An Evaluation of general classification models for multi-resident Activity Recognition

XIN LEI

A thesis submitted to Auckland University of Technology in partial fulfillment of the requirements for the degree of Master of Computer and Information Sciences (MCIS)

Primary Supervisor: Dr Sira Yonchareon

Secondary Supervisor: Dr Jian Yu

July 2019

School of Engineering, Computer and Mathematical Sciences

Abstract

Human activity recognition has become a popular research field in smart environments such as smart homes, classrooms, and offices. Most of the research has been focused on single resident environment of activity recognition. However, in real life the live environment us usually inhabited by more than one person. So the research on multi-resident activity recognition is vitally important. The aim is to recognise human motions based on data collected from different types of sensors. Most works have considered multi-resident activity recognition utilizing various classification models. We believe that the existing methods alone cannot efficiently recognise multi-resident actions in the complicated situations of multi-resident environments.

In this research, we address the research question of what is optimal general machine learning classification model for multi-resident activity recognition. We evaluate six general classification models in four datasets. We found that in six classification models the linear SVM has highest accuracy which obtained 88.57%. Second only to linear SVM is HMM which achieved 81.88% in terms of accuracy. After adopting the statistical analysis test, we conclude that the model will influence the classifying of results. Thus, creating or training an efficient and stable classification method remains an open challenge requiring further study.

Keywords: Multi-resident activity recognition, Machine learning classification model

Table of Contents

List of Figures

List of Tables

Attestation of Authorship

I hereby declare that this submission is my own work and that, to the best of my knowledge and belief, it contains no material previously published or written by another person (except where explicitly defined in the acknowledgments, nor material which to a substantial extent has been submitted for the award of any other degree or diploma of a university or other institution of higher learning.

Signature:

Date:12/07/2019

Acknowledgment

This thesis was completed as a part of the Master of Computer and Information Sciences course at School of Engineering, Computer and Mathematical Sciences in the Faculty of Design and Creative Technologies at the Auckland University of Technology in New Zealand. I express my deep gratitude to my supervisor Sira Yongchareon for providing a lot of help during the whole research process. Beginning of project research, my supervisor had work out a plan for my thesis. In the following year, we had meeting every week to discuss and arrange the tasks for following week. Therefore, I can successfully complete my thesis within the specified time. He always patiently guided me solve every problem.

XIN LEI

12/07/2019

Chapter 1 Introduction

The first chapter gives the whole introduction of this thesis and consists of five sections. The first section mainly introduces the background of research content. The second section discusses the motivation for this research project. The research question is addressed in Section 3. The fourth section will introduce the proposed research and contribution. The fifth section will lay out the structure of this thesis.

1.1 Background

Human activity recognition (HAR) refers to the analysis and recognition of human action types and patterns. Generally, relevant information is extracted from various types of data sequences in the physical world, expressed in an appropriate way, and then interpreted to realise the recognition and learning of human behavior (Tapia, Intille, & Larson, 2004). The early work on activity recognition mainly focused on video analysis by arranging cameras in the environment and then using computer vision and image processing technology to track and recognise people or objects. However, activity recognition based on computer vision is vulnerable to poor illumination and other external factors such as diversity of data and these also involve privacy and other shortcomings, which are not generally accepted by users. Sensor-based activity recognition has many advantages, such as a wide application range and being non-intrusiveness and it has become a hotspot problem of current research. Activity recognition has been widely used in medical, security and entertainment applications. Compared with the activity recognition methods based on video images sensor-based activity recognition has the characteristics of low cost, flexibility and good portability (L. Chen, Hoey, Nugent, Cook, & Yu, 2012).

In the past decade, with the rapid development of microelectronics and computer systems, sensors and mobile devices have unprecedented versatility. Sensors convert physical parameters such as temperature, blood pressure and humidity into electrical signals which can be measured by electronic devices and be output as sensor events (L. Chen et al., 2012). In recent years, sensor prices have fallen rapidly, and wireless technology has been widely used in mobile devices in the real world. The high computing power small size and low cost make sensors ubiquitous. Pervasive computing is also a key step in the process of extracting useful information from data obtained from sensors. The recognition and understanding of human activity based on sensor data will be critical to human-centered computing in the future (Caldeira, Rodrigues, & Lorenz, 2012). Behaviour perception technology is an important branch of perceptual computing. It focuses on user behaviour perception and recognition, and it plays an important role in intelligent and personalised services. We can provide more humanised services through the study of perceptual activity recognition, such as geriatric care (O. Kwon, Shim, & Lim, 2012), somatosensory games, and health care (Korhonen, Parkka, & Van Gils, 2003). Sensor-based human perceptual activity recognition is a new branch of behaviour recognition. Compared with image-based activity recognition, sensor-based is more convenient, freer and more secure. It has low dependence on the external environment, can be freely worn, and does not violate user privacy (A. J. Sarkar & Khan, 2011). An increasing number of scholars and research institutes have participated in the construction of sensor-based activity recognition systems. One of the main research directions is to identify human activity by using different types of sensors. At present, there are two main types of sensor development: the first is based on wearable sensors, such as smartphones or wristbands with three-dimensional accelerometers. The second is the non-intrusive sensors or environmental sensors in the intelligent environment. These sensors do not need to be fixed on the individual, but are installed in the surrounding environment of the individual. For example, passive infrared sensors, temperature and humidity sensors or pressure sensors (D. J. Cook & Krishnan, 2015).

So far, in the process of HAR research, we have paid special attention to the problems of the elderly living alone; that is, monitoring the lives of individual elderly people in smart homes. According to the NHS (National Health Service [in the UK]) report, the potential for using sensor data and activity learning for health surveillance and intervention of the aging population is endless. It is estimated that, by 2050, the number of elderly people over 85 will be three times that of today, and 50% of adults will need help in their daily activities by that time (Bouchachia, 2015). At present, there are more and more studies on monitoring individual residents (Nait Aicha, Englebienne, & Kröse, 2013; A. Sarkar, Lee, & Lee, 2010; van Kasteren, Englebienne, & Kröse, 2011a, 2011b). Because more and more people live longer but often suffer from chronic diseases, more caregivers are needed. Under the condition of a smart home, there will be two or more people in most cases, so we can not only focus on the daily activities of just a single resident. The HAR system should be extended to the case of multiple residents.

So far, the research on multi-resident recognition is not as popular as single-resident recognition because it is still relatively new and has many possible research directions. There are still many outstanding challenges in single-resident activity recognition, such as the recognition of complex activities and interleaving activities. Recent papers have highlighted the challenges in this area, as shown in (Bouchachia, 2015) and (Ni, García Hernando, & de la Cruz, 2015). In a recent study, Amiribesheli et al. (Bouchachia, 2015) discussed challenges related to data processing (i.e., maintaining the security, privacy and reliability of active data) and activity recognition modeling such as: identifying cross and concurrent activities, unbalanced data, online activity learning applications, the flexibility and adaptability of the activity model and the scalability of the activity model.

1.2 Motivation/ Rationale

HAR is an approach to recognising human movement using data collected from different types of sensor (Sikder & Sarkar, 2017) . To some extent, activity recognising can prevent the occurrence of potentially unsafe activities effectively. For the elderly living alone, it is necessary to issue timely warnings in case of sudden falls and dangerous activities (Garcia-Ceja, Galván-Tejada, & Brena, 2018b). There are some practical HAR applications such as environmental assisted living system, home monitoring and so on.

It is estimated that by 2020, the number of adults aged 60 and over will reach 1 billion, and by 2050, it may reach 2 billion. Currently, around the world, one tenth of the elderly live alone (Eunju Kim, 2010). With the development of science and technology, the intelligent environment has become mature. The increasing number of the elderly makes activity recognition a hot research field. Meanwhile, the continuous development of sensor and communication technology ensures the smooth realisation of smart home. Different types of sensor devices installed in intelligent environments can be used to identify the activities of elderly people and then send behavioral information to their family members or caregivers in some way so that they can better care for them. At present, many solutions have been proposed for single resident activity recognition (de la Concepción, Morillo, García, & González-Abril, 2017; Kashimoto et al., 2017a; Triboan, Chen, Chen, & Wang, 2017). However, it should be point out that in the real world, the living environment often complex, not only one person. Sometimes including their guests or family members. So it will produce more complex activities. Therefore, multi-resident activity recognition ability is significant.

Multi-resident is still in its infancy in the smart home environment. Before researching the problem of multiple residents many problems of single resident have not been solved (Álvarez de la Concepción, Soria Morillo, Álvarez García, & González-Abril, 2017; Galván-Tejada et al., 2016; Kashimoto et al., 2017a). In recent years, more and more researchers have focused on the field of multi-residents as a study in the smart home environment and multi-resident complex activity recognition. There are different recognition techniques for the multi-resident activity recognition based on sensor deployment strategies. From a macroscopic aspect, sensors can be divided into two types. The first method is applying the sensors in the smart environment, usually by attaching to an object thus making them mostly stationary. In the second scenario, human take along the sensors and, in normal conditions, these are typically a smartphone and other wearable devices such as wristband and smartwatch, which is portable. From microscopic aspect, there are five varieties of the sensor that are applied to discern human activities and anomaly detection in the smart environment: vision-based sensing, wearable sensing, smartphones sensing, acoustic sensing and ambient sensing (Hande Alemdar， Cem Ersoy, 2017).

People may be more receptive to wearable sensors than to cameras. Compared to vision-based motion recognition systems, wearable sensor-based systems have no data association problems and require fewer data points to process. However, people usually need to change their clothes every day; this is likely to make them forget to wear the sensor again. Even if the power management module is optimally designed, the battery inside the sensor must be periodically charged or replaced, which is inconvenient for the user. Therefore, when the vision-based system is not suitable for deployment and the wearable sensor is not convenient to use, how to use simple sensors to realise daily activity recognition has excellent research significance.

Most of the existing state-of-the-art research is aimed at the living space of a single resident. Nevertheless, from the actual situation, the living space usually has more than one resident. Therefore, it is a great significance to design and research the multi-resident problem. In fact, there are more and more researches focus on the problem of multi-resident complex activity recognition in the future. However, there are many outstanding and complex problems which make the research slow.

1.3 Problem/ Research Question

Through the above introduction of human activity recognition, we have made it clear that the research focus of this thesis is to classify daily activities. Some preliminary analyses before we start the research can help us to understand the framework and process of HAR. Therefore, the research question in this thesis is:

Which is the optimal general machine learning classification model and what parameters are to be used for multi-resident activity recognition?

Before we begin the study of major issues, we also need to solve some sub questions:

Which classification methods could be used for multi-resident activity recognition?

How the general classification methods for multi-resident in activity recognition be evaluated?

Since our main research content is multi-resident activity recognition, we need to evaluate the general classification methods used in our research process and chose the optimal method to achieve the classification.

1.4 Proposed Research and Contribution

The nature of human activities is usually far more complex in a multi-resident environment compared with a single resident environment. The proposed aim of this research is to extend the existing research on single resident activity recognition to multi-resident situations. The main argument in this thesis is around developing a general classification method for multi-resident activity recognition. Based on the existing research of activity recognition, we chose six machine learning models which are Naïve Bayes, Bayes Network, Support Vector Machine, Decision Tree, Random Forest and Hidden Markov Model to deal with the problems of activity recognition classification. The specifics of these methods used in our experiments will be explained in Chapter 4. We provide the theoretical basis for the entire research in Chapter 3. This thesis also introduced the specific method and process of activity recognition. In Chapter 2, we compare details different methods in existing research.

The overall contribution to our research is through comparing the different classification models then finding which are optimal to multi-resident activity recognition. For multi-resident activity recognition, our research work is based on machine learning, and our research result can meet the developmental needs of multi-resident activity recognition. Our study further investigates multi-resident HAR. The aim is to accurately recognise some complex activities using non-intrusive sensors.

