilet flush* are normally kept in the *Toilet* and used for Toilet based activities (see Figure 5.18). *Breakfast* and *Toileting* have context descriptor as asserted as expressions 5.31 and 5.32 below.

$$\begin{aligned} Breakfast \sqsubseteq ADL \sqcap \exists \texttt{hasLocation.} \\ \Big(Kitchen \exists \sqcap \texttt{hasUse.}(Cupboard \sqcup Fridge \sqcup Microwave)\Big) \end{aligned} \tag{5.31}$$

$$\begin{aligned} Toileting \sqsubseteq ADL \sqcap \exists \texttt{hasLocation.} \\ (Toilet \exists \sqcap \texttt{hasUse.} ToiletFlush) \end{aligned} \tag{5.32}$$

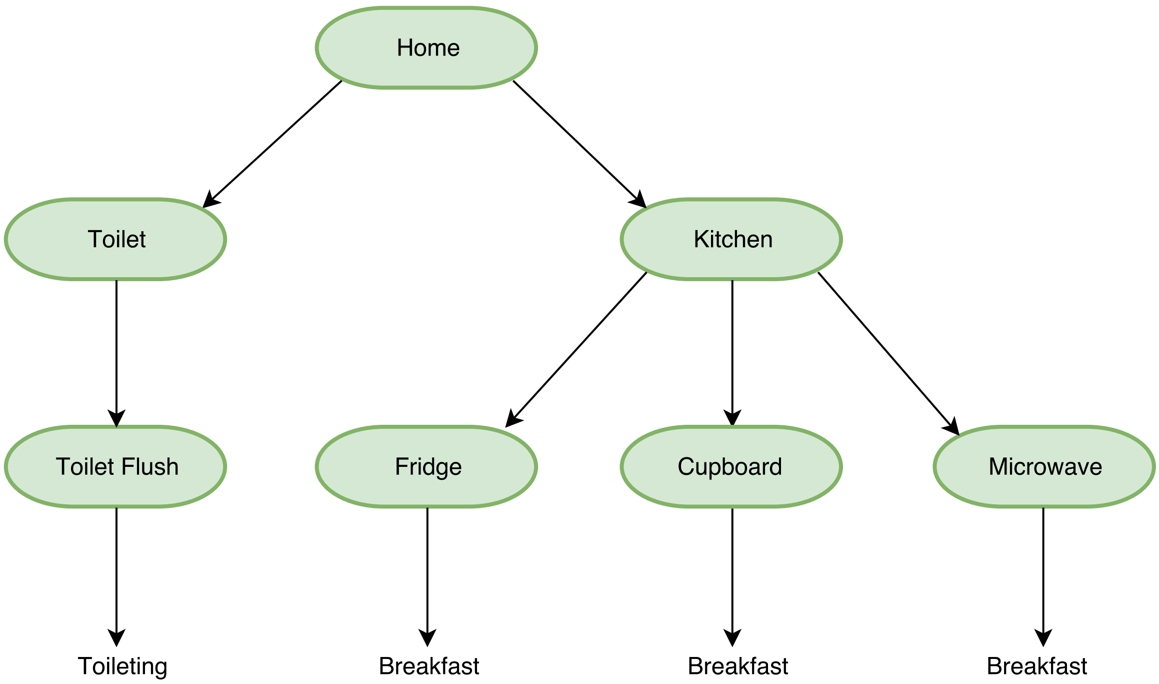


Figure 5.18: *Toileting* and *Breakfast* activity situation in different locations.

$$\beta = \frac{|(Fridge, Cupboard, Microwave) \cap (ToiletFlush)|}{|(Fridge, Cupboard, Microwave) \cup (ToiletFlush)|} \tag{5.33}$$

β in this case is 0.00 which is the lowest in terms of any value of similarity obtainable and indicative of the disimilarity of the *Kitchen* set of object and *Toilet* objects given expression 5.33. β with the object property `hasLastObject` can now be used to determine the persistence of an activity or to signal the end of an activity given consecutive observation of objects.

Order of precedence for the objects leading the activity situation could be any of path in Figure 5.19 and expression 5.34 *Breakfast*. The path for *Toileting* is illustrated in Figure 5.20 and expression 5.35

$$Breakfast \sqsubseteq ADL \sqcap \exists \texttt{hasLocation}.\Big(Kitchen \exists \sqcap \texttt{hasUse}.$$
$$(Cupboard \sqcap \exists \texttt{hasLastObject}.(Microwave \sqcap \exists \texttt{hasLastObject}.Fridge)\Big) \tag{5.34}$$

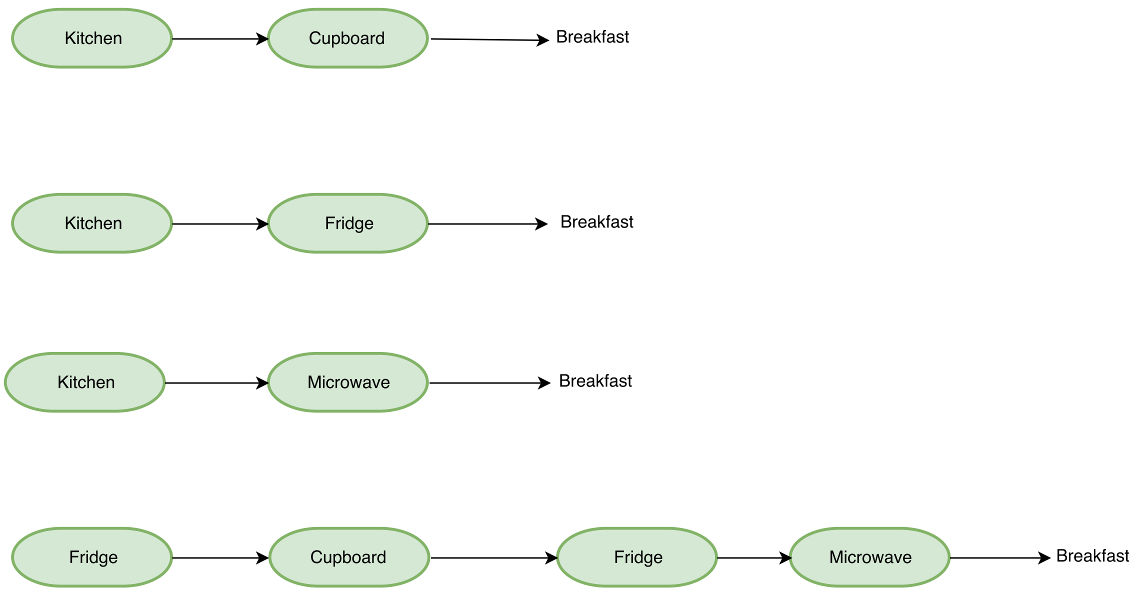


Figure 5.19: *Breakfast* processes

Figure 5.20: Toileting process

$$Toileting \sqsubseteq ADL \sqcap \exists \mathtt{hasLocation}. (Toilet\exists \sqcap \mathtt{hasUse}.ToiletFlush) \tag{5.35}$$

Toileting in the *Toilet* using *Toilet Flush* would have the activity path in Figure 5.20.

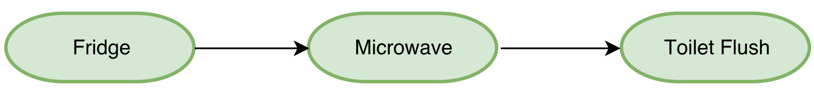


Figure 5.21: Objects from different location groups

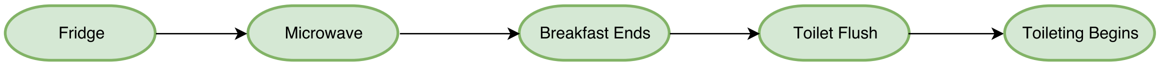


Figure 5.22: Using `hasLastObject` and β to signal end of activity

If objects are observed in the sequence as *Fridge*, *Microwave* and *Toilet Flush* as illustrated in Figure 5.21, `hasLastObject` could be used with β to indicate discontinued *Breakfast* after *Microwave* and commencement of *Toileting* as illustrated in the flow Figure 5.22. This is because β for *Microwave* and *Toilet Flush* falls below the similarity threshold value for the sets of location based objects which they belong.

From the above explanations and scenarios, the object property `hasLastObject` can be used to emphasise precedence given the transitivity it exhibits. Further choices and preferences which arises due to the order of object use can also be encoded using this object property. While it is being used contexts descriptors are also linked and with this granular activities are encoded.

5.6 Activity Recognition by Object Use Query

The activity recognition process uses the Algorithm 3 which performs a mapping of the activity situation using the observed objects. A comparison is made through reasoning by the ontology to retrieve the closest activity situation described by the contexts of object observed as sensor data. Activity recognition also follows the TOQL [14] adapted based on object usage to retrieve activity situations using sensor data concurrently available. The process uses object use query like constructs on the knowledge base to retrieve activity situation fitting the requirements of the query. As an advantage, sensor states and status of object use as implemented in the activity ontology can be used in queries to reflect real situations of object usage in the home environment. A typical query is comprised of SQL like construct (SELECT - FROM - WHERE) for OWL which treats the ontology classes and properties like database tables and columns. An additional AT construct in the query compares the time interval for which a property is true with a time interval or instant. Considering the scenario in the home environment where sensor status and outputs captured are reported as given in Table 5.2. The Algorithm 3 implements the activity recognition process. The input are observed sensor along their time lines as x_1,...x_n from segments s_1...s_n. The process maps sensor or object x_i on to the ontology applying inference rules to determine if x_i is a context descriptor for a static or dynamic activity. if the mapping using the inference rule returns activity, then it is reported as the activity situation for x_i.

Algorithm 3: Activity recognition algorithm.

Input: Observed Sensors X = $\{x_1$,...$x_n\}$ in Partition of Sensor Segment S = $\{s_1...s_n\}$, ADL ontology (ADL)
Result: Static Activity (SA) or Dynamic Activity (DA).
Begin:;
while *data stream is active* **do**
  Extract observed object $x_i \in X$ = $\{x_1$... $x_n\}$, from $s_i \in s_1...s_n$ **for each** $s_i \in S$.
  Create Activities $Z \equiv Z(x_1) \sqcup ... Z(x_n)$
  for each $x \in A$ do;
  Map x_i to a Static Activity (SA) or Dynamic Activity (DA)
  So that;
  $Z \equiv ADL \sqcap \exists$ hasUse.$\Big(x_i\_On \sqcap \exists$ hasStartTime.(=, t)$\Big)$
  Activity Inference
  if *an activity Z_i is returned* **then**
    Report (SA) or (DA) as recognised.;
    for all Repeat process for next x.
  end
end

Sensor ID	Sensor State	Time
6	Microwave_On	09:00:00
7	Fridge_On	09:01:01
8	PlatesCupboard_On	09:01:05

Table 5.2: A sample of sensor status and output

The question would be “What activity situation does these sensor outputs represent this particular time?” The query construct to provide the activity situation at that particular time is as presented below.

SELECT Activities zi, Activities.activityName
FROM Activities, Objects As Object1, Objects As Object2, Objects As Object3
WHERE Activities.hasUse: Object1 AND Object1.ObjectState LIKE ”Microwave_On” AT (09.00) AND Activities.hasUse: Object2 AND Object2.ObjectState LIKE ”Fridge_On” AT (09.00) AND Activities.hasUse: Object3 AND Object3.ObjectState LIKE ”PlatesCupboard_On” AT (09.00)

(5.36)

Notice the simplicity of expression with regards to time. The query automatically determines the activity situation with reference to the given time where the classes and properties are true.

5.7 Activity Ontology Update

Due to the dynamic nature of humans, preference for object usage for routine activities changes with time. Failing health and decline in the cognitive abilities of the elderly could also lead to changing habits and choice of object usage for activities. In this regards, it is logical for activity recognition models to have the capabilities to handle the variations in the choice of objects for activities. Traditional knowledge-driven activity recognition models, manage these changes and modifications by editing the entire activity ontology using editors like Protégé. This does not only make the process of changes cumbersome but also time-consuming.

The SPARQL update language for data manipulation facilitates modifications of ontologies and graphs through update operations for data to be inserted and deleted [46]. This thesis uses the SPARQL update operations to facilitate changes and modifications in object usage for specific routine activities which may arise due to changing preferences and habits. The update process follows the template as given in the schema 5.37. Typically, it performs a delete to remove data `a b c` from the ontology or graph using the DELETE clause; use the INSERT clause to insert or add data `d e f` to the ontology or graph, and the WHERE clause is used to indicate where the DELETE and INSERT modification is to be done e.g where the data corresponds to `x y z` .

```
WITH < Graph />
DELETE {
        a b c
}
INSERT {
        d e f
}
WHERE {
        x y z
}
```
(5.37)

This pattern of the operation follows a triple format of the subject (s), predicate (p) and object (o). The subjects corresponds to `a`, `d` and `x`, predicate corresponds to `b`, `e` and `y` and the object corresponds to `d`, `f` and `z` of the DELETE, INSERT and WHERE clauses respectively. The SPARQL update language protocol maps the subjects, predicates and objects using the DELETE, INSERT and WHERE clauses to the classes and properties of the domain activity ontology to perform the update or modification operations. The update operation does not all have to involve the DELETE and INSERT at the same time. It may be either depending on the modifications and changes to be made. Further illustration is made using the scenario 4.

Scenario 4: In scenario 2, *Breakfast* as an ADL has use (`hasUse`) the context descriptors *Fridge*, *Cupboard*, *Microwave* and *Toaster*. If due to changes in preferences or habits, the context descriptors for *Breakfast* becomes *Fridge*, *Cupboard*, *Cooktop* and *Toaster*. This then requires modifying and updating the ontology so that now *Breakfast* will exclude *Microwave* to be deleted and to include *Cooktop* to be inserted. The process of effecting these changes would involve the DELETE of *Microwave* and INSERT of *Cooktop* WHERE activity `hasUse` (*Fridge*, *Cupboard* and *Toaster*) with the schema 5.38. The modification requires mapping the subject, predicate and object contents of the schema to the activity ontology through the clauses. For example the DELETE clause deletes `'Microwave_On'` mapped with the predicate as object property attribute `hasUse:Object` for the ontology activity class `?Activity`. This is same for the INSERT and WHERE clauses of the update schema.

```
Prefix    adl < Graph />
DELETE  {
            ?Activity  adl:hasUse  'Microwave'
}
INSERT  {
            ?Activity  adl:hasUse  'Cooktop'
}                                                    (5.38)
WHERE  {
            ?Activity  adl:hasUse  'Fridge'.
            ?Activity  adl:hasUse  'Cupboard'.
            ?Activity  adl:hasUse  'Toaster'
}
```

Modifications and updates through this process do not only save time but also the process of editing the entire activity ontology. *Breakfast* is then encoded and expressed as 5.39. The resulting data is added to the ABox for *Breakfast*.

$$Breakfast \sqsubseteq ADL \sqcap \exists \texttt{hasUse}.\Big(Fridge \sqcup Cupboard \sqcup Toaster \sqcup Cooktop\Big) \quad (5.39)$$

Updates to an activity ontology can be initiated and facilitated through the algorithm 4 which triggers the update operation in the event of repeated null activity recognition. In reality, the need for ontology update arises when a set of observed objects fails to retrieve an activity(s) from the object query described above in section 5.6. This means that the object concept change(s) would have to be discovered through a context description process as discussed in Chapter 4 of this thesis. The discovered changes in object concepts are then updated through the ontology update schemes similar 5.37 and 5.38

If a set of observed objects their time lines as x_1,...x_n from segments s_1...s_n fails to retrieve an activity, then the Algorithm 3 is extended to include the context description process and the update operation through the Algorithm 4.

Algorithm 4: Activity ontology update algorithm.

Input: Observed Sensors $X = \{x_1,...x_N\}$ in Partition of Sensor Segment $S = \{s_1...s_s\}$, ADL ontology (ADL)

Result: Static Activity (SA) or Dynamic Activity (DA).

Begin:;

while *data stream is active* **do**

Extract observed objects $x_i...x_n \in X = \{x_1... x_N\}$, from $s_i \in s_1...s_s$ **for each** $s_i \in S$.

Create Activities $Z \equiv Z(x_1) \sqcup ... Z(x_n)$

for each $x \in A$ do;

Map $x_i...x_n$ to a Static Activity (SA) or Dynamic Activity (DA)

So that;

$Z \equiv ADL \sqcap \exists$ hasUse.$\Big(x_i\_On...x_n\_On \sqcap \exists$ hasStartTime.(=, t)$\Big)$

Activity Inference

if *an activity Z_i is returned* **then**

Report (SA) or (DA) as recognised.;

for all Repeat process for next x.

if *an activity Z_i is not returned for $x_i...x_n$.* **then**

Perform Context description and

Update activity ontology for context changes

Repeat activity recognition

Report (SA) or (DA) as recognised.;

end

end

end

5.8 Conclusion

This Chapter presented activity ontology modelling for activity recognition. As part of the process of knowledge representation for activity recognition, modelling of ADL concepts and with contexts was presented. The chapter also considered activity as an ADL concept and further discussed modelling it with contexts. Temporal attributes of ADL concepts through temporal representation was also presented as part of ADL contexts that could help in improving activity recognition. As part of modelling activity concepts, it also presented modelling of sensors and object concept to reflect the situations in the home environment. Modelling status of sensor outputs when in use and integrating object usage was also presented to further reflect the situations involving ADL process in the home environment. To also enhance this, static and dynamic activities, modelling fine grain activities with three scenarios and then activity recognition by object use query. The chapter is concluded with updating activity ontology through the update Algorithm 4 without editing the entire ontology

Chapter 6

VALIDATION

Chapter 5 considered activity ontology modelling for activity recognition. In particular, it focused on how to model objects and activities as ontology concepts. The resulting activity ontology is designed to be composed of the TBox and ABox ensuring that the activity situations in the home environment are reflected in the process. This chapter presents the experimental evaluation which validates the proposed the hybrid approach to activity recognition described in Chapters 3, 4 and 5. This process of evaluation and validation was carried out through various experiments on two publicly available datasets. Details of the experimental methodology and evaluation are presented in section 6.1 of this chapter emphasising on the proposed approach. In addition, section 6.2 presents the experiments, results and performance evaluation. Discussions of results are also presented as a way of highlighting the effectiveness of the proposed hybrid activity recognition approach. The chapter also makes comparisons with results obtained using other methods on the datasets. Section 6.3 summarises and concludes the chapters.

6.1 Experimental Methodology and Evaluation

The process of activity recognition as considered in previous chapters follows several techniques and approaches. Different methods have been followed by researchers to evaluate and validate their approaches as reported in the following [30, 96, 73, 104, 129, 130, 110]. However, these methods to a large extent are similar and can be summarised as;

- Deploy monitoring devices in the target environment.
- People perform activities in the target environment.
- Capture activities through monitoring devices and keep a record of ground truth using an appropriate annotating system.

- Carry out experiments using the developed algorithm, technique or approach on the captured dataset.
- Compare experimental results using ground truth
- Carry out performance analysis on the experimental results.

While these summarised steps have led to the validation of the approaches cited above, limitations still abounds and includes:

- It almost impossible to tag every single object in the home environment with sensors or use wearable sensor to measure every single needed attribute needed for perfect activity recognition. This limitation can be attributed to cost implication of sensors, associated cost of processing sensor data and ease of use with regards to wearable sensors.
- Ground truth are often not perfect and sometimes may not reflect the reality of the dataset annotations. This poses the challenge of affecting the performance of approaches.
- The ethical implication associated to data collection and in some cases the intrusive nature of the process might be invasive to occupant(s) of the home environment. These raises privacy concerns and limits the extent to which the research can go.

Although these limitations abound, they do not discredit the approaches but serve to provide the basis for future validations. Efforts by researchers in this emerging field should be, to take into account these limitations to develop techniques that are significantly comparable and even better in performance.

6.1.1 Dataset

To evaluate and validate the proposed activity recognition approach, this thesis have adopted some of the steps highlighted above in using the Kasteren dataset[1] and the Ordonez dataset[2]. The choice of these dataset fulfils conditions premised on the method of evaluation and to a large extent addresses some of the limitations mentioned above. The dataset chosen so far are publicly available and as such, this addresses the limitations of high cost, and the lengthy process involved in collecting and processing the dataset. Legal and ethical constraints that would have been major limitations are also addressed by the use of these

[1] https://sites.google.com/site/tim0306/datasets
[2] http://mlr.cs.umass.edu/ml/machine-learning-databases/00271/

publicly available dataset. In arriving at a suitable dataset that addresses other limitations mentioned, the activities involved in these dataset were carefully thought of, the object used to perform the activities and the availability of ground truth with which to make comparisons in terms of evaluation and performance. The choice of these dataset was further driven by the fact that the Kasteren and Ordonez dataset contain a lot of sensor activations with dense sensing applied. Different types of sensor (for example pressure sensors, magnetic sensor etc tagged to home objects like microwave, dishes, cups) were used to capture contexts for the different activities to allow for a rich collection of different types of data. These sensor/object activations have been annotated and labelled to suggest activity events, situations and tasks being performed. The sensor/object activations as they are, represents samples of different activities enabling a learning process. To further enhance the learning process, the activities contained therein have been performed in varied ways and accurately annotated in the ground truth. In addition, there are a set of ground truths which have been carefully labelled and annotated as representations of the series of sensor/object activations.

6.1.1.1 Kasteren Dataset

The Kasteren dataset was generated by a set of simple state-change sensors installed in three different environments labelled as Houses A, B and C (See Appendix A: Figure 7.1 for the floor plan). The three datasets includes a set of eight different activities: *Leaving*, *Toileting*, *Showering*, *Sleeping*, *Breakfast*, *Dinner*, *Drink*, and *Idle* which corresponds to times when no activity took place. A brief overview of the dataset are as given below. Also, see table 6.1 for a summary of activity instances for the Kasteren Dataset .

- The House A dataset tracked the activities of a 26 year old man in a single occupancy residence. The dataset was collected over a period of 28 days, recording 2120 sensor events. The sensor network was implemented using digital reed switches, each mounted to a RF Monolithics DM1810 module to form a wireless network. Dense sensing method was applied attaching sensor modules to 14 objects of interest across rooms in the house, each producing a binary reading to indicate whether or not a sensor is firing. The instrumented objects are -Wireless sensor network connected to a base-station to for the record of trace sensor observations, a Bluetooth headset worn by the user was used to for recording of the activity data through speech recognition of the spoken annotation. Object tagged in this House A includes:

    1. Bedroom: *Door*
    2. Main Entrance: *Door*

3. Bathroom: *Door*
4. Toilet: *Door* and *Toilet Flush*
5. Kitchen: *Microwave*, *Groceries Cupboard*, *Plates Cupboard*, *Freezer*, *Fridge*, *Pans Cupboard*, *Cups Cupboard*, *Washing machine*, and *Dishwasher*.

- House B dataset is similar to House A tracking the same activities but with a 28 year old male in an apartment over a period of 14 days. Although House B had more home objects involved than House A, it followed the same dense sensing approach with the use of binary sate change sensors. The setup included pressure mats for detecting lying and sitting, mercury contact sensor to detect movement of objects and infrared sensors for motion detection. Unlike House A, the starting and end times of the activities were recorded from the individual watch. The annotated activities were handwritten in a diary from sheets of paper left where activities have been performed. Object tagged in this House B includes:

  1. Bedroom: *Door*, *Bedmat Right* and *Bedmat Left* (pressure mats), *Dresser*, and a Passive Infra-Red (PIR) sensor.
  2. Main Entrance: *Door*
  3. Bathroom: *Door*
  4. Toilet: *Door* and *Toilet Flush*, *Sink* (float), and a PIR sensor
  5. Kitchen: *Microwave*, *Groceries Cupboard*, *Plates Cupboard*, *Fridge*, *Cutlery Drawer*, *Stove lid*, *Toaster*, and a PIR sensor

- The House C dataset tracked the activities of a 58 year old man in a single occupancy residence over 19 days. Although the same activities tracked in House A and B were tracked in House C, House C significantly differs from Houses A and B due to its having two floors. Dense sensing was also applicable in this setting using reed switches, pressure mats, mercury contacts and passive infrared motion sensors similar to House B. An additional pressure sensor was attached to the couch in the lounge. The activity data was collected and recorded using the same blue tooth approach as in House A. Object tagged in this House B includes:

  1. Bedroom: *Door*, *Bedmat Right* and *Bedmat Left* (pressure mats) and a *Dresser*
  2. Main Entrance: *Door*
  3. Bathroom: *Door* (Left and Right swing doors), *Bathtub* PIR, Sink (float)
  4. Downstairs Toilet: *Door* and *Toilet Flush*

Activities	Instances		
	House A	**House B**	**House C**
Leaving	36	24	47
Toileting	114	27	99
Showering	24	11	14
Sleeping	25	14	19
Breakfast	20	9	18
Dinner	10	6	11
Drink	20	8	10

Table 6.1: Summary of activity instances for the Kasteren Dataset

5. Upstairs toilet: *Toilet Flush*.
6. Kitchen: *Groceries Cupboard*, *Cup Cupboard Bowl Cupboard*, *Herb Cabinet* and *Plate Cabinet*, *Fridge*, *Freezer*, *Food Scraps bin*, *Cutlery Drawer*, *Pots Cupboard* and *Pans Cupboard*, *Microwave*, *Drawer* with keys to the backdoor and a PIR sensor.

6.1.1.2 Ordonez Dataset

The Ordonez dataset is similar to the Kasteren dataset and generated using a set of state-change sensors installed in two different environments (Houses A and B with single occupants). Similar to the Kasteren dataset, the Ordonez dataset includes activities like *Leaving*, *Toileting*, *Showering*, *Sleeping*, *Breakfast*, *Dinner*, *Drink*, and *Idle* which corresponded to times when no activity took place. Other activities in addition which did not feature in the Kasteren include *Lunch*, *Snack*, *Spare time/TV* and *Grooming*. A brief overview of the dataset are as given below. Also, see table 6.2 for a summary of activity instances for the Ordonez Dataset. .

- The House A dataset was collected over a period of 14 days in a 4 room house. 12 state change sensors were used to capture 410 events. Types of sensors and objects tagged includes:

  1. PIR: *Shower*, *Basin*, *Cooktop*
  2. Magnetic: *Maindoor*, *Fridge*, *Cabinet*, *Cupboard*
  3. Flush: *Toilet*
  4. Pressure: *Seat*, *Bed*

Activities	Instances	
	House A	**House B**
Leaving	14	38
Toileting	44	93
Showering	14	11
Sleeping	14	29
Grooming	51	113
Spare Time	77	116
Breakfast	14	22
Lunch	9	13
Dinner	Na	11
Snack	11	47

Table 6.2: Summary of activity instances for the Ordonez Dataset

5. Electric: *Microwave*, *Toaster*.

- The House B dataset was collected over a period of 21 days in a 5 room house. Similarly, 12 state change sensors were used but this time to capture 2314 events. Types of sensors and objects tagged includes:

  1. PIR: *Shower*, *Basin*, *Door Kitchen*, *Door Bathroom*, *Door Bedroom*.
  2. Magnetic: *Maindoor*, *Fridge*, *Cupboard*.
  3. Flush: *Toilet*
  4. Pressure: *Seat*, *Bed*
  5. Electric: *Microwave*

6.1.2 Evaluation Methodology

To validate and evaluate the performance of the proposed activity recognition approach, a 'leave one day out' cross validation was used [10]. Cross validation is a standard technique used to estimate how accurate a predictive model will perform. It involves splitting the given dataset into a training and test subsets. The training subset is used to train the model and then the test subset is used to determine the accuracy of its performance. In k-fold cross-validation, the dataset is split or partitioned into equal sizes of k subsets. A subset is set aside for test while $k - 1$ of the dataset is used to train the model. The cross validation process is repeated for k times with k subset used once for the test purpose. With regards to 'leave one day out' cross validation, the dataset is split according to the number of days for which the dataset was captured. If there are k days amount of data, then $k - 1$ days amount of the data are used to train the model leaving one day amount of the data for testing. The process is then repeated for k times with each day data used once for test purpose.

The accuracy, precision, recall and F-measure (also referred to as also F-Score) are calculated as statistical metrics used to determine the performance of the model.

As earlier mentioned, a 'leave one day out' cross validation was used to evaluate the performance of this approach. To do this, a 60 seconds sliding window intervals was applied to the sensor data of Kasteren and the Ordonez dataset using the start times of the sensor data. The 60 seconds intervals resulted to segments of object observations which were used to form the segment-object-frequency matrix as explained in subsection 4.3.2 for the preparation of "bag of object observations". The resulting matrix were partitioned into k (The number of days for which the dataset represents) days folds such that each fold corresponds to the data in each of the respective days. The training phase involved using the LDA topic model on the $k-1$ amount of training data for the activity-object discovery. The activity-object usage discovered from the training phase was then used as knowledge of object use for the activity situations, and with the application of Algorithm 1 for context descriptions were determined, context descriptors for the activities. The results of the context description process were used to develop the activity ontology on which we ran the test subset for activity recognition. Kasteren et al [133, 132] and Ordonez et al [99] both applied 'leave one day out' cross-validation methodology. However, comparisons were made with the results published using these datasets.

6.1.3 Evaluation Parameters

To evaluate the performance of the proposed approach, there is need to use frequently used activity recognition evaluation metrics. The work presented in this thesis is based on ontology-driven activity recognition dependent on the knowledge of object use from the context description process. The evaluation process is based on activity recognition experiments using the datasets comparing the output results with the ground truth and from this determining the True Positives, True Negatives, False Positives and False Negatives for every activity.

- True Positives (TP): are the activities that have been correctly recognised as belonging to the target category.

- True Negatives (TN): are the activities that have been correctly recognised as not belonging to the target category.

- False Positives (FP): are the activities that have been mistakenly recognised as belonging to the target category. False positives are also referred to as wrong selections.
- False Negatives (FN): are the activities that have been mistakenly recognised as not belonging to the target category. False negatives are also referred to as missed selections.

Since the evaluation aims to determine how well the algorithms recognise tasks from object usage, we also use the following metrics.

- Average Accuracy: The number of correctly recognised situations over the total numbers of the activity situations averaged over the activity situations in the dataset. Where accuracy is calculated as:

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \times 100\% \tag{6.1}$$

- Average Precision: The number of times this proposed technique correctly recognises an activity situation divided by number of times the same activity situation is recognised, averaged over each activity situation in the dataset. Where precision is calculated as:

$$Precision = \frac{TP}{TP + FP} \times 100\% \tag{6.2}$$

- Average Recall: The number of times the proposed technique correctly recognises an activity situation divided by number of times the same activity situation occurs in the dataset, averaged over each situation in the dataset. Where recall is calculated as:

$$Recall = \frac{TP}{TP + FN} \times 100\% \tag{6.3}$$

- Average F-measure: This is dependent on the precision and the recall. It is also the harmonic mean of precision and recall calculated as:

$$F - measure = \frac{2 \times Precision \times Recall}{Precision + Recall} \times 100\% \tag{6.4}$$

In predictive models and activity recognition techniques precision, recall, and F-measure are mostly used. So, the evaluation results were based on these metrics. The precision provides the measure of the ability of this hybrid approach to recognise the desired activity and the F-measure measures the recognition capability of the approach. It important to consider the total proportion on which all these metrics are based and from this, the test results can be drawn. The results are based on the total proportion of the test instances so as to provide an overall measurement in percentage.

6.1.4 Evaluation Framework

To implement the evaluation framework proposed in this thesis, the following development environments with software libraries were sought for and used:

- Matlab R2015b[3] is a computing environment for numerical analysis and supports implementation of machine language algorithms. Due to its ability to implement machine learning algorithms, Matlab R0215b to implement the silhouette analysis through K-Means clustering to determine the number of activity topics was used. Afterwards, we implemented the LDA topic model in the Matlab environment to determine the activity-object distributions.

- Protégé [4] is an open source software environment for developing and editing ontology. Protégé was used for the ontology development and implementation of context descriptors for activity situations. The activity-objects usage discovered using with the topic model in the Matlab environment were then developed as ontology concepts as rightly explained in Chapters 4, 5 and 6. The expressiveness of activity and object concepts which this editor enhances the process of activity recognition especially for the end user.

6.2 Experiments, Evaluation and Results

Using the methods and approach outlined above, experiments were performed to validate the performance of the proposed approach on the datasets. Experiments were performed to determine the model learning capabilities of this approach. The experiments performed includes: *i)* Object Use and Context Descriptor Similarity, *ii)* Activity Ontology and Recognition Performance, *iii)* Evaluation of Static and Dynamic Activities, *iv)* Evaluation of the Learning Model, *v)* Evaluation of the Activity Ontology update, *vi)* Evaluation of the Activity Boundary Detection algorithm. Finally, comparisons were made with the results published using other methods on these datasets.

[3]https://uk.mathworks.com/downloads/
[4]http://protege.stanford.edu/

Home Environment	Number of Segments
Kasteren House A	33,007
Kasteren House B	17,143
Kasteren House C	24,337
Ordonez House A	20,193
Ordonez House B	30,115

Table 6.3: Number of Segments from the Partitioning process from the Kasteren and Ordonez datasets

6.2.1 Activity-Object Use Discovery and Context Description Evaluation

To generate the context descriptors for routine activities, LDA topic model process described in Chapter 4 was followed. Recall that the LDA process requires activity topic numbers and the "bag of object observations".

As previously mentioned, the "bag of sensor observations" can be constructed by partitioning the sensor dataset into time slices of constant length. For the experiments in this thesis, Kasteren and Ordonez sensor data were partitioned and segmented in intervals of time t = 60 seconds, based on the contributions of Kasteren et al [132] and Ordonez et al [99]. This time interval is considered long enough for a segment of object representing a candidate activity and short enough to provide good accuracy labelling results. Also, the shorter the time interval segments tend to be more sparse and larger time interval the partitioned segments lose the capability to capture shorter activities. Table 6.3 represents the number of segments based on the time t = 60 seconds interval for Kasteren and Ordonez datasets. The partitioned segments are then used to form a sensor-segment frequency matrix which also includes the object counts from each segment. The objects are then represented as their aliases as in *Seat* (S), *Basin* (B), *Bed* (A), *Microwave* (M), *Cupboard* (C), *Fridge* (F), *Cabinet* (N), *Toilet* (T), *Shower* (Sh) etc. to be encoded onto the partitioned segments and "bag of objects observations".

The activity topic number is determined using the silhouette method through K-Means clustering rather than a random guess of the number of activities represented in the dataset. To do this, first, cluster the sensor data features using K-Means setting the number of clusters from 5 to 10 as a reasonable range of probable number of activities which this represents. The silhouette analysis which is a measure of the mean silhouette width of each of the clusters was then performed. Charts in Figures 6.1 and 6.2 below illustrates the optimal number clusters indicative of the number of activity topics for Kasteren House A, B and C, Ordonez A and B.

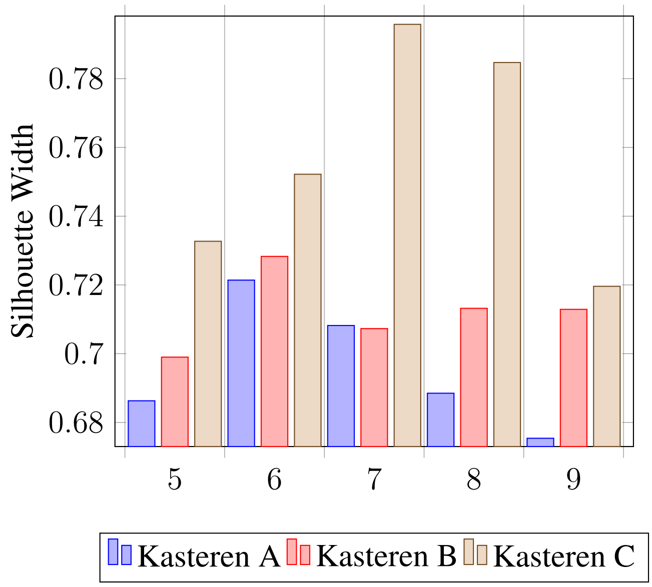


Figure 6.1: Silhouette width results for the Kasteren Houses A, B and C suggesting 6, 6 and 7 activity topics respectively for the houses

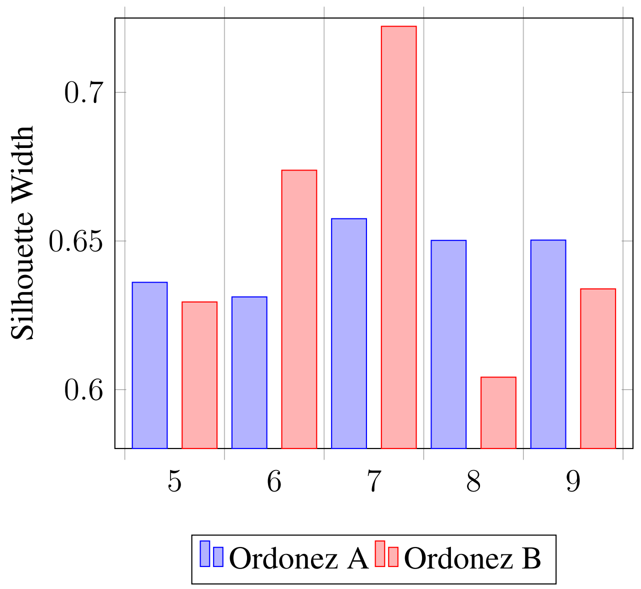


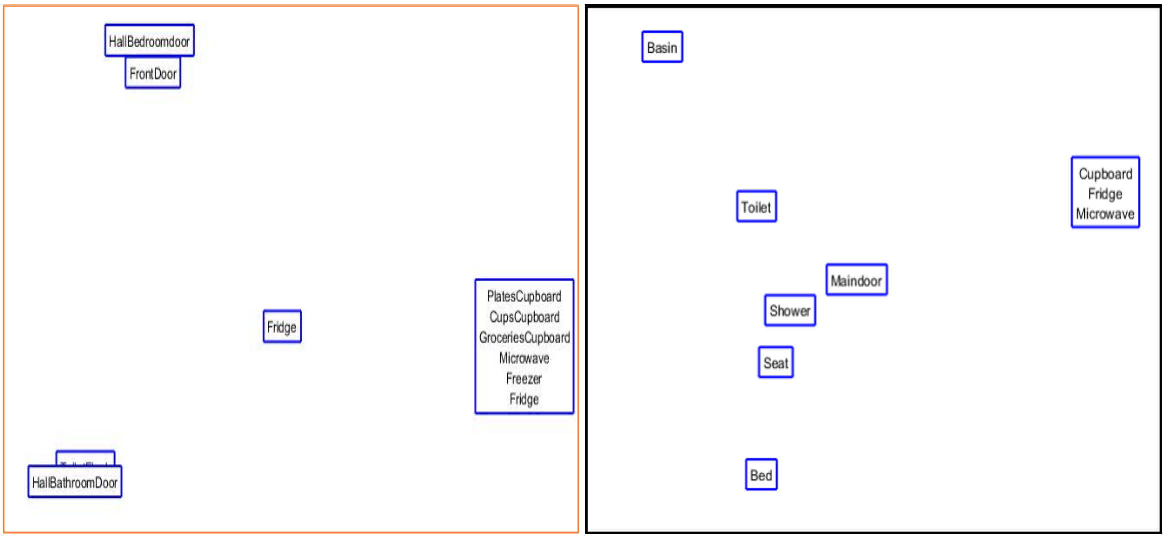
Figure 6.2: Silhouette width results for the Kasteren Houses A, B and C suggesting 6, 6 and 7 activity topics respectively for the houses

For the LDA inference, the constructed "bags of object observations" are used, the optimal activity topic numbers from the silhouette method and set the dirichlet prior α to 0.01. Figures 6.3 (a) and (b) shows sample visualisation activity-object distributions for Kasteren House A and Ordonez A. Visualisations for Kasteren B, C and Ordonez B have been included in appendix A Figure 7.2 . The process of activity topic number determination and activity-object distribution is in its entirety unsupervised. Finally, the context descriptors for the specific routine activities are achieved using the Algorithm 1 above with dependency on the activity-object distributions from the LDA and μ. The idea is that for an object to be a context describing an activity topic, it must have been assigned to an activity topic by a number of times greater than the threshold μ. μ was determined by computing the mean value M and standard deviation SD of the number of times an object has been allocated to an activity topic (see expression 6.5). The threshold values vary as $\mu_1 ... \mu_K$ for the different activities $k_1 ... K$ since the unique objects $x_{1ki} ..., x_{Nki}$ have different numbers of occurrences in the dataset. Further, some of the activity topics were annotated in line the ground truth, and matched them with their respective context descriptors as given in Tables 6.4 and 6.5 for Kasteren House A and Ordonez House A respectively (See Tables Appendix A 7.1, Appendix A 7.2 and Appendix A7.3 for Kasteren Houses B, C and Ordonez B respectively).

$$\mu = M + SD \tag{6.5}$$

Activities	**Context Descriptors**
Leaving	Front Door.
Toileting	Hall Toilet Door, Toilet Flush.
Showering	Hall Bathroom Door.
Sleeping	Hall Bedroom Door.
Make Food	Fridge, Plates Cupboard, Cups Cupboard, Groceries Cupboard, Microwave, Freezer.
Make Drink	Fridge.