1.5 Thesis structure

This thesis is composed of five chapters. Chapter 2 gives an overview of activity recognition literature. Specifically, Section 2.2 discusses the types of activity recognition and general structure of activity recognition. Section 2.3 presents some different classification methods for multi-resident activity recognition. Section 2.4 presents some samples of publicly available datasets and evaluation methods. Section 2.5 mainly introduces the research challenge and open issues of activity recognition and Section 2.6 summaries of this chapter.

Chapter 3 presents our methodology for multi-resident activity recognition. Section 3.2 introduces the data collection and data information. In Section 3.3 we present the process of data pre-processing. Section 3.4 describes the machine learning classification method that we used. Section 3.5 presents the evaluation methods for the experimental results. We selected four evaluation criteria which are accuracy, F-measure, precision and recall. Section 3.6 introduce the statistical test which include T-test and ANOVA to confirm the experiment result. Section 3.7 is the summary. Section 4.5 provide the statistical analysis to the results.

Chapter 4 presents the experimental results and introduces the specific model and implementations of each activity recognition datasets. Section 4.3 discusses the comparison of bagging and boosting results. Section 4.4 discusses the experiments results.

Finally, Chapter 5 concludes the thesis. Specifically, we draw our conclusions and limitations in Sections 5.1 and 5.2. The propose future works will be discussed in Section 5.3.

Chapter 2 Related Work

Through the in-depth analysis of past research issues and theoretical reviews, in this thesis we mainly evaluate the general classification models for multi-resident in activity recognition. Through using the research of existing literature, we can provide a better idea of the current status of multi-resident activity recognition and classification. This chapter will introduce a variety of classification methods and technical application. First of all, we discuss the overview of human activity recognition. Then we summarise the reviewed papers in recent years to highlight the trends of the area. Thirdly, we will describe different types of activity recognition. Finally, we will expound on the general structure of activity recognition.

2.1 Introduction

Nowadays, smart homes are designed to help and facilitate residents' lives by providing context-aware services. In order to provide these services, it is necessary to understand the activities of the residents, as many services are based on these activities. Therefore, activity recognition plays an important role in many applications in smart homes. HAR has developed matured to play a vitally important role in artificial intelligence. In some ambient intelligent environments like homes, offices and even classroom, there has been much research during last decade. So HAR is a popular research field especially in smart home. In the meantime, the development of sensor and communication technology is more significant in this field. The HAR usually uses two approaches for collecting information; these are external and wearable sensors. Some wearable sensor like wristband and smartwatch, the subjects need to wear the devices every day. There is a large variety of research based on this. Larbrador and Lara proposed a survey on HAR using wearable sensors. There are five varieties of the sensor are applied to discern human activities and anomaly detection in the smart environment: vision-based sensing, wearable sensing, smartphones sensing, acoustic sensing and ambient sensing (Hande Alemdar， Cem Ersoy, 2017). Other external sensors are usually used for collecting states of the objects and environment, to monitor activities of the residents (Bouchachia, 2015). Generally, for wearable sensor-based systems there is no data association problem and these require fewer data processes. However, based on some privacy issues, most people are not comfortable with the wearable devices. The battery inside the sensor may breakdown and cause inconvenience. So how to use simple sensors to realise daily activity recognition has excellent research significance.

With the growth of aging population, how to build a safe and intelligent living environment will be particularly important for the elderly who live alone. Based on this issue, many research proposed some real-time ambient assisted living solutions. In order to improve the quality life of the elderly living alone in the smart environment, Nirmalya et al. construct a complete deployment solution for service providers and developers from the main concepts, equipment, technology and models (N. R. A. M. D. Cook, 2016). Qin Ni et al. targeted to independent living life proposed some main activities classification and data processing methods (Qin Ni, 2015). However, even if most elderly people live alone they may have pets or some guests. It's very necessary to develop solutions for multi-residents.

2.2 Overview of Activity recognition

Activity recognition is the process of identifying the specific human movement or action based on different types of sensors. Activity recognition mainly focuses on video surveillance, patient monitoring systems, human computer interaction and similar devices. With the development of the smart environment, HAR has become a continuous research problem. At present, the smart environment is gradually changing people's way of life. Especially for some infants and the elderly who need to be taken care of, the existence of smart environment greatly improves the possibilities for independent life. At the same time, the activities in the smart environment are changeable, complex and interactive. Therefore, there are higher requirements for the actual deployment of sensors and the driving technology of monitoring. The development of HAR has made great progress in monitoring of people’s health. Now it is attracting more and more researchers' attention. HAR can not only detect people's daily movements, but is also used for the clinical management of the elderly in medical institutions (HaiderJanjua，RimHelaoui, 2016).

Many works have been done on the single resident recognition situation. Fox example, Yale Song et al. proposed a solution for the single sequential activity (Jeremie Saives, 2015). Nirmalya et al. proposed a classification of main activities for the independent living elderly (N. R. A. M. D. Cook, 2016). There are also efforts to solve the problem on interleaving activity recognition as described by (T. S. Daniele Riboni, Gabriele Civitarese, Heiner Stuckenschmidt, 2016; Hadi Tabatabaee Malazi, 2018). However, in the real-world situation, human activities are more complicated and sometimes involve more than one person in the home such as collaborative activities by two people. Thus, more attention has been paid to multi-resident activity recognition by the researchers in recent years. From 2010 to 2018, the contribution of single resident activity recognition is presents an upward trend. However, multi-resident activity recognition is still relatively limited compared with single resident activity recognition. With the development of smart home and sensor technology, multi-resident activity recognition even for some complex activity recognition will be significantly developed in the future(Jing Zhao, 2017). There are different recognition techniques for this area based on sensor deployment strategies. Generally, there are two major classes approaches based on sensor deployment and selection: wearable devices and infrastructure sensor which can direct installed into environment component. Wearable devices are body-worn sensors. Specifically, there are five types of sensors used to discern human activities and anomaly detection in the smart environment: vision-based sensing, wearable sensing, smartphones sensing, acoustic sensing and ambient sensing (Hande Alemdar, Cem Ersoy, 2017). The solution based on video surveillance usually installs the camera to capture the human body movement data within a specific area and relies on the image analysis technology to recognise the movement (Uddin, 2017). It is usually used in security surveillance. The elderly who are living independently need these devices to guard against accident. However, this technique has the following disadvantages: (1) Equipment such as cameras, video storage and analysis equipment are expensive, for most users are not affordable; (Asma Benmansour, Bouchachia, & Feham) They undermine user privacy, constant monitoring of video has dramatically intruded on personal privacy (Yin, Fang, Mokhtari, & Zhang, 2016a). Thus, we will not consider video-based sensors in this survey. On the other hand, wearable-based sensors for activity recognition has the characteristics of low equipment cost and low privacy intrusiveness, which effectively makes up for the lack of video surveillance. However, wearable sensors-based technology also has its limitations. Some people, especially the elderly are reluctant to wear the devices, or sometimes they forget to wear them. Therefore, we believe that interaction-based sensors such as PIR sensor, temperature sensors, and humidity sensors are used for helping activity recognition will become a trend in the future (Kashimoto et al., 2017b).

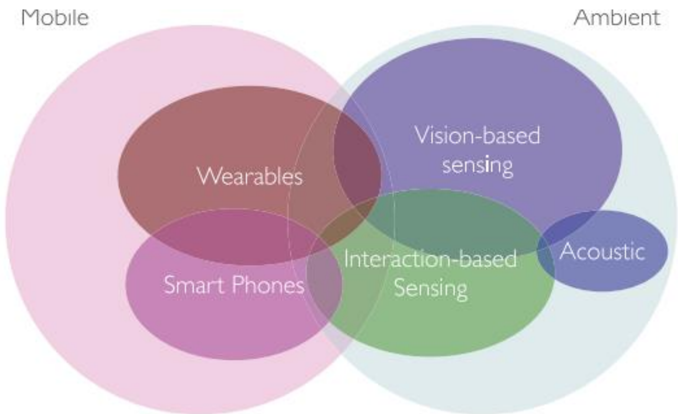


Fig.2.1 Classification of human activity recognition based on sensors (Hande Alemdar, Cem Ersoy, 2017)

In recent years, researchers have made extraordinary progress in the field of daily motion recognition. In general, traditional human body daily motion recognition is based on processing visual information. The typical method based on visual recognition has two main steps, feature extraction and pattern recognition. However, to avoid complex image processing, some researchers use other wearable or binary sensors instead of visual sensors. Wilson et al. (Damla Arifoglu, 2017) proposed the problem of simultaneous target tracking and motion recognition. They used four binary sensors, including motion detectors, photo-interrupt sensors, pressure pads, and contact switches; dynamic Bayesian networks are used to model position tracking and motion recognition indoors, using RaoBlackwellised particle filters (RBPF) to solve data association problems. Zhu et al. (U. A. B. U. A. Bakar, 2015) used motion data and position information to infer human motion. In their study, an inertial sensor was attached to the torso of the human body to provide motion data, and another optical motion capture system provided position information. They use neural networks and hidden Markov models to identify coarse-grained and fine-grained motions, and further use Bayes' theorem to fuse motion and position information.

2.2.1 Types of activity recognition

Most of the researches have studied the recognition of activity for single resident. A single activity usually has only one action. For example, make a phone call. Nevertheless, even with a single person in the house, the activity is more complex because the activity is composed of many sub-activities. For example, washing hands then cooking is a sequential activity. It means each action is performed after another action in a sequence without any interweaving.

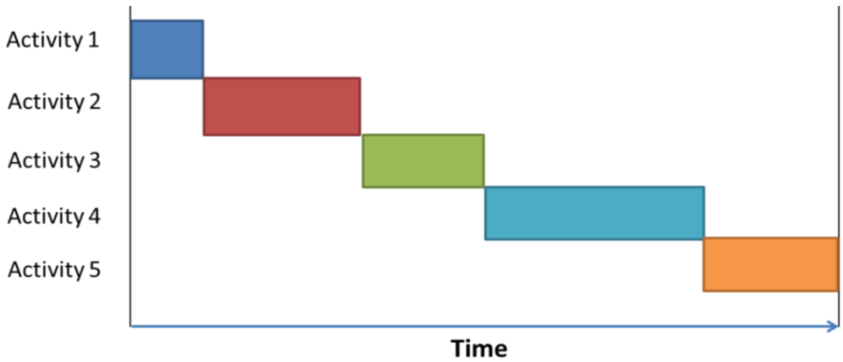


Fig.2.2 Sequential activities (single resident)

Besides sequential activity, there are two other activities, which are interleaving activity and concurrent activity. Interleaving activities refers to a single person shifting activity among some other activities. For example, a resident is cooking food and boiling water in the kitchen; these two activities should execute one by one. The next activity type is concurrent activity. It refers to more than one activity performed by a single resident simultaneously (e.g., watching TV and drinking tea). Concurrent activities are events that occur at the same time, so they share the time intervals. It means that one user performs two different activities simultaneously or multiple users perform one activity at the same time. For example watching TV and drinking tea.