Table 6.4: Activity Concepts and the discovered context descriptors for Kasteren House A



(a) Kasteren A (b) Ordonez A

Figure 6.3: Visualisation of LDA generated Object classifications for Kasteren A and Ordonez A

Activities	**Context Descriptors**
Leaving	Main Door.
Toileting	Toilet, Basin.
Showering	Shower.
Sleeping	Bed.
Make Food	Cupboard, Fridge, Microwave, Toaster.
Spare Time	Seat.
Grooming	Basin, Cabinet.

Table 6.5: Activity Concepts and the discovered context descriptors for Ordonez House A

The outcome of this process was evaluated by analysing the similarities and relatedness between the contexts describing the activities since they were discovered in an unsupervised manner. This is achieved using the Jaccard [72] and Dice [126] Coefficients expressed as eq. 6.6 and 6.7 respectively to measure to similarities of the context descriptors and compared these for all the activities. The Jaccard coefficient measures similarities between sample sets A and B by the ratio of the size of their intersection to the union of the sample sets while the Dice similarity Coefficient is the ratio of twice the size of the intersection of the sets to the sum of the sizes of the sets.

$$Jaccard(A, B) = \frac{|A \cap B|}{|A \cup B|} \tag{6.6}$$

$$Dice(A, B) = \frac{2|A \cap B|}{|A| + |B|} \tag{6.7}$$

The resulting score of similarities are always in the interval of 0 and 1 with a normalising factor 1 which is divided by the non-zero element in the sample sets being compared. The score of the Jaccard and Dice coefficients is 1 if the sample sets are similar and equal. It is 0 when there are no matching elements. The Jaccard and Dice coefficients have been selected, because they are been well founded measures used as basis of determining similarities between sample sets of values. This method of determining the similarities between activity contexts works well with sufficient data, but result may lead to poor conclusion with insufficient data. An implementation example to determine the similarity between activities A = *Microwave*, *Fridge*, *PlatesCupboard*, *Groceries Cupboard*, *Pan Cupboard*, *Freezer*, B = *Microwave*, *Fridge*, *Groceries Cupboard*, *CupsCupboard* and C = *HallBathroomDoor*, *HallToiletDoor*, *ToiletFlush* results to 0.43 and 0.6 between A and B, 0.00 and 0.00 between A and C, 0.00 and 0.00 between B and C for Jaccard and Dice similarity indices respectively. Although, the results suggest marked similarities between A and B using the two methods, a threshold value is needed to conclude on whether both activity concepts are to be considered as the same or similar. On the other hand, A and C, B and C show no similarities and can not be considered to have any context descriptor association. By applying these methods, the resulting similarity indices for context descriptors of the activity topics for the Kasteren House A are presented in the Tables 6.6 and 6.7. Applying the Jaccard and Dice coefficients, the similarity indices were based between the resulting context descriptors of the activities to be above a threshold value θ. The choice of similarity thresholds is dependent on the variability of the similarity indices determined. The author of [10] determined similarity thresholds using median absolute deviation as a means of determining the threshold similarity index which worked well for the similarity indices obtained in their case. Considering the similarity indices obtained for the Kasteren and Ordonez datasets, $\theta = 0.5$ was chosen as the threshold value for Jaccard and Dice similarity coefficients. This choice was based on the upper and lower limits of the coefficients and it represents a more robust and resilient indicator for determining similarities between activity sample sets unlike the values for the standard deviation and median absolute deviation which falls below this choice of threshold.

	Leaving	Toileting	Showering	Sleeping	Make Food	Drink
Leaving	1.00	0.00	0.00	0.00	0.00	0.00
Toileting	0.00	1.00	0.00	0.00	0.00	0.00
Showering	0.00	0.00	1.00	0.00	0.00	0.00
Sleeping	0.00	0.00	0.00	1.00	0.00	0.00
Make Food	0.00	0.00	0.00	0.00	1.00	0.17
Drink	0.00	0.00	0.00	0.00	0.17	1.00

Table 6.6: Jaccard similarity indices for context descriptors for Kasteren House A

	Leaving	Toileting	Showering	Sleeping	Make Food	Drink
Leaving	1.00	0.00	0.00	0.00	0.00	0.00
Toileting	0.00	1.00	0.00	0.00	0.00	0.00
Showering	0.00	0.00	1.00	0.00	0.00	0.00
Sleeping	0.00	0.00	0.00	1.00	0.00	0.00
Make Food	0.00	0.00	0.00	0.00	1.00	0.29
Drink	0.00	0.00	0.00	0.00	0.29	1.00

Table 6.7: Dice similarity indices for context descriptors for Kasteren House A

Discussion

The main aim of the activity-Object use discovery and context description is to determine the likely object usage and contexts for specific routine activities. This is based on the idea that to design and develop the activity ontology model, the knowledge of the contexts which describes the routine activities needs to be acquired through a known robust process rather than guessing or generically determined from an everyday common knowledge of activity process. Topic model LDA was proposed to discover the likely object use for the routine activities. But the number of activity topics needed by the LDA as a parameter was determined by a silhouette method through K-Means clustering as illustrated in Figures 6.1 and 6.2. These activity topic numbers are indicative of the number of activities which corresponds to 6 activities for Kasteren A and B, 7 activities for Kasteren C, Ordonez A and B. The LDA inference results as visualised in Figures 6.3 and Figures 7.2 show the likely object use for the activities. The Algorithm 1 uses the activity-object distributions and a threshold value μ to determine the context descriptors for the activities. Some of the activities were then annotated in line with the ground truth, but it was noticed that *Breakfast*, and *Dinner* share same and similar object use and so they were annotated as *Make Food* so that Kasteren A and B now have 6 activities against 7 activities as in the ground truth. For Ordonez A and B, 7 activities were discovered against 10 activities as in the ground truth (see Tables 6.4 and 6.5 for the context descriptors of Kasteren A and Ordonez A). This was due to *Breakfast*, *Lunch*, *Dinner* and *Snack* using same and similar objects. To assess and evaluate this process, the Jaccard and Dice coefficients was used to determine how similar the contexts describing the activities are. A threshold as a value was also applied to determine whether the contexts descriptors between activities are similar, equal or match. Similarity indices are presented in parts in Tables 6.6 and 6.7. So far, the result suggests marked similarities between *Make Food* and *Drink* for the Kasteren Houses A, B and C but they fall below the threshold suggesting that the sets are not equal and the same. *Grooming* and *Toileting* for the Ordonez dataset showed marked similarities but these also fell short of being classed as the same or equal. The context descriptors discovered through this process provides the basis for modelling the objects and activity concepts in the activity ontology.

6.2.2 Activity Ontology and Recognition Performance

To this point, the context descriptors for the various activities of the Kasteren and the Ordonez datatsets have been generated. To facilitate activity recognition and eventual evaluation of the approach, the activities were modelled and the context descriptors from the previous section into an ontology activity model. The modelling process is designed using OWL language and developed in the ontology editor Protégé. The generic activity ontology is extended in Figure 5.5 to include the context descriptors and activity concepts from the various houses. To enhance and support a unified ontology model and with common concepts shared and which can be reused across the similar home environments, a unified activity ontology for the Kasteren and the Ordonez homes has been developed. Figures 6.4 and 6.5 illustrates the unified concepts for the Kasteren and Ordonez homes respectively. The green coloured rounded rectangles represents common object concepts in both homes. In Figure 6.4, the yellow rounded rectangle has been used specifically for House A concepts which are not shared in B and C, blue rectangle for House B and red rectangle for C concept. Figure 6.5 represents the common concepts for the Ordonez House A and B. Yellow rounded rectangle corresponds to specifically House A concept, Blue for House B concepts and the green rounded rectangle for concepts common in both houses. These unified ontology models can also be extended and adapted further for similar homes thus reducing the amount of time take to construct and develop activity ontologies.

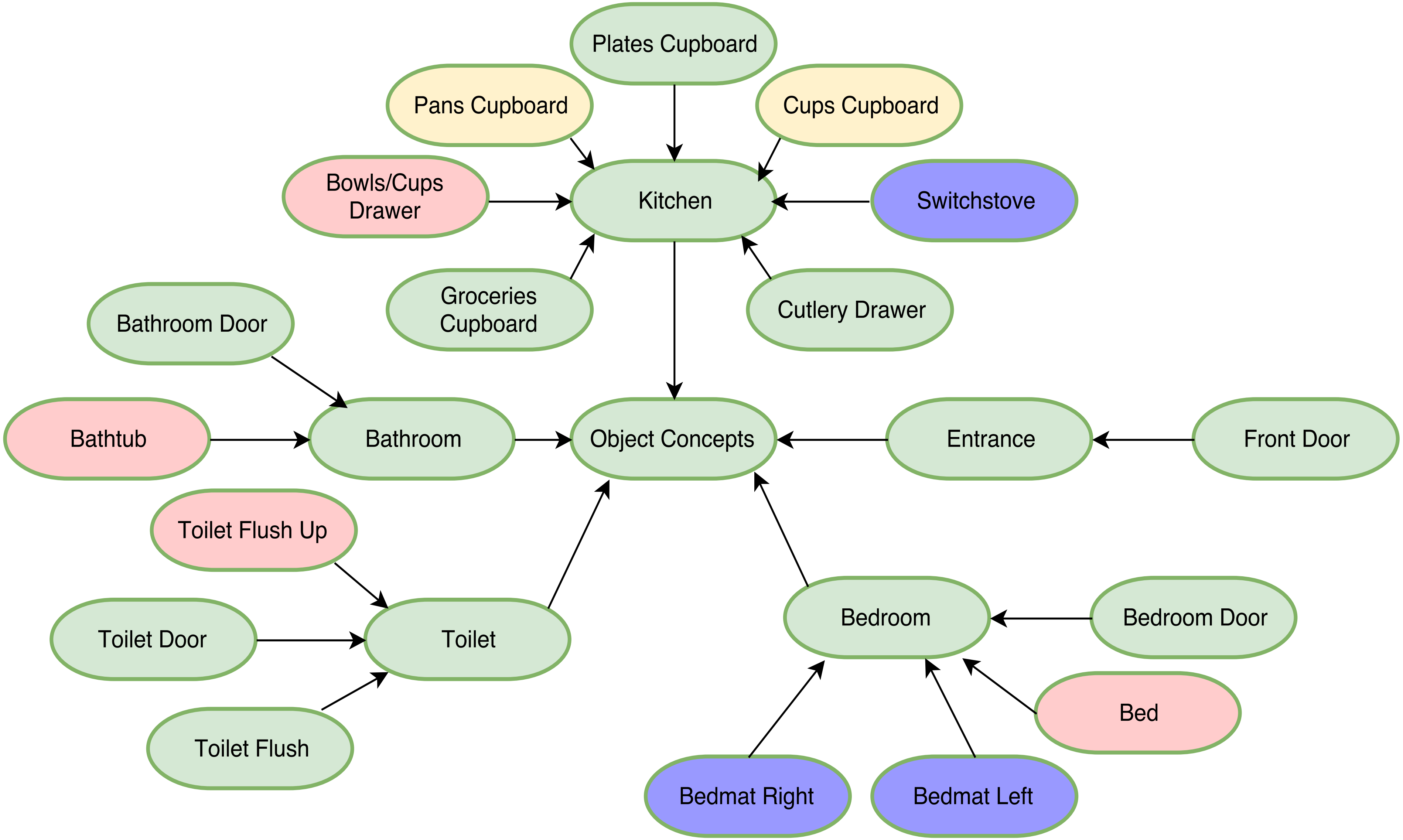


Figure 6.4: Common object concepts for Kasteren Houses A, B and C. Green for common object concepts, yellow for House A, blue for House B and red for House C

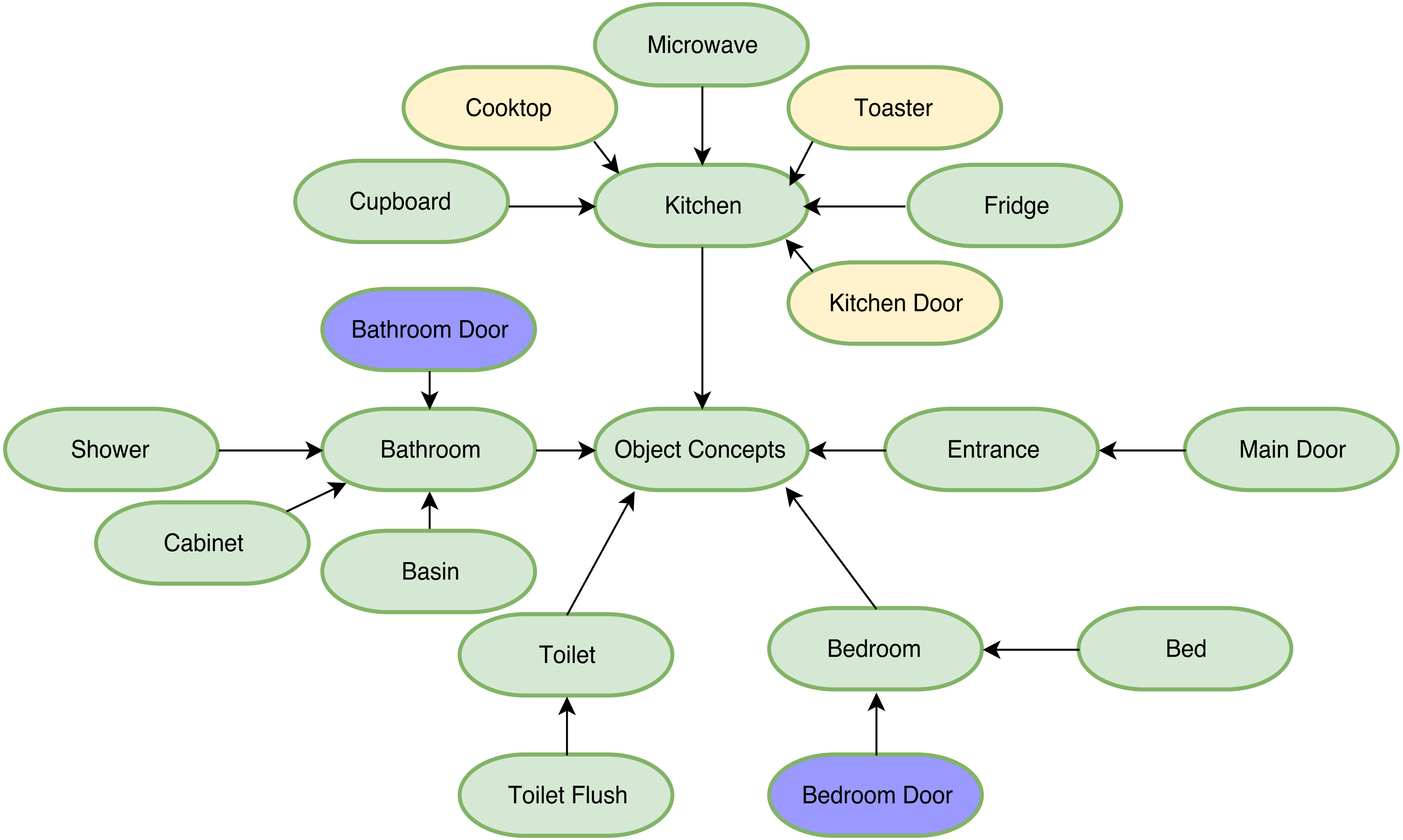


Figure 6.5: Common object concepts for Ordonez Houses A and B. Green for common object concepts, yellow for House A and blue for House B

The activity concepts were modelled accordingly, but due consideration was given to *Make Food* which represents a group of activities from the context description above in subsection 6.2.1. The *Make Food* activity as it implies, corresponds to a group of activities involving making of food and ranges from *Breakfast*, *Lunch*, *Dinner*, *Snack* and even *Drink* which is also seen with the Kasteren and Ordonez homes. Recall, *Make Food* was discovered as an activity with context descriptors and has been classed as a static activity comprising of activities sharing same or similar object interactions but performed at specific times of the day. With regards to the Kasteren and Ordonez dataset, *Breakfast*, *Lunch* and *Dinner* are classed as static activities with super class *Make Food* sharing same or similar context descriptors and also they are performed at specific time of the day. *Drink* and *Snack* have not been limited to time specifics. Extending the activity concepts of the generic ontology in Figure 5.5, *Breakfast*, *Lunch*, *Dinner*, *Drink* and *Snack* are modelled as subclasses of the activity concept *Make Food*. To further enhance shared ontology concepts and reuse, the Kasteren and Ordonez activity concepts are harmonised as illustrated in Figure 6.6 onto the activity ontology to form a set of unified activity concepts. Similar to the object concepts, activity concepts have been colour coded in this unified set of activity concepts with static and dynamic activities as super classes so that the green round rectangle represents common activity concepts, the red rounded rectangle represents activity concepts specific to Kasteren activities and blue specific to Ordonez activities. As part of this proposed hybrid approach, instances and individuals were added to the of the object concepts making the model more expressive (assertions used in populating the ABox) for example instantiating *Microwave* as *Microwave_On* to suggest the state of the object or sensor when in use. The ABox was further populated with assertions using object and data properties as explained in subsection 5.4.2 incorporating the context descriptors for the activity situations preparatory for activity recognition. The modelled activity situations or concepts are then linked to their respective context descriptors as object states through the properties as assertions added to the ABox. Activity recognition is enabled by an object use query to retrieve activity situations based on observed sensor or object data similar to the schema 5.36. The Algorithm 2 implements the recognition process. See Appendix B for the OWL and Ontology implementation of the Kasteren and Ordonez datasets concepts.