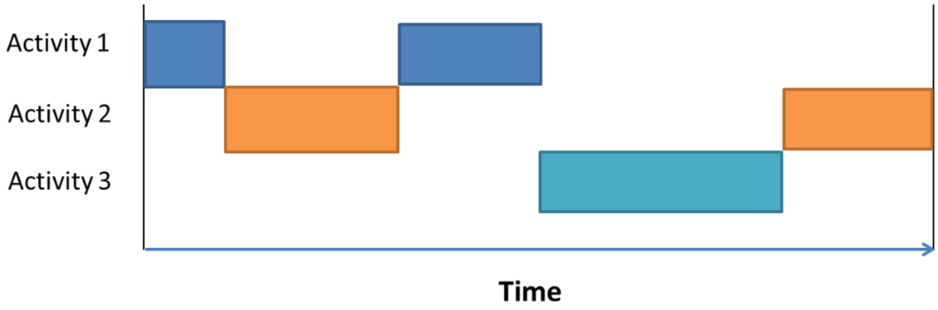


Fig.2.3 Interleaving activities (single resident)

In the real world, the living environment often has more than one person, and human activities are also more intricate (Alaa, Vaidehi, Doreen, hnstedt, & Ralf, 2016). Thus, multi-resident activity recognition is still more challenging, particularly in multi-resident complex activities between two or more people. The problems mainly focus on two kinds of activity, which are parallel activity and collaborative activity (Asma Benmansour et al., 2017). Naturally, with increasing number of residents in the living space, the complexity of activity recognition increases as well. Parallel activity refers to many activities performed by different people in the living space at the same time. For example, one person is making tea in the kitchen, and the other resident is making a phone call in the living room. Collaborative activity is another type of activity recognition in multi-resident. It refers to more than one person join efforts together in a synergistic manner so that each resident performs one same activity together (e.g., two people moving a dining-table together), or they work separately but for one objective (Asma Benmansour et al., 2017) (e.g., two people preparing dinner together in the kitchen).

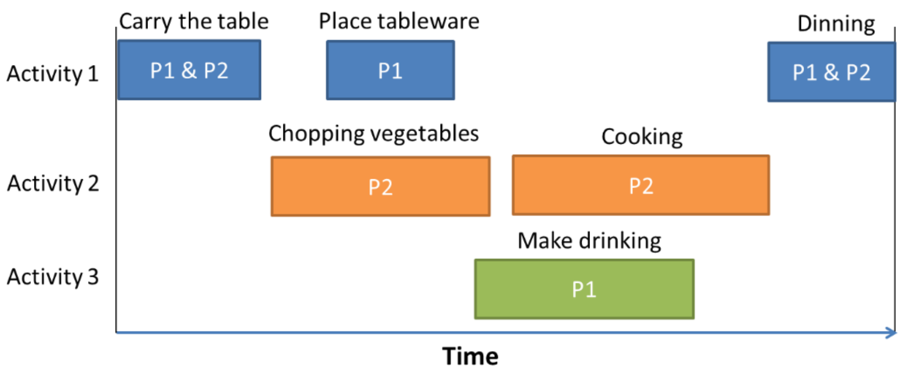


Fig.2.4 Collaborative activities (two residents)

These kinds of activity mentioned above are the main activities in our daily life. Many studies have addressed some problems such as sequential activity (Malazi & Davari, 2017) and interleaving activity (Daniele, Timo, Gabriele, & Heiner, 2016), but it is seldom studied involve parallel activity and collaborative activity, especially complex activities in the multi-resident situation. It is essential to understand the classification of sensor and activity because when we want to identify the activity, the first thing is the chosen sensor, and what types of activity will be recognised. After the sensor is selected, we can start recognising activity.

2.2.2 General structure of activity recognition

There are four steps for the activity recognition system to identify the activities. The first step is raw-data acquisition and then using different types of sensors to process data and obtain contextual information. The second step is the collected data will be processed in different manners to identify activities based on the requirement of research. For example, remove the noise from data and data segmentation. In general, different types of sensor collection activities need to be used simultaneously and correctly identified, and each sensor can collect raw data in a different format. Therefore, in order to obtain useful information that can be applied to the experiment, we must preprocess the data because the original data itself cannot be used in the activity detection algorithm. The third step is feature extraction, and different data features are extracted from processed data. The last part is the classification. There are various segmentation techniques and classification models can be applied to analyse the data of activity recognition (Lara & Labrador, 2013).

In the next section, we will discuss some classification methods used in activity recognition.

2.3 Different classification models for multi-resident activity recognition

In this section, we will describe different technologies can be used to classify multi-resident activity.

2.3.1 Naïve Bayes classifier (NBC)

Naïve Bayesian classifier (NBC) is the most commonly used and also the simplest probabilistic models in the field of activity recognition (Yin et al., 2016a). It depends on the Bayesian theory to establish decision boundaries using all assumptions that input the features independently. These assumptions make the classification easy to handle. The observations and labels in joint probability can be decomposed into

$$p(X, Y) = \prod_{t=1}^{T} p(x_t | y_t)\, p(y_t)$$

In this formula, where $p(y_t)$ is an activity-based prior probability. We assume that all the input data are independent, and then we can directly calculate the conditional probability of the labeled data(X, Y) (Yin et al., 2016a). Thus, the formula can be described as follows: In our settings, the set $X = \{X_1 . X_2 \cdots X_n\}$ expressed as the sensors data and Y represents different activities as shown in Fig.2.5.

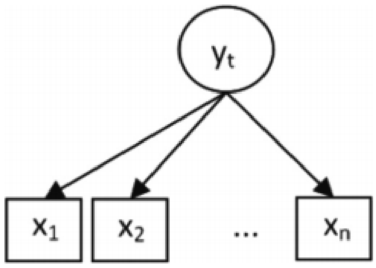


Fig.2.5 The structure of NBC (ASMA BENMANSOUR, 2016)

The Naïve Bayes Classifier has been applied in many multi-resident research studies. The authors (Yin et al., 2016a) used environmental sensors, including temperature sensors, humidity sensors and accelerometers, to track the locations of multiple residents and identify their daily movements at different locations in the room. Then NBC was used for classification and recognition to get accuracy which is 73%. However, NBC is not suitable for modeling time-specific processes due to its own defects. Similarly, Tapia et al. (Emmanuel Munguia Tapia, 2004) used NBC to recognise daily activities. The experimental dataset consists of 77 binary sensors installed on house doors, cabinets, tables and other target objects. Activities include daily activities such as bathing, sleeping and cooking. They got 89% accuracy when they set up NBC differently. NBC, on the other hand, can also be used to process video data. Messing et al.(Ross Messing, 2009) evaluated NBC from video data, and they followed "phone use", "drinking water" and "snacking". The results showed that NBC had good performance with an accuracy rate of 89%.

2.3.2 Bayes Network

Bayes Network is also known as Probabilistic Network or Belief Network. Bayesian network is associates different variables with adjacent temporal steps. This is often called a Bayesian network of "two-time slices," because at any point in time Bayes network the value of a variable can be calculated from the internal regressions and the direct *a-priori* (time T-1)c. DBNs are commonly used in robotics and have proved to have great potential in a large number of data mining applications. For example, BNs have been used in speech recognition, digital forensics, protein sequencing and bioinformatics. BN has also proven to produce Kalman Filters and Hidden Markov Models equivalent solutions (Alam, Roy, Misra, & Taylor, 2016). Fig.2.6 shows the structure of Bayesian network.

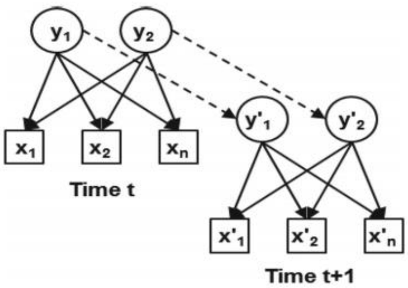


Fig.2.6 The structure of Bayes Network (ASMA BENMANSOUR, 2016)

The first proposed use of discrete Bayesian filters is Wilson and Atkeson (Wilson & Atkeson, 2005). In experiments, they perform both activity recognition and recognition. The problem of multiple user data association is solved by using Rao Blackwellized particle filter. Finally, they only summarized the results of the experiment rather than the real-world data. The results of simulation data show that the accuracy of two people is 98%, and that of three people is 85%.

Alam et al. (Alam et al., 2016) proposed a general dynamic Bayesian model for single resident, which was extended to a coupled HDBN model and applied to multi-resident recognition cases. The experiment used PIR sensors and mobile phones to collect activity information. It is includes macro activities such as washing dishes, preparing meals and using computers. There are also some postural activities. To evaluate the accuracy of the test model, they used CASAS dataset and CACE (real world dataset). Experimental results show that the recognition rate of HDBN is 95% in multi-resident environment. The average accuracy was 20% higher than HMM, 8% higher than CRF, and 5% higher than CHMM.

2.3.3 Support Vector Machine (SVM)

Support Vector Machine (SVM) is a state-of-art classification approach which has been widely used in various applications such as face recognition and stock classification. According to Fig.2.7, we can see that classes are divided by hyperplanes. The margin is the distance from the hyperplane to the nearest data points, which is called support vector. Support vector machines determine the optimal hyperplane by using different optimisation techniques to maximise the margin. SVM has some different kernel functions (e.g., radial basis kernel, polynomial kernel, etc.) Using kernel function to map the nonlinear separable data from the input space into a higher space where data can become linearly separable.

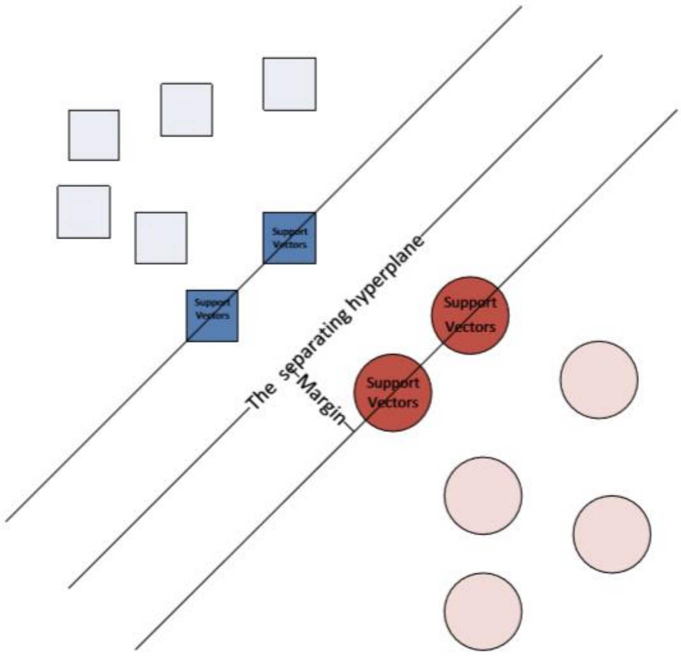


Fig.2.7 The hyperplane, margin and support vectors (Bouchachia, 2015)

In recent years, support vector machines (SVM) have been applied in the field of activity recognition. Fleury et al.(Anthony Fleury, 2010) describe the application of SVM. Experimental data were collected through a set of binary (such as PIR, flood detector) and non-binary (such as microphone, wearable motion sensor) sensors. In the whole experiment, a SVM-based system was used to identify 7 activities. The method of cross validation is used to obtain the satisfactory classification rate, the accuracy of polynomial kernel is 75%, and the accuracy of gaussian kernel is 86%.

Cook et al. (Diane Cook, 2013) used SVM, Naïve Bayesian classifier and Hidden Markov Model for activity discovery and recognition. The data were collected from three elderly people living in SHs it included five daily activities and installed PIR and CSS on doors. The results show that SVM is superior to the other classifier, with an average accuracy of 91.52 % for the three families. In addition, an activity discovery model was introduced to detect new activities and enhance activity recognition. SVM can improve the accuracy by 10% when using activity recognition.

In Luštrek and Kaluža (Mitja Luštrek, 2008), SVM was used in other algorithms (C4.5 DT, NBC, K-nearest Neighbor, and Random Forest) for fall detection. The data was collected using radio-tag equipment installed at key joints in the three men's bodies. The evaluation of the algorithms showed that SVM is better than other classifiers performance which accuracy is 97.7 %.

2.3.4 Decision Tree

Decision Tree (DT) is typically generated top-down. In each tree, event (event in this thesis means decision) can lead to two or more events, which leads to different outcomes. Then this decision branch is drawn into a graph much like a tree's branches, hence we call it Decision Tree. It is a common classification model, which is also a type of supervised learning. A Decision Tree is also utilized for regression when the output data is continuous. As can be seen, it consists of some nodes that represent branches and features and describe the values of the features. Each leaf node represents a class label (Amiribesheli, Benmansour, & Bouchachia, 2015).

Other algorithms such as C4.5 and CART perform two phases, namely tree growing and pruning, while others grow the tree (Lior Rokach, 2005). Based on the wearable triaxial acceleration dataset, Ravi et al.(Nishkam Ravi, 2010) adopted C4.5 for recognising activities. Activities include going up and down stairs, standing up and sitting down. Experimental results show that C4.5 can achieve an accuracy rate of 97.29% after repeated training for the same dataset. Achieved 98.53% accuracy in many user data sets; only 77.95% accuracy was achieved for data not from the same day.

According to Prossegger and Bouchachia (Markus Prossegger, 2014), used Decision Tree to recognise the common daily activities in a multi-resident environment. They proposed a new algorithm which named E-ID5R is an extension of ID5R. In this new algorithm they added leaf nodes and then achieved multi-gagged. The E-ID5R algorithm gradually adopts to new collaborative activities and new instances, so they selected ARAS dataset to evaluate it.

Experimental results show that the algorithm achieves a good classification performance in House B which obtained 82%.

2.3.5 Random Forest

Random forest is a non-probabilistic decision tree based on statistical learning theory. It combines a Bootstrap sampling method and a decision tree algorithm to build a tree classifier set containing multiple basic classifiers. For each tree, only the available sub-sets of the datasets are considered (Nef et al., 2015). New data is categorised, entered into each tree, and classified and predicted using a voting strategy. In addition, for each node only have a random subset of all features is used for splitting. Because this algorithm solves the problem of over-fitting well to some extent, its classification performance is better than that of single classifier.

Tobias et al. (Nef et al., 2015) collected ADL (Activity Daily Life) data by installing passive infrared sensors (PIR) to monitor the daily activities of elderly people. They propose a new technology that uses data collected from existing smart homes for training and then applies it to other smart home activity recognition systems. They asked users to tag all instances of ADL and then applied different data mining techniques to analyse sensor data. To improve the accuracy of activity recognition, they used three different supervised classification algorithms and compared the row energies, including Naïve Bayes, Support Vector Machine and Random Forest. According to the experimental results, the average specificity of RF was 96% and the accuracy was 74% better than NB and SVM classifier. Through the training, mining and clustering of the new framework, the results of RF classifier have been significantly improved.

Lu Xu et al. (Xu, Yang, Cao, & Li, 2017) proposed a new activity recognition method based on random forest. Throughout the study, they used a single wearable device to collect physiological parameters of different parts of the human body as it moved. After a series of processing and analysis it is used to estimate the human movement. According to the activity recognition model and algorithm based on random forest proposed by the authors, they verified the effectiveness of model by experiments, and accuracy reached 90% in recognising walking, walking upstairs and static recognition. The experimental results are compared between decision tree and ANN. Based on this study, we know that random forest can not only calculate the similarity between examples, but can also realise unsupervised cluster analysis and anomaly monitoring.

In the study conducted by Ramon et al., they evaluated the clubbing-based activity approach and improved the automatic selection group number and instance redistribution process (Garcia-Ceja & Brena, 2018). In their experiment, four public activity recognition datasets were applied to seven different classification models, the performance of this model was proved by 10-fold cross-validation. It included Random Forest, SVM, DT and other models. Experimental results show that all statistical signals except the Random Forest model cannot be significantly improved. By comparison with other models, the accuracy of Random Forest in all four datasets is higher than for other models after the improvement of existing activity methods which are 86%, 89% and 75%, respectively.

2.3.6 Hidden Markov Model (HMM)

Hidden Markov Model (HMM) is a probabilistic model based on time series. It describes the process of generating unobservable random sequences from a hidden Markov chain and then generating observation of random sequences from various states (Amiribesheli et al., 2015). From given the input sequence $(x_{1,}x_{2,}\cdots x_T)$, hidden state sequence $(y_{1,}y_{2,}\cdots y_T)$ can be estimated as illustrated in Fig.2.8.

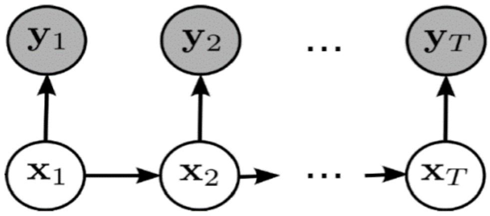


Fig.2.8 The structure of HMM

HMM is one of the most popular probabilistic model deployed to human activity recognition (Bouchachia, 2015). The activities in the ovals from phone call to clean are defined as hidden states as shown in Fig.2.9, while other serial number in the rectangles means observation sequence and the data collected from the sensors. The horizontal arrow represents the probability of transition, and the downward arrow represents the probability of emission for the corresponding observed state.

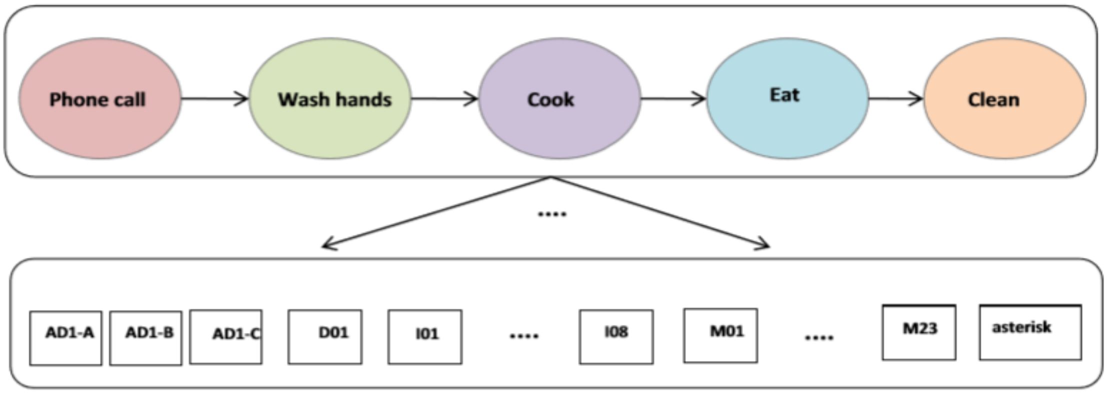


Fig.2.9 Activity modelling using HMM (Bouchachia, 2015)

Research on multi-resident activity recognition has been developing recently due to the increasing demand for health monitoring in ambient intelligent environments. The task can be done by employing sequence models to perform prediction on the activity events over time. The Hidden Markov Models (Markus Prossegger, 2014) is a popular statistical model for sequential data. It is characterised by the dependency of an observation variable on a hidden variable at each time step, and the dependency of the hidden variable itself on its previous state. HMMs can be easily employed for activity recognition. In particular, one can define the observation as the sensors state, i.e. video frame, wearable or/and ambient sensors' values, and the hidden variable as the activity (Eunju Kim, 2010). In multi-resident smart homes, HMMs have been studied intensively, as shown in previous works (D. J. Cook, 2010; Hande Alemdar, 2013) (R. Chen & Tong, 2014; Singla, Cook, & Schmitter-Edgecombe, 2010). A straightforward approach is to use a single HMM for combined activities, i.e. Treating the activities of all residents as a random variable. For example, the activities can be combined as joint labels so that they can be represented by a single hidden variable.

2.4 Evaluation Method and Dataset

In this section, we will consider other important issues related to activity recognition. Firstly, we will describe the datasets for activity recognition in recent years. Then evaluation techniques will be mentioned, and some contributions of multi-resident activity recognition recently will be provided.

2.4.1 Datasets for multi-resident activity recognition

Datasets are important for activity recognition for evaluating the performance of research. There are lots of datasets can be used in activity recognition such as PUCK, mHealth, UCI and other real-life datasets (Al-Nawashi, Al-Hazaimeh, & Saraee, 2017). However, in the multi-resident scenario, most of the researchers choose CASAS or ARAS datasets for evaluation.

Center for Advanced Studies in Adaptive Systems (CASAS) was established at Washington State University in 2007. Up to now, are more than 20 datasets have been collected by CASAS and some of them are related to multi-resident datasets (Ifat Afrin Emi & John A. Stankovic, 2015). For example, one dataset named “Multi-resident ADL Activities” represents two people in the apartment at the same time performing fifteen ADL activities in the apartment. The activities include sequential activity, interleaving activity, parallel activity and collaborate activity. The details of activities are shown in Table 2.1.

Person A	Person B
Fill medication dispenser	Hang up clothes
Water plants	Move the couch and coffee table
Help move the couch and coffee table	read a magazine
Play a game of checkers	Sweep the kitchen floor
Set out ingredients for dinner	Play a game of checkers
Read a magazine	Set dining room table for dinner
Gather food for a picnic	paying an electric bill
Pack food in the picnic basket	Retrieve dishes
	Pack supplies in the picnic basket

Table 2.1 Multi-resident ADL Activities Dataset (Asma Benmansour et al., 2017)

In addition to this, there are seven types of sensors were applied to this scenario, which are motion sensors, item sensors, cabinet sensor, water sensor, burner sensor, phone sensor and temperature sensors (A. Benmansour, Bouchachia, & Feham, 2015). Fig.2.10 depicts the specific deployment of sensors in this room.

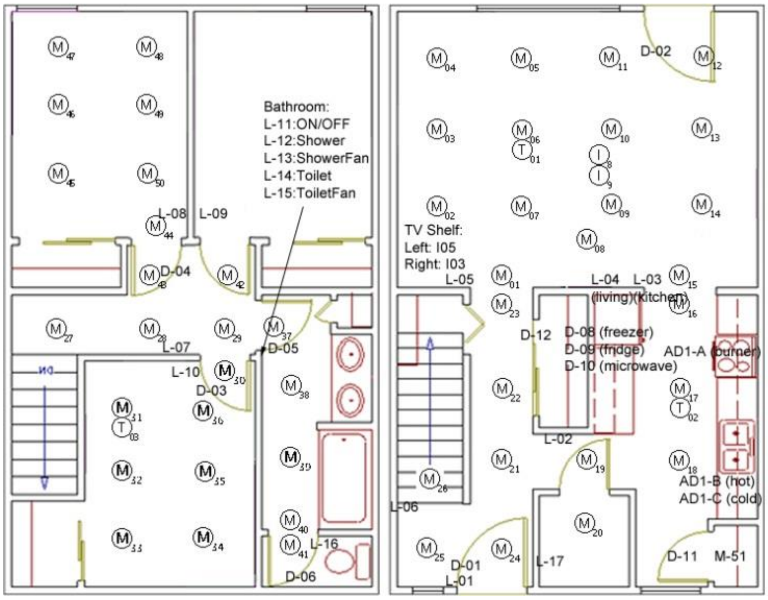


Fig.2.10 Sensor deployment in Multi-resident ADL dataset (Hadi Tabatabaee Malazi, 2018)

ARAS is one of the most used datasets used in the multi-resident activity recognition scenario. This dataset can be divided into two separate datasets, House A and House B. This dataset is the only one dataset for multi-resident using interaction-based sensors. In this dataset, all the same activities were performed in House A and House B but different sensors were used. A total of twenty-seven activities were performed (A. Benmansour et al., 2015). An important feature of ARAS dataset is that it contains a large number of human activities and characteristics. Table 2.2 and Table 2.3 summary the activities and sensors in House A and House B.