To facilitate execution of activity inference, the modelled activity ontology is imported into the TOQL environment [14]. This environment as illustrated in Figure 6.7 is a Java based environment which executes object based queries by mapping the observed objects and its temporal information from the dataset to the closest activity situation in the imported activity ontology through ontological reasoning. The proposed approach was evaluated using the Kasteren and Ordonez sensor data for experiments, thus allowing us to compare

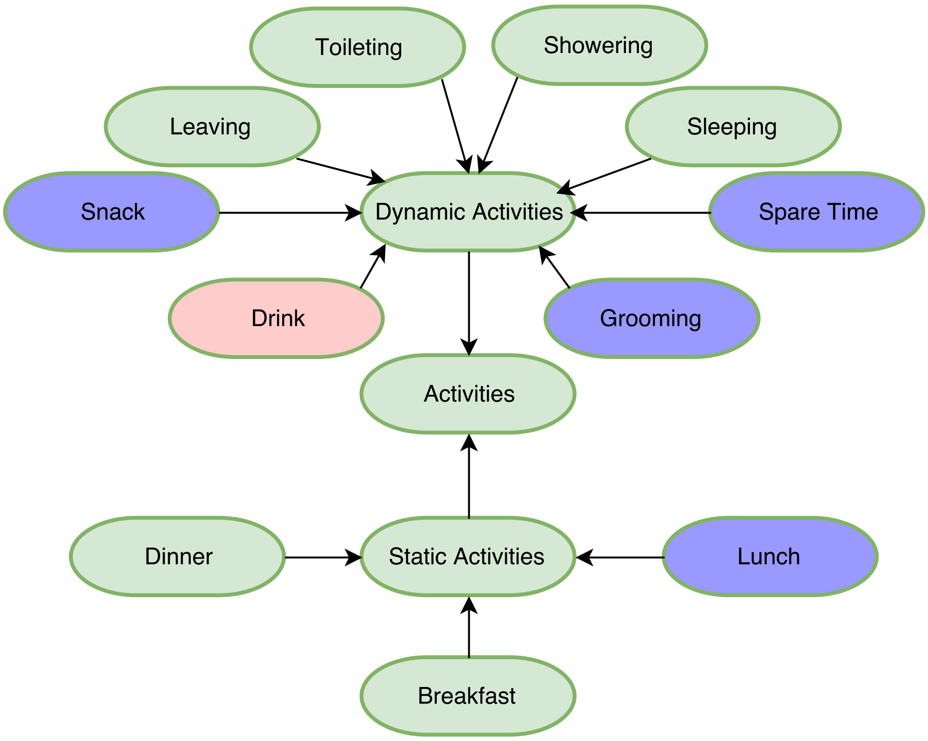


Figure 6.6: Common activity concepts for the Kasteren and Ordonez Houses. Green for common activity concepts, red for Kasteren Houses and blue for Ordonez Houses.

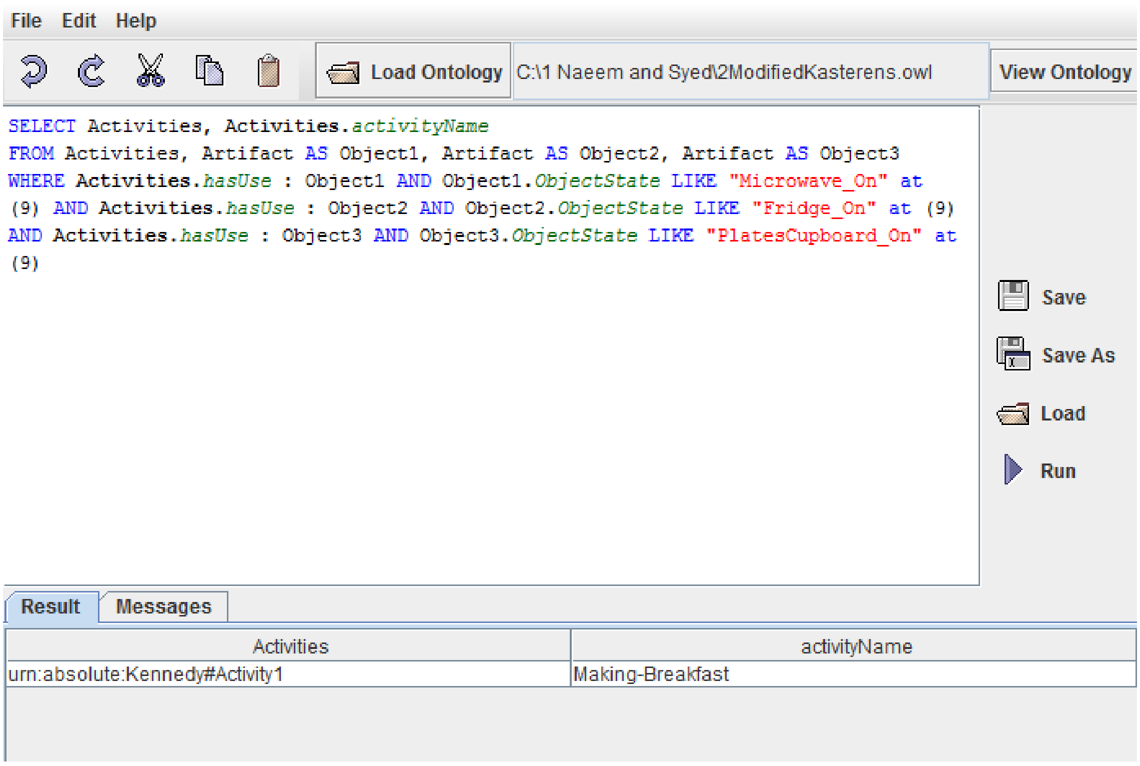


Figure 6.7: Activity recognition by object use query.

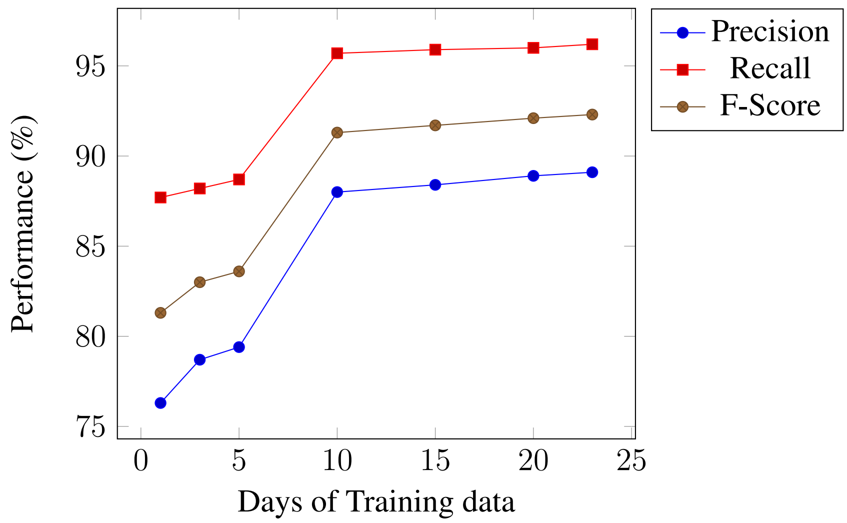


Figure 6.8: Activity recognition performance with different days of training data for Kasteren House A.

the activity situations inferred and recognized by this approach with the ground truths. The criterion for evaluation is to compare recognised activity situations with the ground truth provided with the dataset based on based on the average true positives (TP), false positives (FP) and false negatives (FN) per activity. The results are then further evaluated using precision, recall and F-score.

As mentioned earlier the evaluation methodology of the proposed techniques is based on the 'leave one day out' cross-validation. The justification for this is as illustrated in the Figure 6.8. It illustrates the relationship between the precision, recall and F-Score for the Kasteren House A dataset (See Tables 7.4, 7.5, 7.6, 7.7 and 7.8 in Appendix A for result details of different amounts training data for Kasteren Houses A, B, C and Ordonez Houses A and B). Given the relationships as shown in the curves, there is a marginal increase in terms of amount data requirements for the activity recognition algorithm. However, the performance as illustrated in the Figure 6.8 peaks at 23 days (Out of 24 days) amount of training data for Kasteren House A. With this, thesis bases its validation methodology on the 'leave one day out' cross-validation.

Tables 6.8 and 6.9 presents a summary of recognised activity situations based on the average true positives (TP), false positives (FP) and false negatives (FN) per activity. Leaving, Toileting, Sleeping were recognised with significantly high results for the Kasteren and Ordonez datasets. *Breakfast*, *Dinner*, *Lunch* showed lower performance due to confusions from same and similar object use with *Drink* and *Snack*. The precision, recall and F-score for the activity situations are presented in Figure 6.9 and 6.10 for the datasets.

Activities	True Positives (%) A	B	C	False Positives (%) A	B	C	False Negatives (%) A	B	C
Leaving	100	100	100	0.0	0.0	0.0	0.0	0.0	0.0
Toileting	100	100	100	0.0	0.0	0.0	0.0	0.0	0.0
Showering	95.8	100	100	3.3	0.0	0.0	0.9	0.0	0.0
Sleeping	100	100	100	0.0	0.0	0.0	0.0	0.0	0.0
Breakfast	73.6	74.7	72.8	16.8	15.6	17.5	9.6	9.7	9.7
Dinner	71.2	62.8	69.4	22.9	27.8	23.7	5.9	9.4	6.9
Drink	66.7	55.8	58.8	28.4	30.2	32.1	4.9	14	9.1

Table 6.8: Activity recognition performance for Kasteren A, B and C

Activities	True Positives (%) A	B	False Positives (%) A	B	False Negatives (%) A	B
Leaving	100	100	0.0	0.0	0.0	0.0
Toileting	100	87.6	0.0	8.9	0.0	3.5
Showering	66.7	69.8	23.2	22.1	10.1	8.1
Sleeping	100	100	0.0	0.0	0.0	0.0
Grooming	69.5	69.3	24.9	25.8	5.6	4.9
Spare Time	100	98.3	0.0	1.7	0.0	0.0
Breakfast	82.3	80.7	16.1	13.6	1.6	5.7
Lunch	NA	63.4	NA	25.9	NA	10.7
Dinner	61.9	60.7	28.4	35.8	9.7	3.5
Snack	59.3	59.5	30.7	30.7	10.0	9.8

Table 6.9: Activity recognition performance for Ordonez A and B

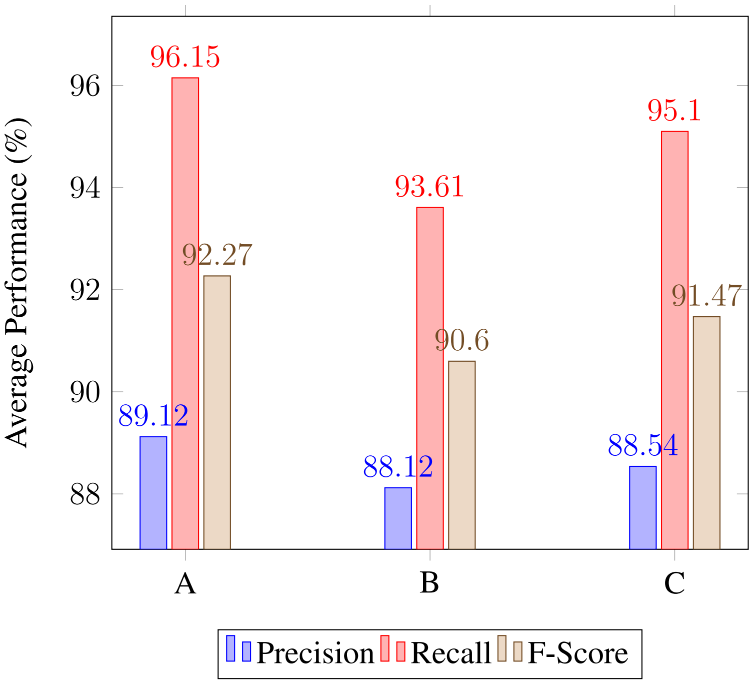


Figure 6.9: Average Precision, Recall and F-Score for Kasteren Houses A, B and C

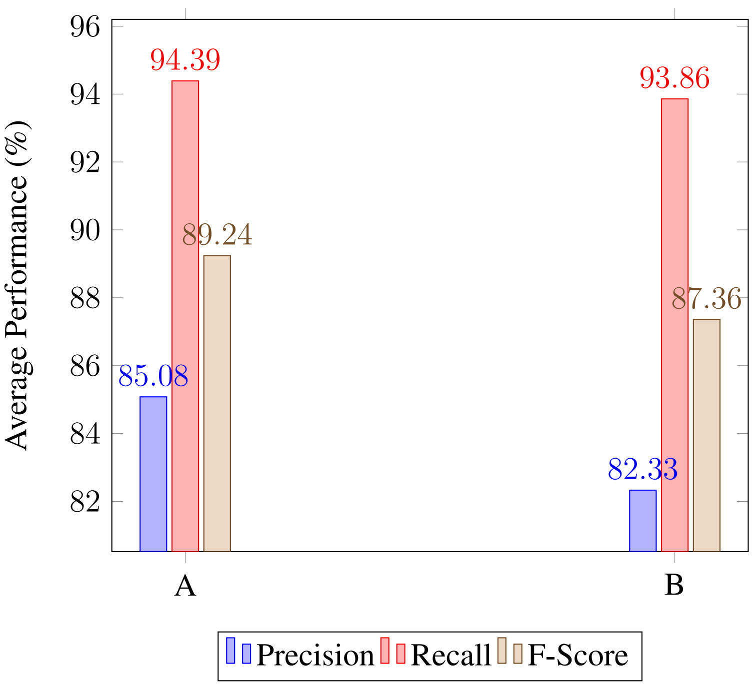


Figure 6.10: Average Precision, Recall and F-Score for Ordonez Houses A and B

Discussion

As earlier mentioned, recognition results for *Leaving*, *Toileting*, *Sleeping* were significantly very good in terms of their performances considering Tables 6.8 and 6.9. This was largely due to the accurate specification of the context descriptors for these activity situations. Also *Showering* in the Kasteren Houses and *Spare Time* in Ordonez performed well. This performance can be attributed to the context descriptors for these activities. The process of discovering likely the object use for routine activities significantly ensured that these activities were associated to the objects used to perform them. Object interactions in the home environment results to sensor firing and are considered as events which are atomic to every activity situation hence the need to use the activity discovery and context description process to aptly map the relevant contexts to describe the activity. This provided the pathway to acquiring the needed knowledge of specific object use or contexts to model in the ontology the specific activity situation which describes it. In addition to this, modelling them as dynamic activities in the ontology also ensured that their temporal attributes are used to help recognise them along their timelines. The performance of *Breakfast*, *Lunch*, *Dinner*, *Drink* and *Snack* are quite significant and encouraging. The activities have shared same and similar object interactions as observed with context description process hence been classed under the super activity *Make Food*. Recall that to further distinguish *Breakfast*, *Lunch* and *Dinner* they were modelled as Static activities given the specific time of the day they are performed. To enhance their recognition, time interval properties and concepts enabled by 4D fluent approach were included. The low performance of these can be attributed to confusions arising from *Drink* and *Snack* for the Kasteren and Ordonez datasets respectively which were classed as dynamic activities. They were often recognised concurrently and led to high false positives in the process. However, the results achieved for them are quite encouraging. Overall, the average precision, recall and F-Score with the datasets as illustrated in Figures 6.9 and 6.10 show impressive performance.

6.2.3 Evaluation of Static and Dynamic Activities

In this experiment, the impact modelling activities was investigated as either static or dynamic activity. Recall in Chapter 5, it was explained that activities can be performed differently, in different ways and times within the 24 hour day path. In some cases, these activities may be performed with same or share similar objects usage in the home environment. *Make Food* is a typical example of the super class activity of *Breakfast*, *Lunch* and *Dinner*. *Breakfast*, *Lunch* and *Dinner* are all examples of different activity concepts which can be performed with same or similar object interactions, but they are all distinct by times of the day they are performed. As sub class activities of the activity class *Make Food*, they

Activities	Precision (%)			Recall (%)			F-Score (%)		
	A	**B**	**C**	**A**	**B**	**C**	**A**	**B**	**C**
Leaving	100	100	100	100	100	100	100	100	100
Toileting	100	100	100	100	100	100	100	100	100
Showering	96.4	100	100	98.2	100	100	97.3	100	100
Sleeping	100	100	100	100	100	100	100	100	100
Breakfast	59.3	54.0	53.0	72.7	71.8	70.3	65.4	61.6	60.4
Dinner	55.3	51.7	53.1	71.2	62.4	59.2	62.3	56.6	56.0
Drink	52.0	44.9	47.1	62.3	48.5	46.6	56.7	46.7	46.9

Table 6.10: Summary of performance with alternate variant of activity ontology for Kasteren Houses

Activities	Precision (%)		Recall (%)		F-Score (%)	
	A	**B**	**A**	**B**	**A**	**B**
Leaving	100	100	100	100	100	100
Toileting	100	89.2	100	94.3	100	91.7
Showering	68.4	70.3	86.1	86.8	76.2	77.7
Sleeping	100	100	100	100	100	100
Grooming	69.0	66.4	89.1	90.8	77.8	76.7
Spare Time	95.1	90.6	98.2	97.4	96.6	93.9
Breakfast	55.7	52.8	78.7	70.7	65.2	60.4
Lunch	49.3	50.4	60.3	52.9	54.4	51.6
Dinner	Na	49.8	Na	56.5	Na	52.9
Snack	48.7	37.2	57.2	58.7	52.6	45.5

Table 6.11: Summary of performance with alternate variant of activity ontology for Ordonez Houses

differ with regards to their respective temporal properties. Whilst they inherit all the properties of *Make Food* by subsumption, they can be easily confused in the recognition process if modelled in the ontology without consideration to their usual times of performance. In the ontology, two classes of activities - dynamic and static activities were created. Activities that are known traditionally to be performed at specific times of the day as static activities (for example *Breakfast*, *Lunch* and *Dinner*) whilst activities that can be performed at any time of the day as dynamic activities (for example *Toileting* or *Showering*). To enhance recognition, 4D fluent approach was applied to integrate temporal information to the activity ontology as explained in subsection 5.5.1. To evaluate the impact modelling activities as either static or dynamic activity, an alternate variant of the activity ontology without the temporal class concepts *Timeslice* and *TimeInterval* was developed. Further, activity recognition was performed using the datasets and compared the performance with results obtained in subsection 6.2.2.

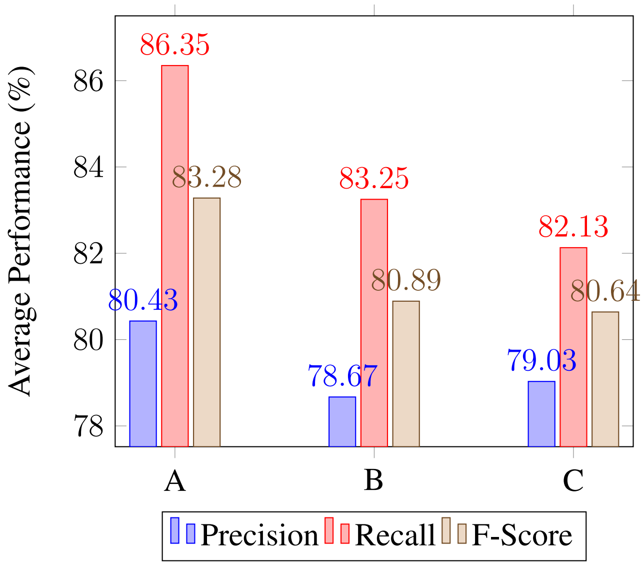


Figure 6.11: Average Precision, Recall and F-Score performance for the alternate variant of activity ontology for Kasteren Houses

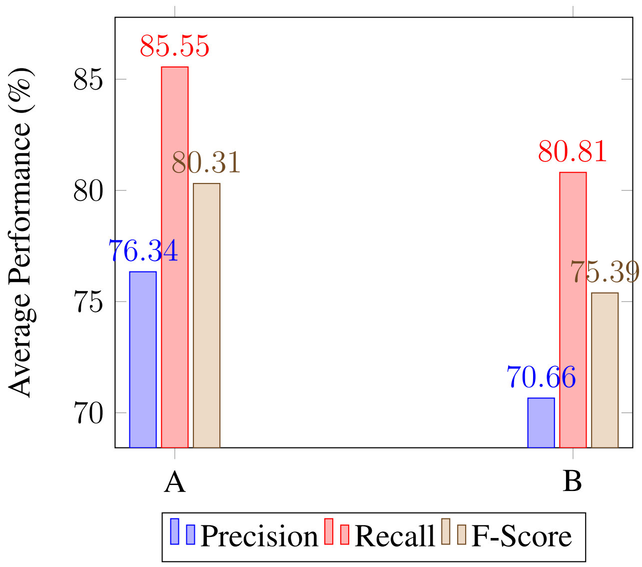


Figure 6.12: Average Precision, Recall and F-Score performance for the alternate variant of activity ontology for Ordonez Houses

Summary of the Precision, Recall and F-Score for the activities using the alternate variant of the activity ontology across folds are as given in Tables 6.10 and 6.11. The results show poorer recognition results (see also Figures 6.11 and 6.12 for the average performance of the Precision, Recall and F-Score using the alternate variant activity ontology) for the activities in comparison with the results using the proposed ontology given in Figures 6.9 and 6.10 Of particular interest are the *Breakfast*, *Lunch* and *Dinner*. Precision results averaged at 55.43% and 54.21% for *Breakfast* in the Kasteren and Ordonez Houses respectively for the alternate variant ontology in comparison with 88.6% and 84.15% for the same houses with the main ontology suggesting the impact of modelling *Breakfast* as static activity. *Snack* and *Drink* in comparison performed with weaker results due to the fact that some of the objects used for these activities are also used to *Breakfast*, *Lunch* and *Dinner*. The performance of the alternate variant ontology generally was due to high false positives with the associated activities. In addition, the use of same and similar objects for raised confusion in the inferencing process leading to multiple activity situations. Therefore, classing these activities as static activities helped to create distinct temporal patterns used to model them in the activity ontology. Hence, the positive impact which led to the performance achieved with the main activity ontology.

6.2.4 Learning Performance of the Approach

In this section, the learning of the proposed activity recognition approach is presented. Activities in the home environment can have different ways of being performed or the patterns of object interactions for activities can differ. A robust activity recognition model should have the ability of recognising activities irrespective of the patterns of object interactions. In this thesis, context description for activities which takes advantage of activity-object discovery to define likely object usage for specific routine activities was proposed. The object contexts from this process describes the activity independent of the sequence or pattern of usage. The model learning of the proposed approach was evaluated in recognising activities traces as they are performed in different ways and patterns using objects. The learning capability at the activity level (the instances of the activities recognised) and at the object level (the similarities of the contexts descriptors independent of order of object usage) are evaluated. The basis for evaluation would be the ground truth. The process of evaluation is to perform activity recognition using the Kasteren and Ordonez datasets and make comparisons with the ground truth. This comparison would involve the number instances of the different activities across folds at the activity level. A good model should be able to return almost the same number of activity traces as in the ground truth. Further, the object usage resulting to the activities recognised with their equivalents in the ground truth are compared

Activities	Jaccard Coeff. (JC)			Dice Coeff. (DC)			Average	
	A	B	C	A	B	C	JC	DC
Leaving	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
Toileting	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
Showering	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
Sleeping	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
Breakfast	0.57	0.67	0.67	0.67	0.75	0.75	0.64	0.72
Dinner	0.67	0.67	0.67	0.75	0.75	0.75	0.67	0.75
Drink	0.5	0.33	0.5	0.67	0.5	0.67	0.44	0.61

Table 6.12: Similarity indices at the object level for the Kasteren Houses

Activities	Jaccard Coeff. (JC)		Dice Coeff. (DC)		Average	
	A	B	A	B	JC	DC
Leaving	1.0	1.0	1.0	1.0	1.0	1.0
Toileting	0.5	0.67	0.5	0.67	0.5	0.67
Showering	0.5	0.67	0.5	0.67	0.5	0.67
Sleeping	1.0	1.0	1.0	1.0	1.0	1.0
Grooming	1.0	0.5	1.0	0.67	0.75	0.84
Spare Time	1.0	1.0	1.0	1.0	1.0	1.0
Breakfast	0.6	0.75	0.67	0.80	0.62	0.78
Lunch	0.6	0.6	.75	0.75	0.6	0.75
Dinner	Na	.6	Na	.75	0.6	.75
Snack	0.5	0.5	0.67	0.67	0.5	0.67

Table 6.13: Similarity indices at the object level for the Ordonez Houses

i.e. if an activity Z is recognised from the interactions of objects x_1, x_2 and x_3 regardless of the order or sequence of usage, these object usage should similarly be observed as the objects resulting to Z in the ground truth. If the upper layer which describes the context have good similarity index, then the lower layer activity ontology should reflect this by having a good measure of correctly recognised activity instances in comparison to the ground truth. The comparisons was based on the Jaccard and Dice similarity coefficients. To evaluate the learning performance at the object level, four samples of each of the activities across the 10 folds were randomly picked and checked for the similarities of context descriptors and the contexts that led to the activities in the ground truth. Tables 6.12 and 6.13 show the results of the similarities. Tables 6.14 and 6.15 below present the learning performance at the activity level.

Discussion At the object level, there is a good measure of similarities between the object usage discovered as context descriptors and that reported in the groundtruth. Of particular interest for the Kasteren Houses are the object used for the activities *Leaving*,

Activities	**Ground Truth**			**Instances Recognised**			**Differences**		
	A	**B**	**C**	**A**	**B**	**C**	**A**	**B**	**C**
Leaving	36	24	47	36	24	47	0	0	0
Toileting	114	27	99	114	27	99	0	0	0
Showering	24	11	14	22	11	14	2	0	0
Sleeping	25	14	19	25	14	19	0	0	0
Breakfast	20	9	18	15	7	14	5	3	4
Dinner	10	6	11	7	4	7	3	2	4
Drink	20	8	10	14	5	6	6	3	4

Table 6.14: Summary of correctly recognised activity instances for Kasteren Houses

Activities	**Ground Truth**		**Instances Recognised**		**Differences**	
	A	**B**	**A**	**B**	**A**	**B**
Leaving	14	38	14	38	0	0
Toileting	44	93	44	79	0	14
Showering	14	11	9	7	5	4
Sleeping	14	29	14	14	0	0
Grooming	51	113	35	76	16	37
Spare Time	77	116	77	113	0	3
Breakfast	14	22	11	17	3	5
Lunch	9	13	5	8	4	8
Dinner	Na	11	Na	6	Na	5
Snack	11	47	6	24	5	23

Table 6.15: Summary of correctly recognised activity instances for Ordonez Houses

Toileting, *Showering* and *Sleeping*. There are marked similarities 1.0 as indices for these activities. Ordonez activities like *Leaving*, *Sleeping*, *Spare Time* also showed very good similarities with the groundtruth. *Breakfast*, *Dinner*, *Lunch*, *Snack* and *Drink* report reduced similarities which can be attributed to multiple object usage which varies depending on what the user feels like using for the activity. But then, these activities share same and similar object usage which can also be linked to this reduced similarities. Considering a threshold of greater than 0.5, we can then say that for all the context descriptors discovered, they are similar to object usage as in the groundtruth. This similarities can also be seen to be linked to the activity recognition results as the share similarities in trend. At the activity level, the number of activity instances recognised as presented in Tables 6.14 and 6.15 are indicative of good learning capability of this proposed approach. *Grooming* and *Snack* of the Ordonez Houses performed worst due to object usage similarities and high false positives. Overall the results are encouraging.

6.2.5 Detection of Activity Boundaries

As part of this hybrid approach, an algorithm for the detection of activity boundaries whilst activity recognition is ongoing was proposed. The aim of the activity boundary algorithm is to signal the end of an activity and the beginning of another activity. It is assumed that objects in particular locations in the home environment are used to perform similar activity and can be grouped into location based subsets i.e. Kitchen based subset of objects for *Making Food* and *Toilet* based subset of objects for *Toileting*. Using Jaccard similarity indexing [72], objects observed consecutively can be checked for similarities to establish whether an activity in continuous or not. Continuity of an activity is dependent on a similarity threshold β for the consecutive objects which must be greater than 0.5 otherwise an activity boundary is detected. The example below illustrates further:

Example: If Fridge and Hall Toilet Door are observed consecutively, β for the two objects is calculated considering the that they belong to two different location based subset of objects. Let A be a set Kitchen objects and B a set of Toilet objects.
A = Microwave, Fridge, Grocery Cupboard, Pans Cupboard, Freezer.
B = Hall Toilet Door, Toilet Flush.
Then Fridge belongs to the Kitchen and Hall Toilet Door belongs to the Toilet. With this, β

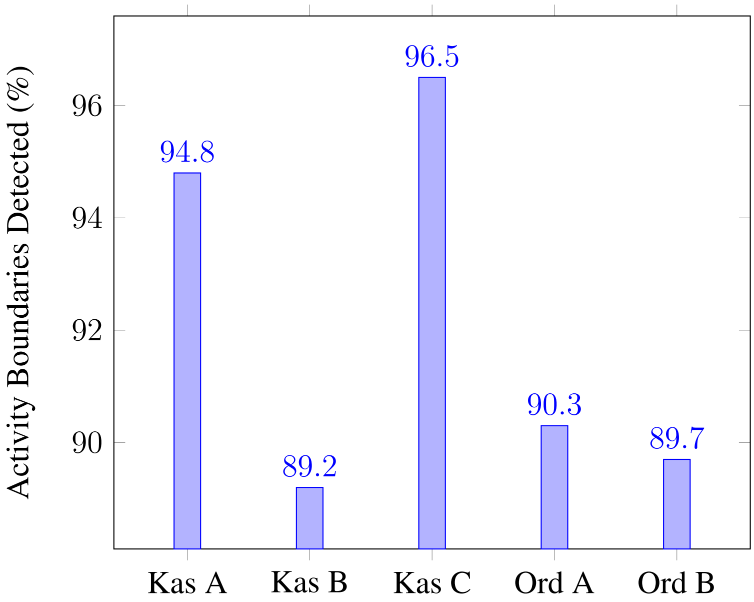


Figure 6.13: Performance of activity boundaries algorithm

is calculated as:

$$\beta = \frac{|A \cap B|}{|A \cup B|}$$
$$= 0.00$$

With β is less than the threshold (0.5), it then implies that an activity boundary is established between Fridge and Hall Toilet Door. An activity ends with Fridge and another begins at this point with Hall Toilet Door.

To evaluate the performance of the activity boundary algorithm, the percentage of accurately recognised activity boundaries (ARAB) is computed using the expression 6.8 where B_a represents number of accurately recognised boundaries and B number represents real boundaries as in the ground truth.

$$ARAB = \sum_{i=1}^{B} \left(\frac{B_a * 100}{B} \right) \tag{6.8}$$

Figure 6.13 illustrates the performance of the activity boundary algorithm for the datasets across folds. The average performance generally was significantly good especially for Kasteren Houses A and C. Boundary recognition performance for Ordonez houses were slightly lower. This was due noise arising from the frequent observation of *Living Door* and *Kitchen Door* which in most cases led to increased false positives for especially for Ordonez House B. In addition, the performance generally is linked to the activity recognition above since the subset suggests closely linked activities.

6.2.6 Evaluation of the Activity Ontology Update

In this experiment, the evaluation of the changes made to the activity ontology through ontology update is presented. Recall in subsection 5.7, the activity ontology update using the SPARQL update language was presented [46]. It was explained that it is logical for activity recognition models to have the capabilities to make ontology updates arising from the changes in the choice of objects usage for activities. With SPARQL update language triggered by the algorithm 4, changes and modifications which may arise due to object usage for the specific routine activities are facilitated. For this purpose, changes to the context descriptors for the Kasteren House A activities *Sleeping*, *Toileting* and *Breakfast* and performed activity recognition as explained in subsection 6.2.2 were made. It is expected that the activity recognition results should be the same for the activities for which the updates were made in comparison to results achieved before the updates. Hitherto, *Sleeping* has context descriptors *Hall Bedroom Door*, *Toileting* has *Hall Toilet Door* and *Toilet Flush* as context descriptors. The context descriptors for *Breakfast* includes *Fridge*, *Plates Cupboard*, *Cups Cupboard*, *Groceries Cupboard*, *Microwave* and *Freezer*. If some changes are made to add the context descriptors *Pillow* to *Sleeping*, *Toilet Roll* to *Toileting*, *Cooktop* to *Breakfast* and remove *Microwave* for *Breakfast*. The Table 6.16 then presents the context descriptors for the activities with their respective mapping objects and data property attributes.

Activity	Context Descriptor	Mapping Property attribute and Data
Sleeping	Hall Bedroom Door	`adl:hasUse 'HallBedroomDoor'`
	Pillow	`adl:hasUse 'Pillow'`
Toileting	Hall Toilet Door	`adl:hasUse 'ToiletDoor'`
	Toilet Flush	`adl:hasUse 'ToiletFlush'`
	Toilet Roll	`adl:hasUse 'ToiletRoll'`
Breakfast	Fridge	`adl:hasUse 'Fridge'`
	Plates Cupboard	`adl:hasUse 'PlatesCupboard'`
	Cups Cupboard	`adl:hasUse 'CupsCupboard'`
	Groceries Cupboard	`adl:hasUse 'GroceriesCupboard'`
	Freezer	`adl:hasUse 'Freezer'`
	Cook top	`adl:hasUse 'Cooktop'`

Table 6.16: Context descriptors and the mapping attributes

The activity ontology was updated with the schemas 6.9, 6.10 and 6.11 respectively for *Sleeping*, *Toileting* and *Breakfast* so that changes as illustrated in Figure 6.14 is a snippets of the changes made. Activity recognition results based on these modifications and updates in comparison with the results obtained before modifications and updates are as illustrated in Figure 6.15.

```
Prefix   adl < Graph />
INSERT  {
          ?Activity  adl:hasUse  'HallBedroomDoor'
}                                                        (6.9)
WHERE  {
          ?Activity  adl:hasUse  'Pillow'
}
```

```
Prefix   adl < Graph />
INSERT  {
          ?Activity  adl:hasUse  'ToiletRoll'
}
WHERE  {                                                 (6.10)
          ?Activity  adl:hasUse  'ToiletDoor'
          ?Activity  adl:hasUse  'ToiletFlush'
}
```

```
Prefix   adl < Graph />
DELETE  {
          ?Activity  adl:hasUse  'Microwave'
}
INSERT  {
          ?Activity  adl:hasUse  'Cooktop'
}                                                        (6.11)
WHERE  {
          ?Activity  adl:hasUse  'Fridge'.
          ?Activity  adl:hasUse  'Cupboard'.
          ?Activity  adl:hasUse  'Toaster'
}
```

```
<!-- urn:absolute:Kennedy\#Activity5TimeSlice3 -->

<owl:NamedIndividual rdf:about="urn:absolute:Kennedy\#Activity5TimeSlice3">
            <rdf:type rdf:resource="urn:absolute:Kennedy\#Slice3"/>
            <rdf:type rdf:resource="urn:absolute:Kennedy\#TimeSlice"/>
            <hasUse rdf:resource="urn:absolute:Kennedy\#ToiletDoorTimeSlice3"/>
            <hasUse rdf:resource="urn:absolute:Kennedy\#ToiletFlushTimeSlice3"/>
            <hasUse rdf:resource="urn:absolute:Kennedy\#ToiletRollTimeSlice3"/>
            <tsTimeInterval rdf:resource="urn:absolute:Kennedy\#TimeInterval3"/>
            <tsTimeSliceOf rdf:resource="urn:absolute:Kennedy\#Activity5"/>
</owl:NamedIndividual>

<!-- urn:absolute:Kennedy\#Activity6 -->
<owl:NamedIndividual rdf:about="urn:absolute:Kennedy\#Activity6">
            <rdf:type rdf:resource="urn:absolute:Kennedy\#Activities"/>
            <rdf:type rdf:resource="urn:absolute:Kennedy\#Showering"/>
            <activityNamerdf:datatype="http://www.w3.org/2001/XMLSchema\#string">Use-
Shower</activityName>
</owl:NamedIndividual>
```

Figure 6.14: OWL syntax snippet after inserting *ToiletRoll* for the activity *Toileting*

With regards to the results obtained for the modified activity ontology, the F-Score for the activities *Sleeping*, *Toileting* and *Breakfast* across folds were 100%, 100% and 92.18% respectively. There were no significant changes in comparison for *Sleeping* and *Toileting* as the context descriptors for these activities are not shared with other activities. *Breakfast* showed improved recognition in comparison because the addition of *Cooktop* as a context descriptor helped to minimise false positives. It also reduced the confusion of *Breakfast* with *Drink* and *Dinner* which do not have *Cooktop* as shared context descriptors. The performance also shows that inserting and deleting of context descriptors through this update operation validates the process and demonstrates the easiness to making changes and modifications in comparison to editing the entire activity ontology using editor like Protégé.

6.2.7 Comparison With Other Recognition Approaches

This thesis made comparisons with the results reported by Kasteren et al [133], Ordonez et al [99], Ye [145], Riboni et al [110] and Kun et al [53]. In addition, comparisons were made following the technique proposed by Okeyo et al [97]. The techniques reported by authors have been all based on the Kasteren House A dataset using the "leave one day out" cross-validation except Ye [145] who primarily used 10-fold validation (Ye also included results

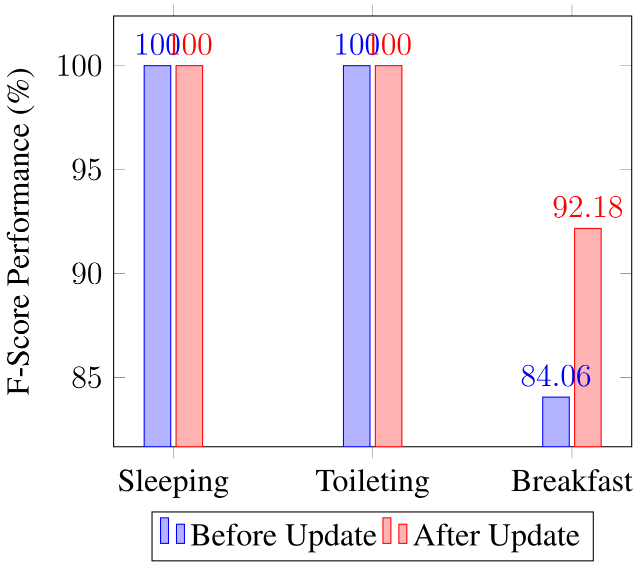


Figure 6.15: Comparison of F-Score performance before and after update

based on a "leave one day out" cross-validation in her report). Kasteren et al [133] reported class accuracy of 49.2% (for Hidden Markov Model) and 44.6% (for Conditional Random Field) which in their definition is same as average recall per class. In comparison with their work, this thesis achieved a higher average recall of 96.2%. Ordonez et al reported F-Score results of 75% and 76% for Hybrid Multi-Layer Perceptrons and Support Vector Machine respectively - both lower in comparison with the F-Score of 92.3% achieved in this thesis. The proposed technique performed better than Kasteren and Ordonez due to its ability to incorporate temporal attributes to sensor and objects through timeslices and intervals enhancing the activity recognition performance. In addition to the 10 fold cross validation used, Ye also included in her report Recall and F-Score of 78.9% and 84.4% respectively for the Kasteren House A dataset using 'leave one day out' cross-validation'. This is also lower than the 96.6% and 92.3% for the Recall and F-Score respectively reported in this thesis. Ye's technique gives a set of probabilities which corresponds to the likelihood of the activity situations given a set of observed sensors. This may have accounted for the lower performance in comparison as it falls short of the context descriptors as presented in this thesis which is specific to the routine activities. In addition to the comparisons above, we considered other hybrid approaches. Riboni et al [110] and Okeyo et al [97] and Kun et al [53] all proposed hybrid and ontology-based activity recognition techniques. Although Okeyo et al [97] did not make their report based on the Kasteren dataset, we assembled their technique in almost the same way as explained by the authors. Comparing our approach with these hybrid approaches using the Kasteren House A dataset, the recognition results presented in this thesis surpassed the others as seen in Table 6.17. Riboni et al

[110] and Okeyo et al [97] reported F-Score of 89.7% and 82.8% respectively. Although Riboni et al [110] followed a 'leave one day out' validation, the ontology concepts had no temporal attribute associations. So with the Kasteren House A, activity recognition led to high false positives for activities like Breakfast, Dinner and Drink which shared same and similar object concepts. Okeyo et al [97] models activities based on generic assumptions of object use. This, however, does not put into considerations the context describing activity situations, hence high false positives for activities especially if they share same or similar object usage. The object use specification as context description routine activities, fine grain activity modelling and temporal association stood as key features which led to minimal false positives with the approach proposed in this thesis over the approaches by Riboni et al [110] and Okeyo et al [97]. Kun et al [53] technique is composed of an infrastructure included a search engine module which performs internet information retrieval for its ontology module for activity recognition which is connected to a 3 layer HMM module to provide assistance. The information and performance results as presented by Kun [53] were minimal making it difficult for the technique to be scrutinized. Their report as suggested led to high false positive due to the results from the internet search which multiple per object search. With the Kasteren House A dataset, there were multiple search results which may have impacted on the ontology concepts and retrieved information.