	Activity	Sensor Type	Place
House A	-Other -Going out -Preparing breakfast -Having breakfast -Preparing lunch -Having lunch -Preparing dinner -Having dinner -Washing dishes -Having snack -Sleeping -Watching TV -Studying	-Photocell -Photocell -IR -Force Sensor -Force Sensor -Distance -Distance -Photocell -Photocell -Photocell -Photocell -Contact Sensor -Contact Sensor -Contact Sensor -Sonar Distance -Sonar Distance -Distance -Distance -Temperature -Force Sensor	-Wardrobe -Convertible Couch -TV receiver -Couch -Couch -Chair -Chair -Fridge -Kitchen Drawer -Wardrobe -Bathroom Cabinet -House Door -Bathroom Door -Shower Cabinet Door -Hall -Kitchen -Tap -Water Closet -Kitchen -Bed

Table 2.2 HouseA details in ARAS (Alemdar & Ersoy, 2017)

	Activity	Sensor Type	Place
House B	-Having shower	-Contact Sensor	-Kitchen cupboard
	-Toileting	-Contact Sensor	-Kitchen cupboard
	-Napping	-Contact Sensor	-House Door
	-Using Internet	-Contact Sensor	-Wardrobe Door
	-Reading book	-Contact Sensor	-Wardrobe Door
	-Laundry	-Contact Sensor	-Shower Cabinet Door
	-Shaving	-Distance	-Tap
	-Brushing teeth	-Force Sensor	-Chair
	-Talking on the phone	-Force Sensor	-Chair
	-Listening to music	-Force Sensor	-Chair
	-Cleaning	-Photocell	-Fridge
	-Having conversation	-Photocell	-Kitchen Drawer
	-Having guest	-Pressure Mat	-Couch
	-Changing clothes	-Pressure Mat	-Couch
		-Pressure Mat	-Bed
		-Pressure Mat	-Bed
		-Pressure Mat	-Armchair
		-Sonar Distance	-Bathroom Door
		-Sonar Distance	-Kitchen
		-Sonar Distance	-Closet

Table 2.3 HouseB details in ARAS (Ifat Afrin Emi & John A Stankovic, 2015)

In a general study, scholars often use more than one dataset to evaluate the performance. For example, Ifat et al. (Ifat Afrin Emi & John A. Stankovic, 2015) used not only ARAS dataset but also CASAS to evaluate their SARRIMA system. Alam et al. (Alam et al., 2016) even collected a real-world dataset named CACE to evaluate their method and also used CASAS for comparison. CASAS and ARAS have become benchmarking datasets for multi-resident activity recognition. With these datasets, we can well evaluate the proposed method.

2.4.2 Evaluation metrics

There are many approaches to evaluate proposed models, and the evaluation approach should illustrate the methods of using data, training data and testing data. Generally speaking, most of the researchers use different kinds of cross-validation to measure the performance of recognition (Lara & Labrador, 2013). The recognition result is used in the confusion matrix $M_{n\times n}$ for the n-type classification problem. In the binary classification problem, the values can be acquired from the confusion matrix included True Positives, True Negatives, False Positives and False Negatives. All of them is the number of positive samples or negative samples to be positive or negative. We can extend these metrics to n classes even though it is binary classification. In some situations, on the basis of the particular class, the instance can be positive or negative (Rodomagoulakis et al., 2017). For example, positives could be all instances of walking while negatives could be all instances other than walking. In the activity recognition, classes represent activities. They used accuracy and false positive rate to calculate each of the two indicators for each type of feature, and then chose the best results to report. And assessed the effect of the feature on the efficiency of resident recognition. For instance, in a two-person household, one person would take much more time at home than the other one, so the probability of an individual event would be attributed to the person responsible for most of event, leading to a higher false positive rate.

In general, accuracy is a typical indicator for classification. By calculating accuracy, we can assess the balance problem. It is the most common evaluation to conclude the entire classification performance. Another valid choice of evaluation metric is precision. It is a measure of accuracy. The third one is recall. It is a measure of coverage and it measure the ratio of positive samples are classified as positive to the total number of positive instances. The last one is F-measure which almost used to many classification projects.

When comparing Naïve Bayes and HMM, Yin et al. use accuracy and F-measure to evaluate the performance of model (Yin, Fang, Mokhtari, & Zhang, 2016b). Yaqing and Yong et al. (Liu, Ouyang, Liu, & Chen, 2017) use accuracy, precision and recall to evaluate the effectiveness of the resident recognition problem. Some of research use all of them to evaluate the experiment result such as (Fahad, Tahir, & Rajarajan, 2014), (M.-C. Kwon & Choi, 2018), (Fahad, Tahir, & Rajarajan, 2015).

The summary of selected research in Table 2.4.

Ref.	Model	Dataset	Result			
			Accuracy	F-Measure	Precision	Recall
[Yin et al.,2016]	Naïve Bayes and HMM	Real-world dataset	-Naïve Bayes: 73.01% -HMM: 73.10%			
Tapia et al. [2004]	Naïve Bayes	ARAS dataset	89%	88%		
Messing et al. [2009]	Naïve Bayes	KTH dataset	89%			
[Alam et al.,2016]	Bayes Network	CASAS and CACE dataset	95%			
[Anthony Fleury,2010]	SVM (Polynomial, Gaussian)	Real-world dataset	-Polynomial: 75.9% -Gaussian: 86.2%			
Cook et al. [2013]	SVM, Naïve Bayes, HMM	CASAS dataset	-SVM: 91.52% -NB: 90.82% -HMM: 90.85%			
(Mitja Luštrek, 2008)	SVM, Naïve Bayes, Random Forest	Real-world dataset	-SVM: 97.7% -NB:84% -RF: 93.4%			
Ravi et al. [2010]	Decision Tree (C4.5), SVM, Naïve Bayes		-SVM: 98.16% -DT: 98.53% -NB: 98.86%			
Markus Prossegger [2014]	Decision Tree (extension of ID5R)	ARAS	82%			
Tobias et al. [2015]	Random Forest, Naïve Bayes, SVM		-RF: 96.53% -NB: 89.89% -SVM:92.34%	-RF: 71.33% -NB: 27.88% -SVM: 41.23%		
Lu Xu et al. [2017]	Random Forest, Artificial Neural Network,		-RF: 90% -DT: 75% -ANN:80%			

	Decision Tree					
Ramon et al. [2018]	Decision Tree, SVM, Naïve Bayes, Random Forest	Real-world dataset	-RF: 84% -DT: 81% -NB: 78% -SVM: 85%			
Crandall et al. [2008]	Naïve Bayes	Real-world dataset	92%	90%		
Crandall et al. [2010]	Naïve Bayes, HMM	CASAS	-NB:93.3% -HMM: 94%			
Cook et al. [2010]	Bayes Network, HMM	Real-word dataset	-BN: 57% -HMM: 90%	-HMM:94%	-HMM: 93%	-HMM: 96%
Chen et al. [2014]	HMM, CRF	CASAS	-HMM: 97.4% -CRF: 97.25%	-HMM: 40.48% -CRF: 39.99%	-HMM: 80.03% -CRF: 80.05%	-HMM: 81.92% -CRF: 79.91%

Table 2.4 The summary of selected research

2.5 Research challenge and gap

In order to bring the multi-resident activity recognition systems enter a more mature stage, some research approaches need to be studied further. Next, we will describe some limitations and open issues for multi-resident activity recognition. As we said before, the single resident situation still has some problems which need to be solved. The multi-resident setting is more complicated and there are some limitations which can be improved. Although more and more researchers have studied the multi-resident activity recognition in recent years, we can say that it still in its infancy. We will highlight some research gaps in this section in order to make significant advances in multi-resident activity recognition.

2.5.1 Existing evaluation and incomplete

In real-world situations, human activities are often complex. The existing research literature on multi-residents has not solved the problem of cooperative activities fundamentally. For different situations, for example, multiple residents carry out staggered or parallel activities ate the same time, or each resident carries out activities simultaneously or in a staggered manner. For the same kind of activity, due to the change of the human body scale, the data collected by the sensor will be different when different people are doing the same action. There exist more complex situations in which a resident switch between an activity and a collaborative activity or performs both in a concurrent manner. In addition, the different habits of each person can also lead to bias in the recognition of movements. For example, one person prefers writing papers while listening to music and another person prefers quiet. Long-term collection of everyone's habits and behaviour is time-consuming, so the existing multi-person activity recognition research is carried out using the same people and sets of activities. However, the situation in real life is more complex, and the existing research does not directly reflect the actual situation. Therefore, we need to further study the characteristics of these actions in order to identify activities that are more complicated.

In complex scenes, obstacles or other objects often obscure activities. Also, different environments and sensors can also interfere with the recognition of movements. Changes in the environment can easily lead to the failure of a fixed apparent model of human activity. The impact of these environmental factors will result in the computer getting different observations, and activity recognition algorithm needs to be able to tolerate the existence of these changes. At the same time, the type and placement of the sensor can have an impact on the algorithm. For device-free sensors, the biggest challenge is how to identify individuals accurately. Current studies such as (Forkan, Khalil, Tari, Foufou, & Bouras, 2015; ZDRAVEVSKI, LAMESKI, & TRAJKOVIK, 2017) used wearable sensors to solve identity individuals problem, but some people are reluctant to wear these devices. Most existing works evaluate their proposed solution for changes for a short period, and some of them used wearable sensors as an assistant. Thus, the handing of dynamic changes over a long run and balanced use of sensors should be investigated.

Many papers published focus on the field of multi- resident activity recognition. However, nearly all of them evaluate the resulting basis on the two-resident situation due to the lack of suitable datasets. Most of researches haven’t investigated the scalability of the models proposed. For the scalability of the activity model, we mainly focus on two aspects. The first one is the scalability of activities. When residents perform new activities, or they receive new guests, the proposed activity model should recognise these activities. The second one is scalability of increasing number of persons (A. Benmansour et al., 2015). Current studies mainly considered two-resident scenario (Alemdar & Ersoy, 2017; Roy, Misra, & Cook, 2016; Yin et al., 2016a), but we still need to improve the models to deal with a large number of residents.

2.5.2 Open issues and research gap

In real life situation, human activities are often executed in a more complicated manner even sometimes more than two residents perform cooperative activities together (Yin et al., 2016a). Although more and more research has addressed interleaved or concurrent activities by multi-residents, there is hardly any research dealing with multi-resident addressing the problems of more than two residents working together.

Specifically, there are two types of activities in the multi-resident activity recognition should be studied in the future. Firstly, concurrent activities or interleaved activities performed by more than two residents (Zhao, Wang, & Lu, 2017). We believe that the difficulty will geometrically increase once there are more than two residents in the smart home environment. Secondly, the most difficult activities for recognition must be the collaborative activities recognition in multi-resident situations (Roy et al., 2016). The existing research has not yet fully addressed the problems of collaborative activities recognition in two residents. Thus, it will still face even more challenges if more than two residents do some collaborative activities.

With the rapid advance of communication technology and sensor technology, increasing multifunctional sensors can be used to identify human activity. As people attach importance to privacy, device-free sensors will be the focus of development. It not only effectively identifies human activities but also does not involve the disclosure of human privacy. Although there have been many studies such as (Alemdar & Ersoy, 2017; Yin et al., 2016a) who applied device-free sensors for multi-resident activity recognition, there are still many difficulties in data association and individual identification. Meanwhile, some studies (Garcia-Ceja, Galván-Tejada, & Brena, 2018a; Huang & Dai, 2017) used PIR sensors, which can capture some analogue signal to analyse the activity and then can recognising the different human activities.

Deep learning is a complex machine learning algorithm that produces good results in speech and image recognition. Deep learning enables the computer to imitate people’s audio-visual and thinking. It solves many problems of complex pattern recognition and makes great advances in artificial intelligence-related technologies. Some international companies such as IBM, Google and others have conducted speech research using DNN (Deep Neural Network). It has been applied in the field of activity recognition in (Wang, Zhang, Gao, Yue, & Wang, 2017). Compared with other methods, the proposed method will be less labour-intensive and allow time for feature selection. The result shows that the proposed method can not only realise people's location but also identify the activity recognition. Therefore, we have reason to believe that deep learning can be ideally applied to the field of multi-resident activity recognition.

2.6 Summary

This chapter is mainly about analysis of existing literature. In the first section we talked about the overview of HAR. We described different types of activity recognition and then expounded on the general structure of activity recognition. Then we classified and discussed different models for multi-resident activity recognition and enumerated those models that are more likely to be used for multi-resident activity recognition in the future but are currently only used for single-resident activity recognition. In the third section we discussed the datasets and evaluation methods. In terms of datasets, CASAS and ARAS are the two most important datasets, but more and more researchers are beginning to collect real-world data for research. For the evaluation method, accuracy, precision, recall and F-measure are mainly used for evaluation. The last section is about research challenges and limitations.

Chapter 3 Methodology

In this chapter we mainly talk about how to use specific methods to answer research questions. In the process of activity recognition, the first step is collecting raw sensor data and then pre-processing and segmenting the raw data in different ways based on the requirement. The next step is feature extraction. Different data features are extracted from processed data. The last step is classification which is focus of this chapter. Throughout the chapter, we describe the model that we used to understand the research more clearly.

3.1 Introduction

From the literature study, we have gained a basic concept of how to carry out activity recognition. In this section, we describe a research method to address the research question. Then we give brief overview of our proposed solution for complex multi-resident activity recognition to provide a basic understanding of our approach. The framework is shown in Fig.3.1.

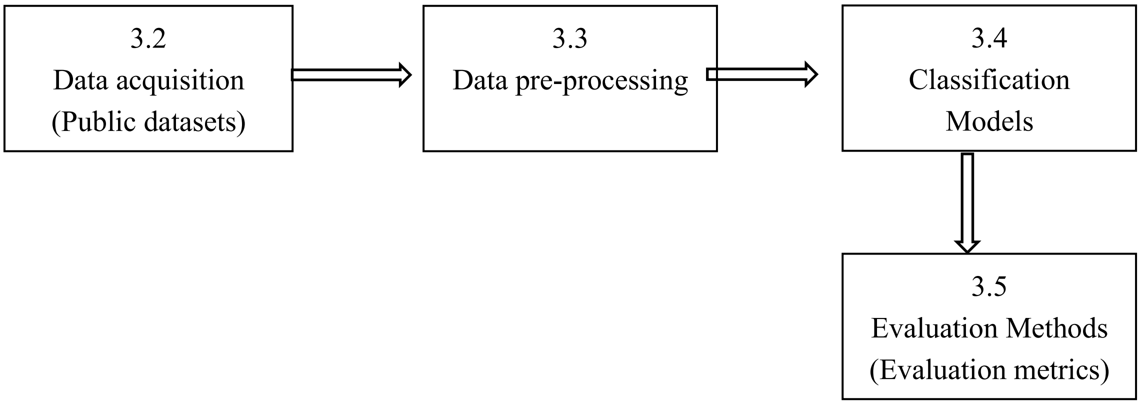


Fig.3.1 The structure of activity recognition

As shown in Fig.3.1, the flowchart is the basic concept of the specific research process. In our proposed method of identifying multi-resident complex actions, the first step is collecting human action data. There are different recognition techniques for the multi-resident activity recognition based on sensor deployment strategies. In our research, we selected public datasets which are ARAS and CASAS. The details will be addressed in Section 3.2.

The second step is data pre-processing. We need to process the raw data acquired by the sensor so as to improve the quality of data mining, then that processed data can be used in the later experimental stages. Data pre-processing is a very important step in the whole process because it affects the accuracy of the output results, also the recognition results of the final model.

The third step is using classification model to classify activities. In traditional machine learning activity recognition, we have to extract and select features before performing activity classification. These features include a peak-to-peak amplitude, a passage duration, the number of peaks, a time for a wave peak to return to the reference line. In this way, activity recognition is a supervised learning method, which generates data by collecting predefined activity classes and mapping them to sensory readings. The main purpose of classification is to identify household living activities from temporal-sequence motion sensor data. In this research, we select 6 general machine learning classification models: Bayes Network (BN), Naïve Bayes (NB), Support Vector

Machine (SVM), Decision Tree (DT), Random Forest (RF) and Hidden Markov Model (HMM). After training the existing experimental data, then predicted the target sample.

The last step is evaluation of the result using metrics. The evaluation method should describe how data is used and trained, as well as how it is validated and tested. The performance metrics used to evaluate models are important in the validation of each classification model. Choosing the appropriate metrics depends largely on the specific problem (e.g., classification, regression). In the following section, we will display 4 criteria used in this research to evaluate the performance of multi-resident activity recognition; these are precision, recall, F-measure and accuracy. Finally, in order to confirm the experiment result we use statistical test.

3.2 Data acquisition

At present, the development of HAR system is mainly targeted on developing and evaluating datasets to a great extent. Some publicly available datasets are important for evaluating activity recognition classification algorithms. Currently, there are lots of datasets about single resident used in activity recognition (L. P. Daniele Riboni, Laura Radaelli, Claudio Bettini, 2011), Kasteren et al. (T.L.M. van Kasteren, 2010). However, there is a real dataset which named CASAS (Center for Advanced Studies in Adaptive Systems) which is collected from multi-resident house. The CASAS dataset included several datasets which are "twor.2009," "twor. summer.2009," "twor.2010," "Tulum," "tulum2," "cairo," and "Multi-resident ADLs." The other dataset, called ARAS which contains two House A and House B datasets. To our knowledge, these datasets are the only ones that can be publicly recorded from multiple residents using pervasive sensors. So in our research we chose the number of them as experiment datasets. Table 3.1 summarises the characteristics of the two multi-resident datasets, which will be described in the following sections.

Dataset	No. of residents	Duration	No. of sensors	No. of ADL	No. of sensor event	Environment	Scripted	Annotation medium
House A of ARAS	1 pair	1 month (continuous)	20	27	2592000	Real house	No	GUI
House B of ARAS	1 pair	1 month (continuous)	30	27	2592000	Real house	No	GUI
"Multi-Resident	26 pairs	Spread over 2 months	37	15	17258	Lab.	Yes	diaries

ADLs”								
“twor. 2009”	1 pair	Continuous period of 2 months	71	9	137789	Lab.	No	diaries
“twor.summer. 2009”	1 pair	Continuous period of 2 months	86	8	772544	Lab.	No	diaries
“twor. 2010”	1 pair	2009–2010 academic year	87	13	2804813	Lab.	No	diaries
“tulum”	1 pair	4 months (Several days are missing)	20	9	486912	Lab.	No	diaries
“tulum2”	1 pair	2009–2010 academic year	36	15	1085902	Lab.	No	diaries
“cairo”	1 pair +1 pet	Continuous period of 2 months	32	11	726534	Lab.	No	diaries

Table 3.1 Characteristics of ARAS and CASAS multi-resident datasets

3.2.1 CASAS datasets

CASAS is a multi-resident dataset based on clinical questionnaires collected from the WSU intelligent apartment test-bed (Reisberg et al., 2001). The dataset is annotated by recording the start and end times of daily activities. There are two types in the CASAS dataset: first one is unscripted active datasets, including "twor.2009", "twor", "Summer.2009", "Twor.2010", "Tulum", "Tulum2" and "Cairo". In this experiment, we selected "twor.2009" and "Twor. 2010". The other one is a scripted active dataset such as "multi-resident ADL". Various types of non-intrusive sensors, such as temperature sensors, pressure sensors, motion sensors and light switch sensors, are installed on the WSU smart apartment test bench. Detailed summary information is shown in Table 3.2. We will discuss in detail of unscripted and scripted datasets information further.

Datasets	Activities	Motion sensors	Door sensors	Light sensors	Item sensors	Temperature sensors	Electricity sensors	Water flow sensors	Burner sensors
"twor. 2009" (D. J. Cook & Schmitter-Edgecombe, 2009)	-clean -Meal preparation -Bed to toilet -Personal hygiene -Sleep -Work -Study -Wash bathtub -Watch TV	51	9	7	1	5		2	1
"twor.summer. 2009"(D.J.Cok &Schmitter-Edgecombe, 2009)	-Bed to toilet -Cleaning -Cooking -Grooming -Shower -Sleep -Wakeup -Work	51	15	10	4		1		
"twor.2010" (D. J. Cook & Schmitter-Edgecombe, 2009)	-Bathing -Bed to toilet -Eating -Enter home -Housekeeping -Leave home -Meal preparation -Personal hygiene -Sleep -Not sleeping in bed -Wandering in room -Watch TV -Work	51	15	11	4	5	1		

Datasets	Activities	Motion sensors	Door sensors	Light sensors	Item sensors	Temperature sensors	Electricity sensors	Water flow sensors	Burner sensors
"tulum"(D.J. Cook&Schmittr-Edgecombe, 2009)	-Cook breakfast -Cook lunch -Enter home -Group meeting -Leave home -Eat breakfast -Snack -Wash dishes -Watch TV	18				2			
"tulum2"(D.J.Cook, 2010)	-Bathing -Bed to toilet -Eating -Enter home -Leave home -Meal preparation -Personal hygiene -Sleeping -Wash dishes -Watch TV -Work bedroom1 -Work bedroom 2 -Work living room -Work table -Yoga	31				5			

Datasets	**Activities**	**Motion** sensors	**Door** sensors	**Light** sensors	**Item** sensors	**Temperature** sensors	**Electricity** sensors	**Water flow** sensors	**Burner** sensors
"cairo"(*WSU CASAS Datasets*, 2007)	--Bed to toilet -Breakfast -Sleep -Wake -Work in office -Dinner -Laundry -Leave home -Lunch -Night wandering -Take medicine	27				5			
"Multi-residents ADLs"(Asma Benmansour, Bouchachia, & Feham, 2017b)	-Filling medication dispenser -Hanging up clothes -Reading magazine -Sweeping floor -Setting the table -Watering plants -Preparing dinner -Moving furniture -Playing checkers -Paying bills -Gathering and packing picnic food	27	8						

Table 3.2 Summary of CASAS datasets (*WSU CASAS Datasets*, 2007)

- *Unscripted Multi-resident datasets*

According to the existing literature, there are six datasets which are “twor.2009”, “twor.summer.2009”, “twor.2010”, “Tulum”, “tulum2” and “cairo”, have not been used as experimental data in the research of multi-resident activities. By understanding the datasets’ information, we know that each dataset was collected by a pair of residents who performed unscripted activities. In detail, "tulum" and "tulum 2" mainly collected the activity data of a married couple, whereas "cairo" contains three kinds of data: the activity data of a couple, the motion data related to their dog and the data of occasional visits by their children. The “twor.2009” and “twor.2010” also collected by two residents of their daily life. This also reflects the variability of the subjects. All datasets are continuous for recording activities. The total length of "twor.2009", "twor.summer.2009" and "cairo" is two months, and "tulum" takes four months, while "twor.2010" and "tulum 2" have gone through a whole year. The dataset details, including Date, Time, Sensor ID, Sensor value, etc., are recorded and described in tables. Each activity is recorded in Residentid_ActivityName Begin and Residentid_ActivityName End.

- *Scripted Multi-resident datasets*

Unlike previous datasets, “Multi-resident ADLs” collection has been used in many research (Hande Alemdar，Cem Ersoy, 2017) (U. A. B. U. A. Bakar, 2015) (Jeremie Saives, 2015). But it was collected in the laboratory, so to some extent, it is not able to reflect the real life of residents. Secondly, the dataset is not continuously recorded. "Multi-Resident ADL" is a dataset consisting of 26 pairs of volunteers who perform activities that have been pre-planned. The activities details shown in Table 3.1. The dataset includes repeated activities within two months, some of which are performed independently by two residents and most of which are performed by two residents in cooperation. This dataset illustrates intrasubject variability. In the “Multi-resident ADLs” dataset, activities collected by sensors are manually marked with TaskID and ResidentID respectively, and detailed time (including Date, Time, SensorID, SensorValue) of each activity is carefully recorded to ensure the accuracy of the label. Each file of the dataset contains 15 activities, for a total of 17,232 events. If the event is triggered by two residents at the same time, it is represented by the Date, Time, SensorID, SensorValue, Resident ID, TaskID, ResidentID, TaskID. Manually annotate the dataset by recording the activity that start and end.