Technique		Leaving	Toileting	Showering	Sleeping	Breakfast	Dinner	Drink	Average
Riboni et al [110]	Precision	98	76.8	90.7	93.9	77.6	89.5	69.6	85.2
	Recall	100	93.5	96.7	98.9	92.4	95.6	88.9	95.1
	F-Measure	98.9	84.4	93.6	96.4	84.4	92.5	78	89.7
Okeyo et al [97]	Precision	100	100	93.9	100	58.6	55.4	54.1	80.3
	Recall	100	100	95.1	100	72.3	70.8	62.7	85.8
	F-Measure	100	100	94.5	100	64.7	62.2	58.1	82.8
Ye et al [145]	Precision	92	98.6	82.5	91.7	96.4	92.2	81.6	90.7
	Recall	58.5	85.9	94	78.6	91.4	77.7	66	78.9
	F-Measure	71.5	91.8	87.9	84.6	93.8	84.3	72.9	84.4
This thesis	Precision	100	100	96.7	100	81.4	75.7	70.1	89.1
	Recall	100	100	99.1	100	88.5	92.3	93.2	96.2
	F-Measure	100	100	97.9	100	84.8	83.2	80	92.3

Table 6.17: Comparison of our approach with other hybrid approaches

6.3 Summary and Conclusion

The hybrid activity recognition approach enhanced using topic model proposed in this thesis provides the basis to learn and recognise activities. Experiments carried out using the Kasteren and Ordonez datasets. The performance of the context description process was assessed. The performance of the approach to recognise activities was also evaluated. This thesis also evaluated the impact of modelling activities as static and dynamic activities. Further, it evaluated the learning capability of the proposed approach. Finally, it evaluated the performance of the activity boundary detection algorithm. In addition to the experiments, this thesis compared the results to the results published using the same dataset in other literature. Based on the experiments and evaluations, the benefits and limitations of the proposed approach are discussed below:

- **Assessment of the activity-object use and context description process**: As part of the hybrid approach, acquiring knowledge of object usage by the object discovery and context descriptions for the activity situations was proposed. To assess the performance of the process, similarity assessments between the sets of context description make up for the activities was carried out. The results clearly suggests no similarities between the activity sets. The set of context descriptions are unique for the activity situation using the Jaccard and Dice similarity coefficients. However, activities like *Breakfast*, *Lunch* and *Dinner* sharing same or similar object use are considered as activity situations which can be made distinct by modelling them as static activities in the ontology. The main benefit of this process is its ability to discover unique object use as context descriptor for activity situations. Limitations may arise for other similar activity situations like *Drink* and *Snack* as observed with the datasets.

- **Performance of the activity recognition process**: Experiments carried out on the datasets suggest good recognition performance for activities. Although the performance was encouraging for most activities, recognition was confused for activities sharing same and similar object use. Notably in this case was *Drink* and *Snack* which was modelled as dynamic activities. Given the general performance of this activity recognition process as illustrated with Figure 6.9 and 6.10, the activity recognition process on the average is comparable.

- **Impact of the Static and Dynamic Activities**: The impact of modelling activities as static and dynamic activities was assessed. The comparison with the alternate variant of the activity ontology suggested a weaker performance. This was largely due to the activities like *Breakfast*, *Dinner*, *Lunch*, *Drink* and *Snack* reported for the datasets in table 6.10 and 6.11. These activities traditionally are performed with same and similar object leading to high false positives and recognition confusions. It is obvious from Tables 6.8 and 6.9 modelling the activities as static and dynamic activity minimised false positive and confusion from the activities across folds.

- **Model Learning Performance**: The aim of this evaluation was to assess the model learning ability. The similarities of contexts descriptions at the object level and at the activity level were assessed. From the results, the contexts which led to activity situations in the ground truth are similar to the contexts descriptors discovered, hence the result achieved at the activity level. Almost the same number of activity traces for the datasets in comparison with the ground truth suggesting good and significant learning were obtained. Although, results were weaker for the Ordonez dataset set due to noise and similarities in contexts for some activities. The results at the object and activity level can be seen to be directly linked to the activity recognition performance with regards to Tables 6.8 and 6.9.

- **Activity Boundary Detection**: The performance of this proposed algorithm. Activity boundaries detected suggests good performance were assessed. It also found out that the performance is closely linked to the activities recognised. However, the results achieved are significant and encouraging.

- **Activity Ontology Update**: Finally, the performance of the object use changes made through activity ontology update were assessed. The results obtained suggests good performance for the updates made. The performance of *Breakfast* improved due to reduced false positives and minimal confusions with *Dinner* and *Drink*. The change of inserting *Cooktop* to *Breakfast* further enhanced *Breakfast* as a distinct activity.

With the experiments, assessment and evaluation using publicly available datasets, it can be said that *i)* The process of activity-object use and context description of activity situation provides accurately the needed object and activity concepts for the ontology modelling process. *ii)* Modelling activities as static and dynamic activities helps to improve activity recognition especially for activities with same and similar object interactions. *iii)* Given the results from the activity recognition process in comparison with other results published using the same datasets, it concludes that it is significantly good and encouraging. The experimental and evaluation process using these datasets suggests that the features, components and the entire activity recognition process have been fully verified.

Chapter 7

CONCLUSION AND FUTURE WORK

The motivation of work in this thesis was to provide the basis to acquire knowledge of object usage for specific routine activities to support an ontology-driven activity recognition instead of developing activity ontologies from assumptions and every day common knowledge of object use for activities. The outcome is an enhanced knowledge-driven activity recognition approach, which utilizes topic modelling to discover the context descriptors for specific activity situations. This work has extended the traditional ontology driven activity recognition with improvements in the way ontology concepts in this domain are acquired. To achieve this, Latent Dirichlet Allocation LDA topic models were applied to generatively discover activities-object distributions. In addition to this, an algorithm was developed that used the activity-object distributions to define the context descriptors for the specific routine activities. A description logic DL formalism for ontology and knowledge representations was followed to build a knowledge base for the activities and the respective context descriptions as concepts. TBox and ABox were also populated to allow for activity information retrieval enabled by the reasoners. The proposed hybrid approach also included the development of an activity recognition algorithm and an activity boundary detection algorithm. As part of the performance and validation process, the work in this thesis was compared to the approaches of Riboni et al [110], Kasteren et al [132], Ordonez et al [99], Okeyo et al [97], Kun et al [53] and Ye [145] who have published work evaluated on the same datasets. Riboni et al [110] in their work modelled the object use for the activity situations using ontological techniques. Although they extended the basic ontological techniques with the addition of temporal reasoning which outperformed the results from Kasteren et al [132], they were limited with the mention on how the knowledge of object use for the activity situations were acquired to develop the activity ontology. Ye [145] used context lattice exhaustively models all possible conjunctions of sensor firings. They labelled the sensor observations with the associated probability of each of the situation's occurrence. The work in this thesis differs from theirs with the context descriptions for specific activity situations modelled on to an ontology knowledge base system to support activity inference retrieval based on observed object use information. Activities are eventually recognised by querying the knowledge base using the observed object use information. The following sections provide a summary of the research contributions and a discussion on future work. The work reported in this thesis is an extension to our previous work reported in Ihianle et al [68] with the use of LDA and inclusion of the activity ontology to introduce semantic clarity, enhance expressiveness and improve activity recognition whilst retaining the ability to handle noisy situations through the LDA. Our work in Ihianle et al [68] used the topic model PLSA to discover and recognise activities, but it still remains that the process is limited and lacks expressiveness of the activities recognised for the end user. As part of the extension, we determined the number of activities automatically using silhouette analysis through K-Means clustering and used the LDA instead of the PLSA to discover the activity-object use distributions. However, the approach we used to determine the context descriptors for specific routine activities provides an alternate method to acquire knowledge of object use for activities and context descriptions for activities.

7.1 Contributions

The work in this thesis proposed an enhanced knowledge driven activity recognition approach to acquire the knowledge of object use as context descriptors for the activity situations. Below is a summary the contributions:

- **The State of the art of the Recognition of Activities of Daily Living in the Home**; Environment. In Chapter 2 we presented a review of research efforts in the area of Ambient Intelligence, Context Awareness Pervasive computing with regards to activity recognition. We also considered a broad overview of these attempts highlighting data and knowledge driven approaches. From these discussion we made distinction of the features of the approaches and identified emerging approaches towards activity recognition. We also identified limitations and possibly how the approaches can be complementary. We concluded by pointing out that activity recognition as an emerging area of technology has yet to be fully develop.
- **A Context description of object use for specific routine activities.**; In Chapters 4, we presented the proposed hybrid approach and methodology and the approach to acquire knowledge context description for activity ontology respectively. The context description module is an extension to the traditional ontology driven activity recognition approach. It serves as a novel approach and an alternate solution to the

traditional method to acquiring knowledge of object usage which hitherto has been by assumption and the generic knowledge of every day object usage in the home environment. We applied the topic model LDA to discover activity-object distribution and developed a context description algorithm to satisfactorily assign objects as context descriptors for specific routine activities which were then designed, developed and modelled in the activity ontology as concepts and instances etc.

- **An approach to recognise Static and Dynamic Activities.**: In Chapter 5, we presented an approach to model static and dynamic activity situations by combining ontology formalism and 4D fluent approach. We realise that some are carried out at particular times of the day and the others are not constrained to be performed at any particular time of the day. To achieve this model, we applied the 4D fluent approach to create time intervals of the day and made the activities to range onto these time intervals as concept through an interval property. We also created timeslices as concepts and made the objects to have properties ranging to these timeslices. With these concept and properties, activities like *Breakfast*, *Lunch* and *Dinner* sharing same or similar object interactions were modelled to belong to different time intervals to make them distinctive in the activity recognition process.

- **Modelling fine grain activities situations by enabling the atomic events in precedence.**: In Chapter 5, we presented fine grain activities recognising that activities are a result of atomic events occurring in patterns and orders. The pattern and orders differ and in some cases the patterns determine the activity situation. To achieve precedence, we applied the rule of transitivity enable by the object property `hasLastObject` so that the order of object evolution in the emergence of an activity would be followed. In addition to this we also extended the object and activity concepts to have instances and individuals asserted using object and data properties to populate the ABox. Activity recognition is achieved by object use query of the instances.

- **Algorithm for Activity Boundary Detection**: In Chapter 5, we proposed an algorithm to recognise activity boundary. We introduced location concept for objects within the same location of a home environment. We were of the assumption that if two consecutively observed objects or sensor belonged to same location in the home environment it suggests the persistence of an activity. We applied similarity to objects in the same location to indicate this persistence and used dissimilarity of consecutively observed objects also based on location to suggest an activity boundary i.e. the end of an activity and the beginning of an activity. Similarity must be above a threshold value while dissimilarity below the threshold value.

- **Ontology update for object use**: Also in Chapter 5, we presented activity ontology object use update without the process of editing the entire ontology. Ontologies have been known to be static, so to give room for changes in the activities and object usage we introduce the ontology update. We use the insert and delete data for SPARQL update language to update the ontology when there is a change in the activity situations or with object usage.

- **Evaluation and Validation of the Proposed Hybrid Approach.**: In Chapter 6 we performed evaluation and validation experiments on the proposed hybrid approach using publicly available datasets. The evaluation includes assessment, of the similarities of the context descriptors of activities, of the activity recognition performance, recognition performance of static and dynamic activities, performance of the boundary recognition algorithm, and comparison with the results published using other techniques on the datasets.

With these contributions, the principal aim "*To design and implement a hybrid activity recognition approach that recognises routine activity situations as events from sensor datasets by the accurate specification of the object use as the context describing the activities*" as highlighted in chapter 1 of this thesis has been fully achieved.

7.2 Future Work

Given that the performance and evaluation metrics indicated good performance for the proposed approach, we make good to say that it is not perfect and all problems regarding activity recognition have not been taken into account. In the course of this research, we identified the following issues and opportunities:

- **Concurrent and Interleaved**: Activities considered in this thesis are interleaved activities. We have not considered recognition of activities concurrently evolving. We have not also considered non-interleaved activities. Concurrent and non-interleaved activities present complex situations involving the start of another activity whilst there is an ongoing activity. This would require extensive model in the ontology multiple instances and the use of precedence property to assert the activities as they evolve. Future work should investigate extending our proposed work to include concurrent and non-interleaved activities.

- **Multi-individual Occupancy**: This approach has been designed and developed on the assumption of a single occupancy home environment. In real world situations, homes are occupied by multiple users or in the case of the elderly and cognitively impaired you have cases when the carers are with the elderly or in a multi occupancy care resident. Future work can extend the context descriptions and ontology model to include multiple actors and individuals. The sensor network could be extended to include heterogeneous sensing systems. Individual sensor firings can be distinguished by making the individuals wear identifying sensors in the case of wearable sensing protocols or sensor protocol pairings when object are used on the home environment. It could also include mobile phone as a distinguishing mechanism.

- **All-In-One Unified Model**: We have developed this approach on a multi-platform. The context description module has been developed and implemented in a Matlab environment, the activity ontology has been designed and developed in the Protégé environment which is different from the former. In a realistic situation, the approach should be in a single platform thus allowing for almost a real time activity recognition process. Future improvements on this would involve developing a system or an algorithm capable of performing context description and ontology model in the same platform.

- **Automatic Model Adaptation**: The approach as it is currently manually updates the ontology based on changes in the reported activities in comparison to the ground truth. This process is not only slow but could be prone to error. An improvement in the future could be to bootstrap the context description module to report immediately changes in the context describing the activity situations. The process would iteratively provide the needed update to automatically adapt the activity ontology for the needed activity recognition. Finally this hybrid approach has been evaluated using dataset collected in a controlled home environment. The home setting falls short of real world situations. Activities have been performed to prescription. The activities in number are limited as well the limited number of objects tagged with sensors. Uncertainties and incompleteness which characterises human life have not been put into consideration. Given this future work could involve deploying extensive sensor network in real home environments and the experiment replayed.

7.3 Concluding Remarks

This thesis has provided significant contributions to the emerging research area of activity recognition. A review of the state of the art was made and areas of opportunities highlighted. To address some of the opportunities we proposed this hybrid approach as an extension most especially to the traditional ontology-driven activity recognition. We proposed the acquiring knowledge object use for specific routine activities, a process we termed context description of the activity situations. Use modelled these contexts as ontological concepts to allow for a more expressive way of activity recognition from object use. Although the performance and evaluation process suggests good performance, there are opportunities to improve on this proposed work in the future to the benefit of the elderly and cognitively impaired.

Appendix A

Activities	Context Descriptors
Leaving	Front Door.
Toileting	Hall Toilet Door, Toilet Flush.
Showering	Hall Bathroom Door.
Sleeping	Hall Bedroom Door.
Make Food	Fridge, Plates Cupboard, Cups Cupboard, Groceries Cupboard, Microwave, Freezer.
Make Drink	Fridge.

Table 7.1: Activity Concepts and the discovered context descriptors for Kasterens House B

Activities	Context Descriptors
Leaving	Front Door.
Toileting	Hall Toilet Door, Toilet Flush.
Showering	Hall Bathroom Door.
Sleeping	Hall Bedroom Door.
Make Food	Fridge, Plates Cupboard, Cups Cupboard, Groceries Cupboard, Microwave, Freezer.
Make Drink	Fridge.

Table 7.2: Activity Concepts and the discovered context descriptors for Kasteren House C

Activities	Context Descriptors
Leaving	Main Door.
Toileting	Toilet, Basin.
Showering	Shower.
Sleeping	Bed.
Make Food	Cupboard, Fridge, Microwave, Toaster.
Spare Time	Seat.
Grooming	Basin, Cabinet.

Table 7.3: Activity Concepts and the discovered context descriptors for Ordonez House B

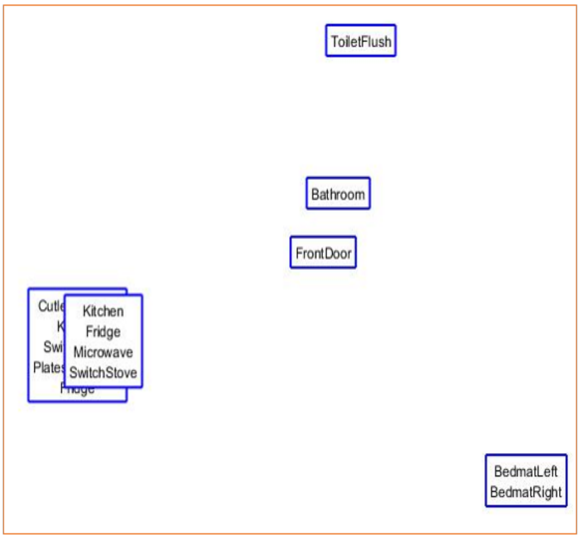
(a) House A

(b) House B

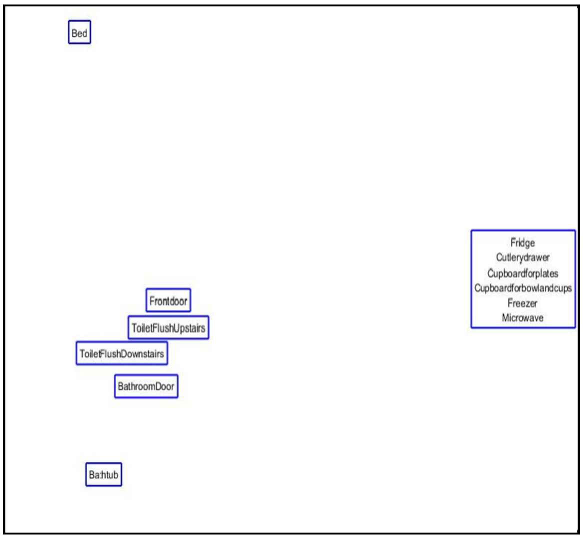
(c) House C, First floor

(d) House C, Second floor

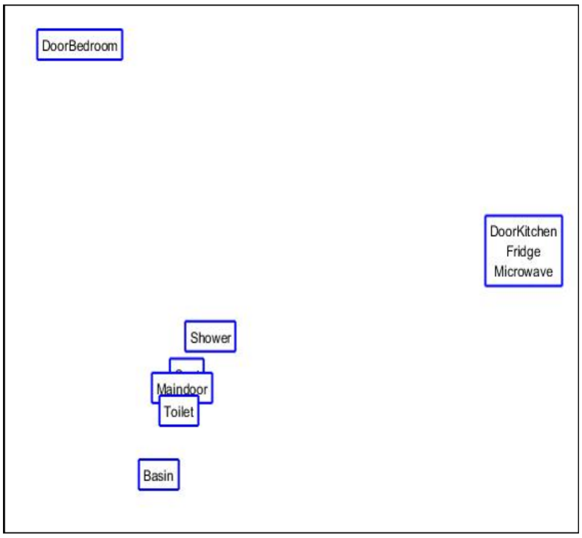
Figure 7.1: Floorplan of houses A, B and C, the red boxes represent wireless sensor nodes



(a) Kasterens B



(b) Kasterens B



(c) Ordonez B

Figure 7.2: Visualisation of LDA generated Object classifications for Kasterens B, Kasterens C and Ordonez B

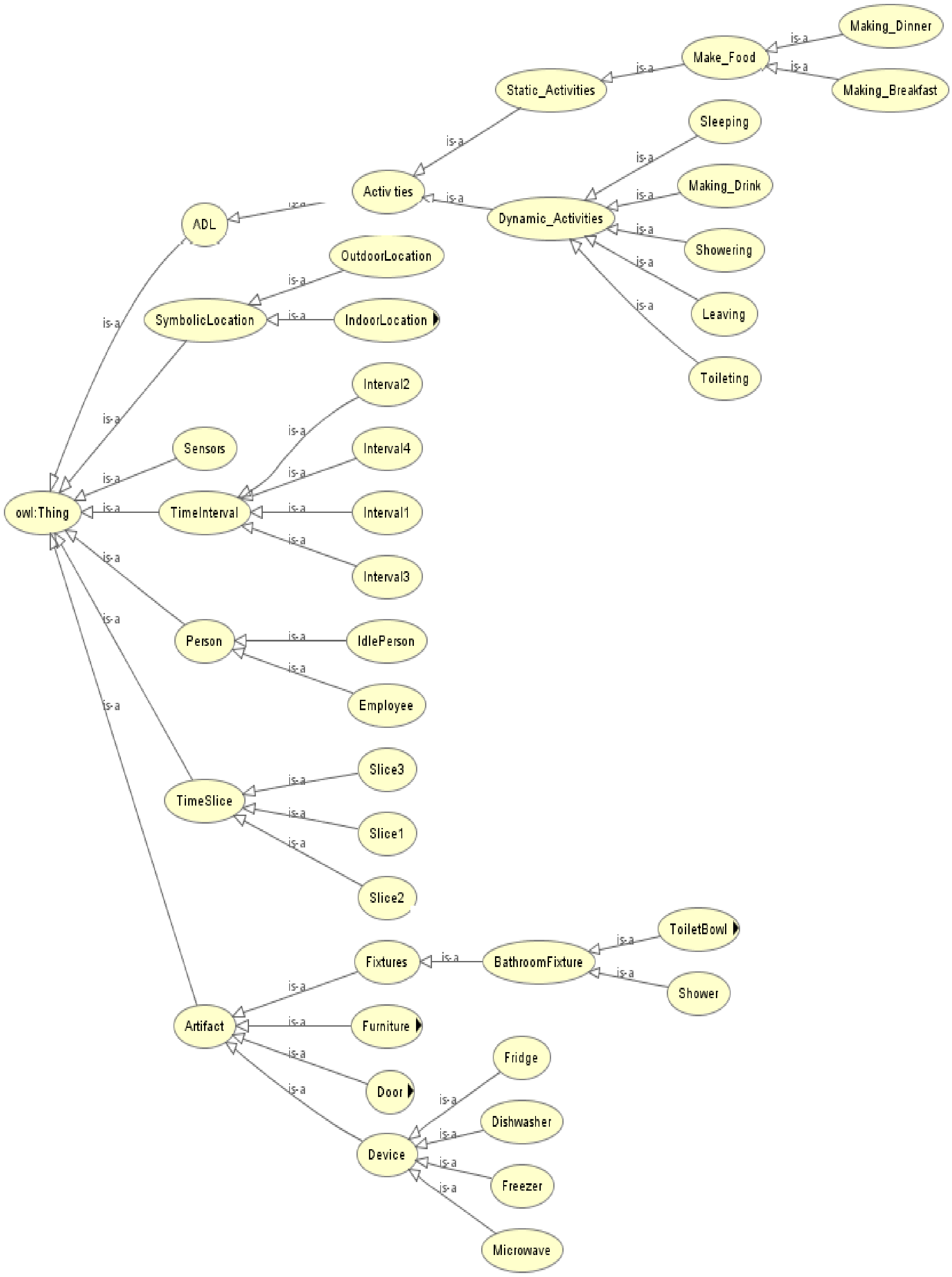


Figure 7.3: Class Concepts of the Kasteren House Ontology

Figure 7.4: Class Concepts of the Ordonez House Ontology

Days		Leaving	Toileting	Showering	Sleeping	Breakfast	Dinner	Drink	Average
1	Precision	97.6	94.3	85.8	83.4	57.5	60.7	55.0	76.3
	Recall	98.7	97.9	91.8	95.6	79.4	72.9	77.6	87.7
	F-Measure	98.1	96.0	88.7	89.1	66.7	66.3	64.4	81.3
3	Precision	98.2	95.2	88.6	84.6	64.6	63.8	56.2	78.7
	Recall	99.1	99.4	93.4	96.2	78.3	73.0	78.0	88.2
	F-Measure	98.7	97.2	90.9	90.0	70.8	68.1	65.3	83.0
5	Precision	98.7	95.8	89.8	84.0	65.9	64.9	56.6	79.4
	Recall	100	99.4	95.4	97.1	78.1	73.5	77.2	88.7
	F-Measure	99.3	97.5	92.5	90.1	71.5	68.9	65.3	83.6
10	Precision	99.3	100	96.0	100	79.6	73.2	68.2	88.0
	Recall	100	100	98.33	100	87.7	91.6	92.3	95.7
	F-Measure	99.1	100	97.1	100	83.4	81.3	78.4	91.3
15	Precision	100	100	96.5	100	80.6	73.4	68.6	88.4
	Recall	100	100	98.8	100	88.4	91.6	92.3	95.9
	F-Measure	100	100	97.6	100	84.3	81.5	78.7	91.7
20	Precision	100	100	96.4	100	80.6	75.6	69.8	88.9
	Recall	100	100	98.7	100	88.4	92.3	92.8	96.0
	F-Measure	100	100	97.5	100	84.3	83.1	79.6	92.1
23	Precision	100	100	96.7	100	81.4	75.7	70.1	89.1
	Recall	100	100	99.1	100	88.5	92.3	93.2	96.2
	F-Measure	100	100	97.9	100	84.8	83.2	80	92.3

Table 7.4: Precision, Recall and F-Score results with different amount of training data for Kasteren House A

Days		Leaving	Toileting	Showering	Sleeping	Breakfast	Dinner	Drink	Average
1	Precision	89.5	84.8	76.1	88.7	58.5	52.0	49.5	71.3
	Recall	97.2	95.6	93.8	96.1	76.8	70.3	71.5	85.9
	F-Measure	93.2	89.9	83.0	92.2	66.4	59.7	58.5	77.6
3	Precision	92.5	91.4	88.4	94.5	66.6	57.1	53.5	77.7
	Recall	98.5	97.5	95.9	97.6	82.5	79.4	74.9	89.5
	F-Measure	95.5	94.3	92.0	96.0	73.7	66.4	62.4	82.9
5	Precision	96.4	97.2	93.5	98.0	74.7	62.3	58.6	82.9
	Recall	100	99.4	98.4	99.3	86.4	84.1	77.8	92.2
	F-Measure	98.2	98.3	95.9	98.6	80.2	71.6	66.8	87.1
10	Precision	100	99.6	95.1	100	76.4	67.1	61.5	85.7
	Recall	100	100	99.2	100	87.1	84.8	78.9	92.9
	F-Measure	100	99.8	97.1	100	81.3	74.9	69.1	88.9
13	Precision	100	100	100	100	82.7	69.3	64.9	88.1
	Recall	100	100	100	100	88.5	86.9	79.9	93.6
	F-Measure	100	100	100	100	85.5	77.1	71.6	90.6

Table 7.5: Precision, Recall and F-Score results with different amount of training data for Kasteren House B

Days		Leaving	Toileting	Showering	Sleeping	Breakfast	Dinner	Drink	Average
1	Precision	89.5	82.9	77.5	84.2	57.6	52.4	46.3	70.1
	Recall	96.1	95.1	93.3	96.2	77.6	74.1	75.9	86.9
	F-Measure	92.7	88.6	84.7	89.8	66.1	61.4	57.5	77.3
3	Precision	93.6	87.9	84.8	89.6	61.2	56.5	48.8	74.6
	Recall	97.5	97.2	95.3	98.6	79.0	77.2	79	89.1
	F-Measure	95.5	92.3	89.7	93.8	69.0	65.2	60	80.8
5	Precision	96.1	90.5	89.0	94.7	66.2	60.1	53.2	78.5
	Recall	98.2	98.6	96.1	98.8	81.1	79.4	84.1	90.9
	F-Measure	97.1	94.4	92.4	96.7	72.9	68.4	65.2	83.9
10	Precision	98.4	93.5	93.4	99.1	71.3	66.1	55.1	82.4
	Recall	100	96.7	98.2	100	84.2	81.5	86.4	92.4
	F-Measure	99.2	96.5	95.7	99.5	77.2	73.0	67.3	86.9
15	Precision	100	96.8	100	100	78.3	73.9	59.6	86.9
	Recall	100	99.7	100	100	86.5	90.1	88.3	94.9
	F-Measure	100	98.2	100	100	82.2	81.2	71.2	90.4
18	Precision	100	100	100	100	80.6	74.5	64.7	88.5
	Recall	100	100	100	100	88.2	90.9	86.6	95.1
	F-Measure	100	100	100	100	84.3	81.9	74.1	91.5

Table 7.6: Precision, Recall and F-Score results with different amount of training data for Kasteren House C

Days		Leaving	Toileting	Showering	Sleeping	Grooming	Spare Time	Breakfast	Lunch	Snack	Average
1	Precision	85.8	84.3	52.2	83.7	50.5	82.4	57.9	48.9	48.1	65.9
	Recall	94.2	93.9	76.9	96.8	78.9	95.4	83.8	76.1	68.8	84.9
	F-Measure	89.8	88.8	62.6	89.7	61.6	88.5	68.5	59.6	56.6	73.9
3	Precision	92.44	90.1	60.2	89.5	55.5	87.1	66.6	53.9	51.5	71.9
	Recall	96.33	95.8	80.1	97.5	84.4	97.1	91.4	80.1	72.0	88.3
	F-Measure	94.3	92.9	68.7	93.3	66.9	91.8	77.1	64.4	60.0	78.8
5	Precision	95.6	92.8	65.0	94.2	63.9	95.1	72.6	58.3	56.9	77.2
	Recall	98.7	98.2	83.2	99.7	88.9	98.4	97.1	81.3	76.5	91.3
	F-Measure	97.2	95.4	73.0	96.9	74.4	96.7	83.1	67.9	65.2	83.3
10	Precision	97.8	95.5	68.9	98.2	67.9	98.7	81.4	64.1	59.9	81.4
	Recall	100	99.7	84.8	100	90.3	100	97.8	84.8	77.9	92.8
	F-Measure	98.9	97.6	76.1	99.1	77.5	99.3	88.8	73.0	67.7	86.4
13	Precision	100	100	74.2	100	73.6	100	83.6	68.5	65.8	85.1
	Recall	100	100	86.8	100	92.5	100	98.1	86.5	85.6	94.4
	F-Measure	100	100	80.0	100	82.0	100	90.3	76.4	74.5	89.2

Table 7.7: Precision, Recall and F-Score results with different amount of training data for Ordonez House A

Days		Leaving	Toileting	Showering	Sleeping	Grooming	Spare Time	Breakfast	Lunch	Dinner	Snack	Average
1	Precision	75.6	68.4	49.6	68.0	47.6	69.8	45.4	45.7	44.3	43.1	55.7
	Recall	89.5	82.0	66.9	88.2	69.9	85.9	71.3	71.5	72.0	70.8	76.8
	F-Measure	82.0	74.6	56.9	76.8	56.6	76.9	55.5	55.8	54.9	53.6	64.4
3	Precision	77.9	73.6	50.1	75.6	51.3	74.5	49.4	48.1	47.7	44.9	59.3
	Recall	90.1	85.6	66.7	89.8	71.0	89.4	74.7	74.4	71.6	71.0	78.4
	F-Measure	83.6	79.1	57.2	82.1	59.5	81.3	59.4	58.4	57.2	55.1	67.3
5	Precision	87.2	80.6	57.4	81.0	56.9	82.5	62.5	53.7	51.5	49.9	66.3
	Recall	96.2	90.9	78.3	90.5	78.5	92.4	79.7	77.3	78.4	69.7	83.2
	F-Measure	91.5	85.4	66.3	85.5	66.0	87.2	70.0	63.3	62.1	58.2	73.6
10	Precision	93.6	85.6	64.6	90.9	64.7	89.6	76.5	60.9	55.8	56.8	73.9
	Recall	98.7	92.8	84.3	98.1	83.7	97.2	86.1	81.2	84.0	79.2	88.5
	F-Measure	96.1	89.1	73.1	94.4	73.0	93.2	81.1	69.6	67.2	66.1	80.3
15	Precision	100	89.8	71.8	97.5	69	94.7	81.1	68.1	59.9	62.2	79.4
	Recall	100	94.5	88.9	100	91.3	97.8	91.0	84.2	93.5	84.1	92.5
	F-Measure	100	92.1	79.4	98.7	78.6	96.2	85.8	75.3	73	71.5	85.1
20	Precision	100	90.8	75.9	100	72.9	98.3	85.6	70.9	62.9	66.0	82.3
	Recall	100	96.2	89.6	100	93.4	100	93.4	85.6	94.5	85.9	93.9
	F-Measure	100	93.4	82.2	100	81.9	99.1	89.3	77.6	75.5	74.6	87.4

Table 7.8: Precision, Recall and F-Score results with different amount of training data for Ordonez House B

Bibliography

[1] ACTIVE and ASSISTED LIVING PROGRAMME. Myguardian project. http://myguardian-project.eu/. Accessed: 2018-05-20.

[2] Active and Assisted Living Programme. Aalp objectives. http://www.aal-europe.eu/, 2015. Accessed: 2017-05-28.

[3] J. K. Aggarwal and M. S. Ryoo. Human activity analysis: A review. *ACM Computing Surveys, Press*, 2011.

[4] U. Akdemir, P. Turaga, and R. Chellappa. An ontology based approach for activity recognition from video. *in Proceedings of the 16th ACM international conference on Multimedia*, 12(4):709–712, 2008.

[5] A. Ankoleka, M. Burstein, J. R. Hobbs, O. Lassila, D. Martin, and D. McDermott. Daml-s: Web service description for the semantic web. *in Proceedings of the Tenth International Conference on Principles of Knowledge Representation and Reasoning KR'06*, 2342:348–363, 2002.

[6] K. Annemarie, H. Huosheng, and P. Matthew. Sysiass—an intelligent powered wheelchair. *in First International Conference on Systems and Computer Science*, 2012.

[7] G. Antoniou, E. Franconi, and F. v. Harmelen. Introduction to semantic web ontology languages. *Lecture Notes in Computer Science*, 3564:1–21, 2005.

[8] Alzheimer's Association. Stages of alzheimer's. http://www.alz.org/alzheimers_disease_stages_of_alzheimers.asp, 2015. Accessed: 2017-05-28.

[9] J. C. Augusto. Ambient intelligence: Basic concepts and applications. *First International Conference on Software and Data Technologies*, 2006.

[10] G. Azkune. Learning for dynamic and personalised knowledge-based activity model. *Ph.D Thesis - Universidad de Deusto*, 2015.

[11] F. Baader, I. Horrocks, and U. Sattler. Description logics. *International Handbooks on Information Systems*, pages 21–43, 2009.

[12] F. Baader, D. L. McGuinness, D Nardi, and P. F. Patel-Schneider. Description logic handbook: Theory, implementation and applications, 2003.

[13] L. Bao and S. S. Intille. Activity recognition from user-annotated acceleration data. *in 2nd International Conference Pervasive Computing (PERVASIVE 2004)*, 2004.

[14] E. Baratis, E. G. M. Petrakis, S. Batsakis, N. Maris, and N. Papadakis. Toql: Temporal ontology querying language. *in Advances in Spatial and Temporal Databases*, 2009.

[15] S. Batsakis and E. G. M. Petrakis. Sowl: A framework for handling spatiotemporal information in owl 2.0. *in 5th International Symposium, RuleML*, 2011.

[16] S. Bechhofer, F. V. Harmelen, J. Hendler, I. Horrocks, D. L. McGuiness, P. F. Patel-Schneider, and L. A. Stein. Owl web ontology language. :`https://www.w3.org/TR/owl-ref/`. Accessed: 2016-10-16.

[17] T. Berners-Lee, J. Hendler, and O. Lassila. The semantic web, 2001.

[18] D. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *Machine Learning Research*, 2003.

[19] B. Bouchard, S. S. Giroux, and A. Bouzouane. A smart home agent for plan recognition of cognitively-impaired patients. *Journal of Computers*, 1:10, 2006.

[20] A.K. Bourke and G.M. Lyons. A threshold-based fall-detection algorithm using a bi-axial gyroscope sensor. *Medical Engineering Physics*, 30:84–90, 2008.

[21] O. Brdiczka, J. L. Crowley, and P. Reignier. Learning situation models in a smart home. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 39:56–63, 2009.

[22] M. Brown. Supporting user mobility. *in International Federation for Information Processing*, 1996.

[23] M. A. Kang C. Roussey, F. Pinet and O. Corcho. An introduction to ontologies and ontology engineering. *in Ontologies in Urban Development Projects, Advanced Information and Knowledge Processing*, pages 9–387, 2011.

[24] D. Cao, O. T. Masoud, D. Boley, and N. Papanikolopoulos. Human motion recognition using support vector machines. *Computer Vision and Image Understanding*, 113:1064–1075, 2009.

[25] A. Cesta, G. Cortellessa, and M. V. Giuliani. The robocare assistive home robot: Environment, features and evaluation. the robocare technical reports. *National Research Council of Italy*, 2006.

[26] D. Chen, J. Yang, and H.D. Wactlar. Towards automatic analysis of social interaction patterns in a nursing home environment from video. *in Proceedings of the International Workshop Multimedia Information Retrieval*, pages 283–290, 2004.

[27] L. Chen, J. Hoey, C. Nugent, D. Cook, and Z. Yu. Sensor-based activity recognition. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Review*, 42:790–808, 2012.

[28] L. Chen and C. D. Nugent. Ontology-based activity recognition in intelligent pervasive environments. *International Journal of Web Information Systems*, 5(4):410–430, 2009.

[29] L. Chen and C.D. Nugent. A logical framework for behaviour reasoning and assistance in a smart home. *International Journal of Assistive Robotics and Mechatronics*, 9(4):20–34, 2008.

[30] 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*, 2011.

[31] T. Choudhury, M. Philipose, D. Wyatt, and J. Lester. Towards activity databases: Using sensors and statistical models to summarize people's lives. *IEEE Data Engineering Bulletin*, 29:49–56, 2006.

[32] R. Cooper, G. Grindle, and J. Vazquez. Personal mobility and manipulation appliance—design, development, and initial testing. *in IEEE*, 2012.

[33] J. Cooperstock, K. Tanikoshi, G. Beirne, T. Narine, and W. Buxton. Evolution of a reactive environment. *in CHI '95*, 1995.

[34] C. K. M. Crutzen. Invisibility and the meaning of ambient intelligence. *International Review of Information Ethics*, 2006.

[35] B. d. Ruyter and E. Pelgrim. Ambient assisted-living research in carelab. *Interactions*, 14(07):30–33, 2007.

[36] A. K. Dey and G. D. Abowd. Towards a better understanding of context and context-awareness. *in Workshop on the What, Who, Where, When and How of Context-Awareness affiliated with the CHI 2000 Conference on Human Factors in Computer Systems*, 2000.

[37] A. K. Dey, G. D. Abowd, and A. Wood. Cyberdesk: A framework for providing self-integrating context-aware services. *in Knowledge-Based Systems*, 1999.

[38] F. Doctor, H. Hagras, and V. Callaghan. Ambient assisted-living research in carelab. *Interactions*, 35(1):55–65, 2005.

[39] S. Elrod, R. Costanza G. Hall, M. Dixon, and J. des Rivieres. Responsive office environments. *in CACM*, 1993.