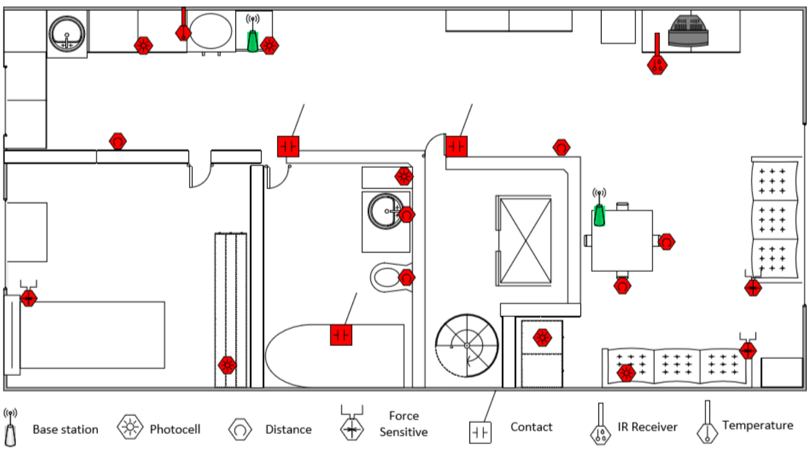
3.2.2 ARAS collection

The ARAS dataset was collected from two real houses with multiple residents and included 27 different activities (Hande Alemdar, 2013). Each house was equipped with 20 different types of binary sensors, and a month of detailed sensor data and activity information was collected from each household. As a dataset collected in the real world, ARAS has higher significance than data collected in the laboratory environment, because it truly reflects the natural activity of people. And this dataset contains more human activity and information than any other dataset. Table 3.3 shows the activities and sensor infrastructure details of ARAS. The reason for selecting this dataset is because it contains higher precision and a larger total number of data points, so it has more profound significance for training machine learning models.

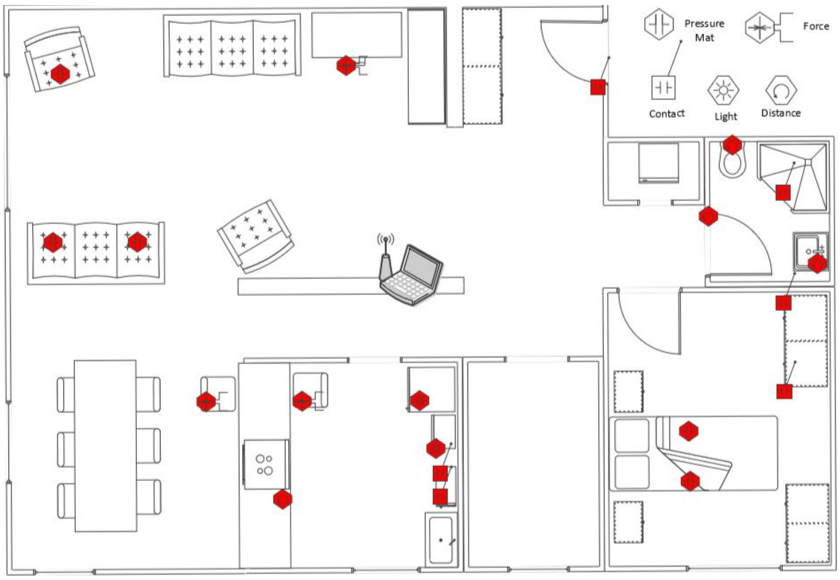
	Activities	Sensor
House A	-Other -Going out -Preparing breakfast -Having breakfast -Preparing lunch -Having lunch -Preparing dinner -Having dinner -Washing dishes -Having snack -Sleeping -Watching TV -Studying	1 wardrobe photocell 1 Convertible couch photocell 1 TV infrared receiver 2 Couch force sensors 2 Chair proximity sensors 1 Fridge photocell 1 Kitchen drawer photocell 1 Wardrobe photocell 1 Bathroom cabinet photocell 1 House DCS 1 Bathroom DCS 1 Shower cabinet DCS 1 Hall sonar distance 1 Kitchen sonar distance 1 Tap proximity sensor 1 Water closet proximity sensor 1 Kitchen temperature sensor 1 Bed force sensor
House B	-Having shower -Toileting -Napping -Using Internet -Reading book -Shaving -Brushing teeth -Talking on the phone -Listening to music -Cleaning -Having conversation -Having guest -Changing clothes -Laundry	2 Kitchen cupboards CSs 1 House DCS 2 Wardrobe DCSs 1 Shower cabinet DCS 1 Tap distance sensor 3 Chair force sensors 11 Fridge drawer photocells 2 Kitchen drawer photocell 2 Couch pressure mat 1 Bed pressure mat 1Armchair pressure mat 1Bathroom door sonar distance 1 Kitchen sonar distance 1 Closet sonar distance

Table 3.3 Activities and Sensor infrastructure of ARAS (Ifat Afrin Emi & John A Stankovic, 2015)

As far as we know, recorded and annotated datasets are very important for studying multi-resident activity recognition. In ARAS data set, binary sensors are selected, including infrared sensors, temperature sensors, pressure sensors and so on. From these, 7 species are placed in House A and 6 species are placed in House B. Refer to Fig.3.2. One set of the participants was two males and the other set was a married couple. In the process of collecting the data, no cameras of any type were used and no tags were installed on residents in consideration of privacy and other important factors. The entire dataset consists of a 22*86400 matrix stored in the file. House A and B files contain 30 documents in each of them. In the matrix, the first 20 columns display the sensor binary value, 0 or 1; the remaining two columns are active labels for Resident 1 and Resident 2. There were 2,177 activities in House A and 1,023 in House B. In order to better understand the sensor dataset, Fig.3.3 shows the activity duration distribution of the two residents in House A during 30 days. Through the time distribution map, we can get a clearer understanding of the lifestyle of each resident. In (Hande Alemdar, 2013), Hande et al. used HMM to model activities and sensor then reported the percentage of correctly classified labels for 30 days. The results show that the average accuracy of A house is 61.5%, and that of B house is 76.2%.

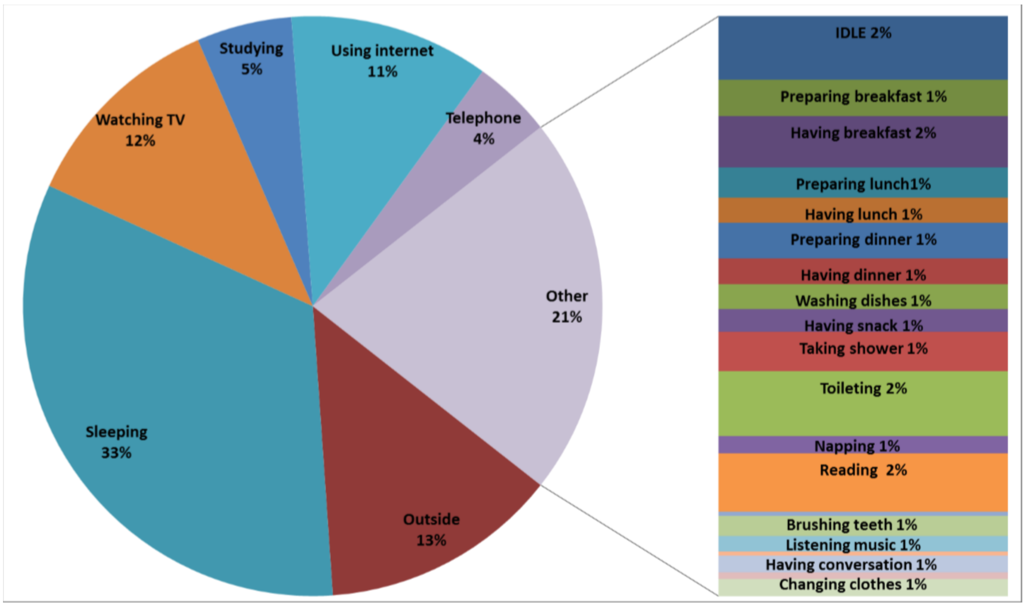


(a) House A

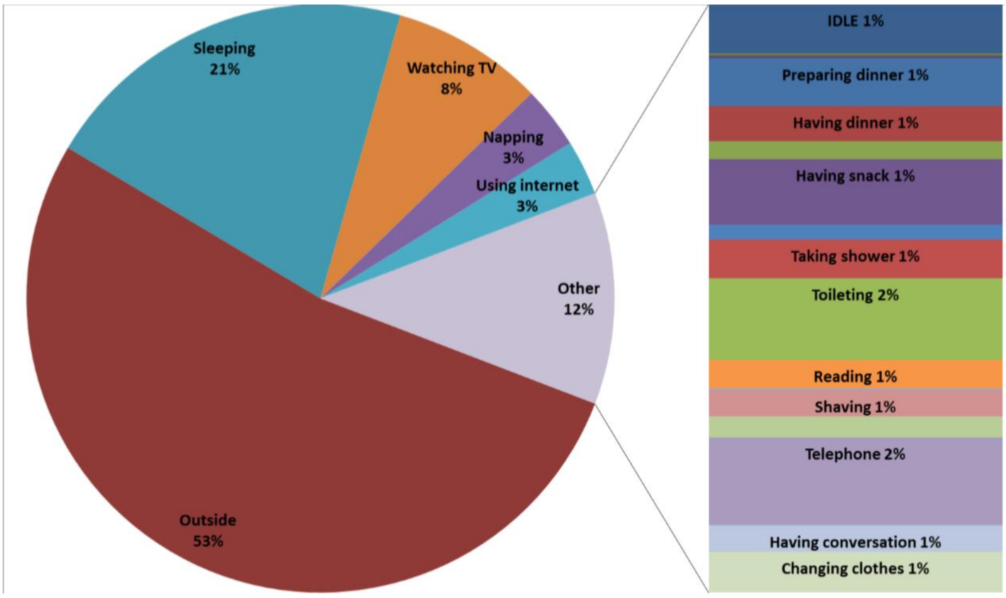


(b) House B

Fig.3.2 Layouts of the Houses(Hande Alemdar, 2013)



(a) Resident1



(b) Resident2

Fig.3.3 Activity duration distribution of House A (Hande Alemdar, 2013)

3.3 Data pre-processing

In the process of data mining, the data we use must conform to the following principles: give explicit meaning to attribute names and values as much as possible: remove unique attributes; remove repeatability and select the associated field correctly (Kotsiantis, Kanellopoulos, & Pintelas, 2006). Data preprocessing, as an important part of data mining, refers to the necessary processing such as audit, screening, sorting, transformation, protocol and summary before classification or grouping of the collected data. (García, Luengo, & Herrera, 2015) Through data acquisition, most of the raw data exhibit different kinds of problems such as irregular information, data not belonging to the same dimension, missing key data points, and data types do not meet the requirements. However, the quality of data determines the prediction and generalisation ability of the model directly. It involves many factors, including accuracy, completeness, consistency, timeliness, credibility, and interpretability. So we need to use different processing methods to organise the data. The quality of data and features determines the final effect of machine learning model.

In a previous study, Crandall and Cook et al. (Crandall & Cook, 2008) adopted the data preprocessing method to generate new features after extracting the features from the original data. They extracted more specific features from dates and times, such as "a day of the week" or "an hour of the day," to deal with data association problems. Their research suggests that the temporal features vary widely in capturing individual behavior. Moreover, when the time feature is "an hour of the day", the classifier performance of resident recognition is optimum. They also show that using different types of data characteristics can lead to different classification results such as: residents' daily living habits, laboratory environment or real homes. Among them, the hour is the most discriminating feature. In addition, in the comparing of Naïve Bayes and HMM, they found that feature extraction was not applicable to all classification models because the effect on HMM was not obvious.