[40] D. Fensel, F. v. Harmelen, I. Horrocks, D. L.McGuinness, and P. F. PatelSchneider. Oil: an ontology infrastructure for the semantic web. *IEEE Intelligent Systems*, 16(2):348–363, 2001.

[41] T. Finni, M. Hu, P. Kettunen, T. Vilavuo, and S. Cheng. Measurement of emg activity with textile electrodes embedded into clothing. *Physiological Measurement*, 28:1405–1419, 2007.

[42] Office for National Statistics (ONS). National population projections: 2014-based statistical bulletin. `https://www.ons.gov.uk/`, 2015. Accessed: 2017-05-28.

[43] A.R.J Francois, R. Nevatia, J. Hobbs, and R.C. Bolles. Verl: An ontology framework for representing and annotating video events. *in IEEE MultiMedia*, 12(4):76–86, 2005.

[44] E. Friedman-Hill. Jess, the rule engine for the javatm platform. `:http://herzberg.ca.sandia.gov/`. Accessed: 2017-05-28.

[45] A. Garate, N. Herrasti, and A. Lopez. Genio—an ambient intelligence in home automation and entertainment environment. *in Joint Conference on Smart Objects and Ambient Intelligence Innovative Context-Aware Services: Usages and Technologies*, 2005.

[46] P. Gearon, A. Passant, and A. Polleres. Sparql 1.1 update. :`https://www.w3.org/TR/sparql11-update/`, 2013. Accessed: 2016-10-16.

[47] T. George, G. Shalu, and K. Sivanandan. Sensing, processing and application of emg signals for hal (hybrid assistive limb). *in International Conference on Sustainable Energy and Intelligent Systems (SEISCON 2011)*, 2011.

[48] B. Georis, M. Maziere, F. Bremond, and M. Thonnat. Towards automatic analysis of social interaction patterns in a nursing home environment from video. *in 2nd Workshop Intell. Distributed Surveillance System*, pages 46–50, 2004.

[49] B. C. Grau, I. Horrocks, B. Motik, B. Parsia, P. Patel-Schneider, and U. Sattler. Owl 2: The next step for owl. *Web Semantics: Science, Services and Agents on the World Wide Web*, 6:309–322, 2008.

[50] T. L. Griffiths and M. Steyvers. Finding scientific topics. *in Proceedings of the National Academy of Sciences*, 2004.

[51] European Commission'sInformation Society Technologies Advisory Group. Scenarios for ambient intelligence in 2010 final report. `https://www.semanticscholar.org/paper/Scenarios-for-Ambient-Intelligence-in-2010-Final-Ducatel-Bogdanow 479a7a314f47c30cfafaeaaa984cddb20114c149`, 2010. Accessed: 2017-05-28.

[52] T. R. Gruber. A translation approach to portable ontology specifications. *Knowledge Acquisition*, 5:119–220, 1993.

[53] Kun Guo, Yonghua Li, Yueming Lu, Xiang Sun, Siye Wang, and Ruohan Cao. An activity recognition-assistance algorithm based on hybrid semantic model in smart home. *International Journal of Distributed Sensor Networks*, 2016.

[54] K. J. Heilman and S. W. Porges. Accuracy of the lifeshirt r (vivometrics) in the detection of cardiac rhythms. *Biological Psychology*, 75:300–305, 2007.

[55] S. Helal, W. Mann, H. El-Zabadani, J. King, Y. Kaddoura, and E. Jansen. Gator tech smart house: a programmable pervasive space. *Computer*, 38:50–60, 2005.

[56] P. Hitzler, M. Krotzsch, and S. Rudolph. Foundations of semantic web technologies, development, and initial testing, 2009.

[57] J. Hobbs, R. Nevatia, and B. Bolles. An ontology for video event representation. *in IEEE Workshop on Event Detection and Recognition*, 2004.

[58] T. Hofmann. Probabilistic latent semantic analysis. *in Proceedings of Uncertainty in Artificial Intelligence, UAI'99*, 1999.

[59] S. Hongeng and R. Nevatia. Large-scale event detection using semi-hidden markov models. *in Ninth IEEE International Conference on Computer Vision*, 2003.

[60] I. Horrocks, O. Kutz, and U. Sattler. The even more irresistible sroiq. *in Proceedings of the Tenth International Conference on Principles of Knowledge Representation and Reasoning KR'06*, pages 57–671, 2006.

[61] I. Horrocks, P. F. Patel-Schneider, S. Bechhofer, and D. Tsarkov. Owl rules: A proposal and prototype implementation. *Web Semantics*, 3:23–40, 2005.

[62] I. Horrocks, P. F. Patel-Schneider, H. Boley, S. Tabet, B. Grosof, and M. Dean. Swrl: A semantic web rule language combining owl and ruleml. :`https://www.w3.org/Submission/SWRL/`. Accessed: 2016-10-16.

[63] R. Hull, P. Neaves, and Bedford-Roberts. Towards situated computing. *in 1st International Symposium on Wearable Computers*, 1997.

[64] T. Huynh, U. Blanke, and B. Schiele. Scalable recognition of daily activities with wearable sensors. *3rd International Conference on Location and Context-Awareness LoCA'07*, 2007.

[65] T. Huynh, M. Fritz, and B. Schiele. Discovery of activity patterns using topic models. *in 10th international conference on Ubiquitous computing UbiComp '08*, 2008.

[66] T. Huynh and B. Schiele. Unsupervised discovery of structure in activity data using multiple eigenspaces. *2nd International Workshop on Location and Context-Awareness (LoCA 2006)*, 2006.

[67] iCare. icare project: Intelligent care guidance and learning services platform. `http://www.aal-europe.eu/projects/icarer/`, 2015. Accessed: 2017-05-15.

[68] I. K. Ihianle and A. Tawil U. Naeem. Recognition of activities of daily living from topic model. *Procedia Computer Science*, 2016.

[69] S. S. Intille, K. Larson, E. M. Tapia, J. S. Beaudin, P. Kaushik, J. Nawyn, and R. Rockinson. Using a live-in laboratory for ubiquitous computing research. *in 4th International Conference on Pervasive Computing (PERVASIVE 2006)*, 1998.

[70] J., T. Bai, and H. Kautz. Improving the recognition of interleaved activities. *in 10th Inter. Conf. on Ubiquitous Computing (UbiComp 2008)*, 2008.

[71] Y. Ayatsuka J. Rekimoto and K. Hayashi. Augment-able reality: Situated communication through physical and digital spaces. *in 2nd International Symposium on Wearable Computers*, 1998.

[72] A. K. Jain and R. C. Dubes. Algorithms for clustering data. *Prentice-Hall*, 1988.

[73] F. Katayoun and D. Gatica-Perez. Discovering routines from large-scale human locations using probabilistic topic models. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2(3), 2011.

[74] H. Kautz. A formal theory of plan recognition and its implementation, reasoning about plans. *Morgan Kaufmann, San Mateo*, 12(4):69–125, 1991.

[75] W. A. Khan, M. Hussain, M. Afzal, M. B. Amin, and S. Lee. Healthcare standards based sensory data exchange for home healthcare monitoring system. *in Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, 2012.

[76] C. D. Kidd, R. Orr, G. D. Abowd, C. G. Atkeson, I. A. Essa, B. MacIntyre, E. Mynatt, T. E. Starner, and W. Newstetter. The aware home: A living laboratory for ubiquitous computing research. *in Second International Workshop, CoBuild'99*, 1999.

[77] M. Krotzsch, F. Simancık, and I. Horrocks. A description logic primer. `:https://arxiv.org/pdf/1201.4089.pdf`. Accessed: 2016-10-15.

[78] D. Kulic, W. Takano, and Y. Nakamura. Representability of human motions by factorial hidden markov models. *in International Conference on Intelligent Robots and Systems(IROS'07)*, 2007.

[79] K. Lagus, T. Honkela, and T. Kohonen S. Kaski. Self-organizing maps of document collections: A new approach to interactive exploration. *in Proceedings of the Second International Conference on Knowledge Discovery and Data Mining KDD '96*, 1996.

[80] P. Langley, W. Iba, and K. Thompson. An analysis of bayesian classifiers. *in 10th National Conference on Artificial Intelligence*, 1992.

[81] F. Latfi and B. Lefebvre. Ontology-based management of the telehealth smart home, dedicated to elderly in loss of cognitive autonomy. *in CEUR Workshop Proceedings*, 2007.

[82] S.-W. Lee and K. Mase. Activity and location recognition using wearable sensors. *in IEEE Pervasive Computing*, 2002.

[83] C. Leonardi, C. Mennecozzi, F. Pianesi, and M. Zancanaro. Supporting older adults social network: the design of e-inclusion communication services. *in Sixth Interaction Conference of International Society for Gerontechnology*, 2008.

[84] J. Liao, Y. Bi, and C. Nugent. Weight factor algorithms for activity recognition in lattice-based sensor fusion. *in 9th International Conference on Information Technology and Applications in Biomedicine (ITAB 2009)*, 2009.

[85] L. Liao, D. Fox D. J. Patterson, and H. Kautz. Learning and inferring transporation routines. *Artificial Intelligence*, 171(5-6):311–331, 2007.

[86] L. Liao, D. Fox, and H. Kautz. Extracting places and activities from gps traces using hierarchical conditional random field. *International Journal of Robotics*, 26(1):119–134, 2007.

[87] W. Lui, X. Li, and D. Huang. A survey on context awareness. *in International Conference on Computer Science and Service System (CSSS 2011)*, 2011.

[88] D. Lymberopoulos, A. Bamis, T. Teixeira, and A. Savvides. Behaviorscope: Real-time remote human monitoring using sensor networks. *in International Conference on Information Processing in Sensor Networks (IPSN 2008)*, 2008.

[89] J. Lilius M. Saleemi, N. Diaz and I. Porres. A framework for context-aware applications for smart spaces. *in The 4th Conference on Smart Spaces*, 2004.

[90] E. Maier and G. Kempter. Aladin—a magic lamp for the elderly? *in Handbook of Ambient Intelligence and Smart Environment*, 38:1201–1227, 2010.

[91] Alex Mathew. The need of ambient intelligence technology in various applications will boost its future demand. `https://www.coherentnews.com/the-need-of-ambient-intelligence-technology-in-various-applicatio` 2018. Accessed: 2018-05-20.

[92] B McBride. Jena: a semantic web toolkit. *IEEE Internet Computing*, 6, 2002.

[93] Marvin Minsky. Frame-system theory. *Thinking: Readings in Cognitive Science*, 1977.

[94] Usman Naeem and John Bigham. Activity recognition in the home using a hierarchal framework with object usage data. *Journal of Ambient Intelligence and Smart Environments*, 1(4):335–350, 2009.

[95] H. Nakashima, H. Aghajan, and J. Augusto. Handbook of ambient intelligence and smart environment. *Amsterdam: IOS Press*, 2010.

[96] G. Okeyo, L. Chen, H. Wang, and R. Sterritt. Dynamic sensor data segmentation for real-time activity recognition. *Pervasive and Mobile Computing*, 2012.

[97] G. Okeyo, L. Chen, H. Wang, and R. Sterritt. A hybrid ontological and temporal approach for composite activity modelling. *in IEEE 11th International Conference on Trust, Security and Privacy in Computing and Communications*, 2012.

[98] P. Oliveira, J. Cunha, and J. Coelho. Design of pid controllers using the particle swarm algorithm. *in Proceedings of the twenty-first IASTED international conference: Modelling, Identification, and Control (MIC 2002)*, 2002.

[99] J. Ordonez, P. Toledo, and A. Sanchis. Activity recognition using hybrid generative/discriminative models on home environment using binary sensors. *Sensors*, 13(5):5460–5477, 2013.

[100] J. Oresko, J. Cheng Z. Jin, S. Huang, Y. Sun, H. Duschl, and A. C. Cheng. A wearable smartphone-based platform for realtime cardiovascular disease detection via electrocardiogram processing. *IEEE Transaction on Information Technologies in Biomed*, 14(3):734–740, 2010.

[101] Y. Ou, L. Cao, and C. Zhang. Adaptive anomaly detection of coupled activity sequences. *IEEE Intelligent Informatics Bulletin*, 10:12–16, 2009.

[102] P. Palmes, H. K. Pung, T. Gu, W. W. Xue, and S. Chen. Object relevance weight pattern mining for activity recognition and segmentation. *Pervasive and Mobile Computing*, 6(1):43–57, 2010.

[103] J. Pascoe. Adding generic contextual capabilities to wearable computers. *in 2nd International Symposium on Wearable Computers*, 1998.

[104] D. J. Patterson, D. Fox, H. Kautz, and M. Philipose. Fine-grained activity recognition by aggregating abstract object usage. *in Ninth IEEE International Symposium on Wearable Computers*, 2005.

[105] D. J. Patterson, D. Fox L. Liao, and H. Kautz. Inferring high-level behavior from low-level sensors. *5th Conference on Ubiquitous Computing (UbiComp)*, 2003.

[106] E. Prud'hommeaux and A. Seaborne. Sparql query language for rdf. :`https://www.w3.org/TR/2008/RECrdf-sparql-query-20080115/`. Accessed: 2016-10-15.

[107] H. Qian, Y. Mao, W. Xiang, and Z. Wang. Recognition of human activities using svm multi-class classification. *Pattern Recognition Letters*, 31(2):100–111, 2010.

[108] P. Lord R. Stevens, C. Wroe and C. Goble. Ontologies in bioinformatics. *International Handbooks on Information Systems*, pages 635–657, 2004.

[109] P. Rashidi and D. 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.

[110] D. Riboni, L. Pareschi, L. Radaelli, and C. Bettini. Is ontology-based activity recognition really effective? *in IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops)*, 2011.

[111] Daniele Riboni and Claudio Bettini. Cosar: hybrid reasoning for context-aware activity recognition. *Personal and Ubiquitous Computing*, 15:271–289, 2011.

[112] N. D. Rodriguez, M. P. Cuellar, J. Lilius, and M. D. Calvo-Flores. A survey on ontologies for human behavior recognition. *ACM Computing Surveys*, 2014.

[113] M. Rosen-Zvi, T. Griffiths, M. Steyvers, and P. Smyth. The author-topic model for authors and documents. *in Proceedings of the 20th conference on Uncertainty in artificial intelligence UAI '04*, pages 487–494, 2004.

[114] P. J. Rousseeuw. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20:53–65, 1987.

[115] L. Ruijiao, L. Bowen, and D. M. Klaus. Cognitive assisted living ambient system: A survey," digital communications and networks. *Digital Communications and Networks*, 1:229–252, 2015.

[116] S. J. Russell and P. Norvig. Artificial intelligence: A modern approach. *Pearson Education*, 2002.

[117] N. Ryan, J. Pascoe, and D. Morse. Enhanced reality fieldwork: the context-aware archaeological assistant. *Computer Applications in Archaeology*, 1997.

[118] Saguna Saguna and Dipanjan Chakrabort Arkady Zaslavsky. Complex activity recognition using context-driven activity theory and activity signatures. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 20, 2013.

[119] A. M. D. Saha and S. Bandyopadhyay. Networking infrastructure for pervasive computing enabling technologies and systems. *Academic Publishers*, 2003.

[120] D. Salber, A. K. Dey, and G. D. Abowd. Ubiquitous computing: Defining an hci research agenda for an emerging interaction paradigm. *Georgia Tech GVU Technical Report*, 1998.

[121] P. Salhofer, B. Stadlhofer, and G. Tretter. Ontology driven e-government. *Electronic Journal of e-Government*, 7(4):415–424, 2009.

[122] A. Scalmato, A. Sgorbissa, and a. R. Zaccaria. Describing and recognizing patterns of events in smart environments with description logic. *IEEE Transactions on Cybernetics*, 43(6):1882—-1897, 2013.

[123] B. Schilit, N. Adams, and R. Want. Context-aware computing applications. *in 1st International Workshop on Mobile Computing Systems and Applications*, 1994.

[124] R. Simpson, S. Levine, and D. Bell. Supporting older adults social network: the design of e-inclusion communication services. *in Sixth Interaction Conference of International Society for Gerontechnology*, 1998.

[125] G. Singla, D. J. Cook, and M. Schmitter-Edgecombe. Recognizing independent and joint activities among multiple residents in smart environment. *J Ambient Intell Humaniz Comput.*, 1(1):57–63, 2010.

[126] T. Sorenson. A method of establishing groups of equal amplitude in plant sociology based on similarity of species content and its application to analyses of the vegetation on danish commons. *I kommission hos E. Munksgaard*, 1948.

[127] J. Sowa. Knowledge representation. *Brooks Cole Publishing*, 2000.

[128] G. Stevenson. An approach to situation recognition based on learned semantic models. *Ph.D. thesis, University of St Andrews*, 2014.

[129] E. M. Tapia, S. S. Intille, W. Haskell, K. Larson, J. Wright, K. A, and R. Friedman. Real-time recognition of physical activities and their intensities using wireless accelerometers and a heart rate monitor. *in Eleventh IEEE International Symposium on Wearable Computers*, 2007.

[130] E. M. Tapia, S. S. Intille, and K. Larson. Activity recognition in the home using simple and ubiquitous sensors. *in 2nd Inter. Conf. In Pervasive Computing (PERVASIVE 2004)*, 2005.

[131] United Nations Population Funds (UNFPA). World population ageing. `http://www.un.org/en/development/desa/population/publications`, 2015. Accessed: 2017-05-28.

[132] Tim van Kasteren, G. Englebienne, and Ben Kröse. Human activity recognition from wireless sensor network data: Benchmark and software. *Atlantis Press Book*, pages 165–185, 2010.

[133] Tim van Kasteren, Athanasios Noulas, G. Englebienne, and Ben Kröse. Accurate activity recognition in a home setting. *Proceedings of the 10th international conference on Ubiquitous computing UbiComp '08*, pages 1–9, 2008.

[134] W3C. Rdf- semantic web standards. `:http://www.w3.org/RDF/`. Accessed: 2017-05-28.

[135] W3C. Sparql query language for rdf. `:http://www.w3.org/TR/rdfschema/`. Accessed: 2017-05-28.

[136] X. H. Wang, D. Q. Zhang, T. Gu, and H. K. Pung. Ontology based context modeling and reasoning using owl. *Workshop Proceedings of the 2nd IEEE Conference on Pervasive Computing and Communications*, 2004.

[137] M. Weiser. The computer for the twenty-first century. *Scientific American*, 1991.

[138] M. Weiser and J. S. Brown. Designing calm technology. *Xerox PARC*, 1995.

[139] C. Welty and R. Fikes. A reusable ontology for fluents in owl. *in Artificial Intelligence and Applications*, 2006.

[140] D. Wyatt, M. Philipose, and T. Choudhury. Unsupervised activity recognition using automatically mined common sense. *in Proceedings of the 20th national conference on Artificial intelligence AAAI'05*, 1:21–27, 2005.

[141] N. Yamada, K. Sakamoto, G. Kunito, Y. Isoda, K. Yamazaki, and S. Tanaka. Applying ontology and probabilistic model to human activity recognition from surrounding things. *IPSJ Digital Courier*, 3:506–517, 2007.

[142] H. A. Yanco and T. Wheelesley. A robotic wheelchair system: indoor navigation and user interface. *Assistive Technology and Artificial Intelligence, Lecture Notes in Computer Science*, 1458:256–268, 1998.

[143] J. Y. Yang, J. S. Wang, and Y. P. Chen. Using acceleration measurements for activity recognition: An effective learning solution for constructing neural classifiers. *Pattern Recognition Letters*, 29:2213–2220, 2008.

[144] S. S. Yau and F. Karim. Contention-sensitive middleware for real-time software in ubiquitous computing environments. *in 4th IEEE symposium on ObjectOriented Real-time Distributed Computing ISORC*, 2001.

[145] Juan Ye. Exploiting semantics with situation lattices in pervasive computing. *Ph.D. thesis, University College Dublin*, 2009.