In order to better determine the experimental effect of data preprocessing, Hus et al. started from three aspects: environmental data, room-level data and raw data (Hsu et al., 2010). Among them, environmental data is all sensor data collected in the whole house. Room-level data divides all environmental data into different rooms. If someone triggers the motion sensor in the room, the state of the sensor will be displayed as "on". However, according to the author's research, environmental data does not help to distinguish residents. When people are in a multi-resident environment, there is no way to know exactly who triggered the sensor based on the data. For room-level data, the author adopts the preprocessing method, and each room is represented by features. The results show that the accuracy of combining the raw data with the model is the highest, up to 64%. The other two kinds of data are around 30% accurate. They argue that if environmental data were used, all the information will have too much noise and this will affect the accuracy of the model. The low accuracy of room-level data is due to the small dependencies between its functions and related activities. To sum up, in this study, we used the raw data, followed by cleaning, integrating and reduction the data. More details are shown in the following subsection.

3.3.1 Data cleaning

The raw data we collected from the sensor contains redundant information and even erroneous data to some extent. Most of these data are caused by sensor faults or intermittent communication signals. So it is a necessary step to clean data for the training model. The purpose of the cleaning is not only to eliminate errors, redundancy, and data noise, but also to align datasets from different and incompatible rules (Hand, 2006). Fig.3.4 shows the cleaning process clearly. In the process of obtaining information and data in the real world, there will be various reasons for data loss and vacancy. In the case of lack of data, we usually adopt the following methods:

When we have a large dataset and lack multiple values in tuples, we can ignore tuples. Or if the missing rate of the variable is higher (more than 80%) and the coverage and importance are lower, we can directly refer to the deletion of this variable (García et al., 2015).

Secondly, when the missing rate and importance of variables are low, we can choose to fill in the attribute mean manually to make up the missing value. When the dataset contains noisy data, we usually use binning methods. Since the dataset we selected is sorted, the whole data is divided into segments of the same size for individual processing.

Fig.3.4 Data cleansing(trifacta, 2019)

Wilson and Atkeson (Daniel Wilson, 2005) collect raw data from binary sensors and RFID sensors in the sensor environment and pre-process the data using Bayes and particle filters. Experimental results show that Bayes filter can track users well despite noise. Particle filters work well with multiple users. Guettari et al. (T. Guettari, 2014) selected sensor datasets collected in different areas of the smart home and preprocessed them with median filters, which can largely avoid abnormal measurements.

There are two obvious problems with sensor data collected from intelligent environments: class imbalance and class overlap. These two factors may affect the accuracy of data analysis and classification. The class imbalance is mainly manifested in the fact that some activities occur more times than others, leading to insufficient compensation in the training process. Class overlap is represented by many repeated activities, leading to ambiguity. To overcoming the above two problems, Barnan et al. (Barnan Das, 2013) proposed a clustering-based under-sampling (ClusBUS), This method is applied to CASAS dataset to deal with the data overlap points with unbalanced class distribution. Experimental results show that this method can provide important information for minority groups.

3.3.2 Data reduction

When we need to deal with complex data, we spend a lot of time in data analysis and data mining. The advent of data reduction techniques can help us simplify datasets without compromising the integrity of the original data, and at the same time produce high quality data. There are some strategies for data reduction. For example, numerical reduction; Selection of attribute subsets (detecting and removing redundant and irrelevant feature dimensions); Aggregate operations are used when the dataset has multidimensional data. Data reduction works in two ways: by reducing the number of volumes and attributes (dimensions). In our experiment, the activity data of residents in the multi-resident data set of CASAS are composed of many days, so we segment the activity data, which will greatly reduce the sample of initial and transitional estimation. So a sequence is formed for each file and then a 10-fold cross-validation of the 26 activity sequence is performed.

3.4 Classification Models

Recognition classifiers are inseparable from machine learning. Machine learning can be divided into three categories: unsupervised learning, semi-supervised learning and supervised learning.

As a small branch of machine learning, HAR is often carried out by supervised learning. The classification model is trained by the training sample set of the known category, and then the unknown data category is identified based on the trained model, which is classified into one of the known categories. The training sample set consists of several training samples with the same format. Each training sample consists of a known action label and a feature vector which is composed of multiple feature attribute values. It can be expressed as as $<w_1, w_2, , \dots , w_n, c_i>$ where w_i represent the feature attribute value and c_1 represents the action label. In HAR, different recognition algorithms are selected, and the methods of building recognition models are also different. The evaluation of a recognition algorithm mainly includes the following four aspects:

(1) Recognition accuracy:

which means the ability of the model to correctly identify test data sets.

(2) Recognition speed:

which means the time spent on training recognition models and using them to identify.

(3) Robustness:

the ability of the model to accurately identify when the data contains noise points or missing values.

(4) Scalability:

the ability to efficiently train the model when the training set is large.

The activity recognition models used by the research are: Bayes Network, Naïve Bayes, Support Vector Machine, Decision Tree, Random Forest and Hidden Markov Model.

3.4.1 Bayes Network

Bayes network, also known as reliability network or belief network, as an extension of Bayes method, was proposed by Pearl in 1988. It is a theoretical model for uncertain knowledge representation and reasoning which is more effective. It has become a hot research topic in recent years. It avoids using joint probability to infer directly and instead uses the independent relationship between variables to divide jointly. The Bayesian network can qualitatively and quantitatively analyse the dependencies between features, and then establish the network structure for probabilistic reasoning. This classification process based on probability theory can effectively guarantee the correctness of reasoning structure. Bayesian network consists of two parts, network structure and parameter structure. The network structure is a directed acyclic graph, in which the nodes represent random variables, the edge between nodes represents the dependence between variables; parameter structure refers to the local probability dependence, and each node has a probability distribution. Therefore, the learning of Bayesian networks is divided into two processes: structure learning and parameter learning. In order to illustrate the two learning processes of Bayesian networks, this paper assumes that class variables are represented by C, and attribute change. The attribute variable set is $X = \{X_1, X_2,\ldots, X_n\}$(ASMA BENMANSOUR, 2016). The dataset $D = \{u_1,\ldots, u_d,\ldots, u_m\}$, where $u_d = \{x_1^d,\ x_2^d,\ldots, x_N^d,\ c^d\}$ represents the value of the N attribute variables and the metric of the category C in a data sample. Now we know a sample $x_d = \{x_1^d,\ x_2^d,\ldots, x_N^d\}$, and the classification prediction is based on its attribute variable to predict the value of its class variable C.

Structural learning of Bayesian networks refers to the process of generating the network topology structure with the highest fitting degree with the given data set. If the unknown network structure is represented by B_s, then the network structure B_s with the greatest posteriori probability $p(B_s|D)$ is required according to the known data set D. According to the Bayesian formula (Friedman, Geiger, & Goldszmidt, 1997) in Eq.3.1:

$$p(B_s|D) = \frac{p\ (B_s,D)}{p(D)} = \frac{p\ (D|B_s)\ p(B_s)}{p(D)} \quad (3.1)$$

Where $p(D)$ is the normalization coefficient related to dataset D, $p\ (D|B_s)$ is the boundary likelihood. Based on boundary likelihood find out logarithmic likelihood in Eq.3.2:

$$LL(B_s|D) = \sum_{d=1}^{N} log\, p_{B_s}\,(u_d) = \sum_{d=1}^{N} log\, p_{B_s}\,(c^d\,|x_1^d,\ x_2^d,\dots,x_N^d) + \sum_{d=1}^{N} log\, p_{B_s}(x_1^d,\ x_2^d,\dots,x_N^d)$$

(3.2)

The first item in Eq.3.2 is conditional logarithmic likelihood of class variables for given attribute variables, and the second item is joint distribution likelihood of predicted attributes for B_s. For classification, only the first item related to posterior probability is concerned.

3.4.1.1 Related Algorithm

It will be a NP problem to search for the best topological structure from all possible network structures, so it is common for Bayesian networks to use many heuristic algorithms as search algorithms for structure learning. There are K2, TAN, Hill-climbing, Simulated Annea-ling and so on. In this paper, we chose Tree Augmented Naïve Bayes (TAN), K2, HillClimber and TabuSearch used as a search algorithm for structural learning of Bayesian network classifiers.

- TAN (Tree Augmented Naïve Bayes)

TAN is based on the Naïve Bayesian Network, which relaxes the assumption that Naïve Bayesian requires conditional independence among attributes, and allows each attribute node to rely on, at most, one non-class node, that is, to allow each node to have a parent besides the class node.

Firstly, the mutual information function between any two attribute variables is defined as in Eq.3.3:

$$F(X_i, X_j/C) = \sum_{x_i,x_j,c} P\ (x_i, x_j|c)\ log\frac{P\ (x_i,x_j|c)}{P\ (x_i|C)P\ (x_j|C)} \quad (3.3)$$

This mutual information function represents the weight between any two attribute variables. The larger the weight, the greater the dependence between attribute variables X_i and X_j and, the more likely that the parent-child relationship exists. On the contrary, there is no parent-child relationship between the two variables. The process of establishing Bayesian network structure is as follows.

1) Calculate conditional mutual information between every two attribute variables $\mathrm{F}(X_i, X_j|\mathrm{C})$
2) Establishing the complete undirected graph of a node which is a set of attribute variables $\{X_1, X_2,..., X_n\}$. The mutual information of two attribute variables connected by either side is taken as the weight of that side.
3) The edges are sorted according to their weights, and the maximum spanning tree is composed of the edges selected in the order of the weights from large to small according to the principle that they cannot form a loop.
4) Choose a root node and set the direction of all edges from the root node outward to form a directed tree.
5) Adding class variable nodes and adding edges from class nodes to attribute nodes.

- K2 algorithm

K2 algorithm is the most famous fractional based on Bayes Network in recent decades. It uses greedy search to process the model selection. Firstly, a scoring function is defined to evaluate the structure of the network. Then starting with a network, the highest score node is selected as a parent node according to the node order and the number of parent node. There is no restrict on the number of parents node. When the score stops to increase, the node stop adding partents (Cooper & Herskovits, 1992).

- HillClimber

HillClimber is a heuristic search algorithm which tries to find the optimal solution within a reasonable time. Similar to K2 algorithm, it is also use greedy search to find the optimal node. The difference is that the HillClimber checks each neighbouring node one by one and then selects the best node as the next node. The algorithm has three types which are Simple HillClimbing, Steepest-Ascent HillClimbing and Stochastic HillClimbing (Rawat).

- TabuSearch

TabuSearch algorithm is the improvement of HillClimber algorithm. In HillClimber, we can only search the local optimal solution, and the obtain result is completely dependent on the relationship between the initial solution and the neighborhood. The improvement is proposed in TabuSearch. The outstanding characteristic of the algorithm is that it does not take the local optimal solution as the stop criterion (Brownlee, 2011).

3.4.2 Naïve Bayes

Naïve Bayesian algorithm is one of the most efficient algorithms in the field of machine learning and data mining. It is widely used in classification and other issues. At present, the Naïve Bayesian algorithm is also commonly used in HAR. Based on Bayesian theory, the algorithm has the advantages of fast training and recognition, lower error rate and fewer parameters to be estimated. It can handle both discrete and continuous data well. At the same time it is insensitive to missing data. The idea of the algorithm is simple and relatively easy to implement. The disadvantage is that the attributes of the feature vector need to be independent of each other.

In activity recognition, Naïve Bayesian is a typical classification model based on statistical method. It uses the prior information of training set and the sample data of test set to determine the posterior probability of time, and then decides the action according to the posterior probability.

The classification in activity recognition is described as: classifying the feature a$(w_1, w_2, \ldots,$ $w_n)$ representing the vector into the category c_i with the largest probability value, where $c_i \in (c_1, c_2, c_3, \ldots, c_m)$, $w_1, w_2, \ldots, w_n$ are the features of a.

Separate calculation of the probability $P_1, P_2, P_3, \ldots, P_m$ that a $(w_1, w_2, \ldots, w_n)$ belongs to the categories $c_1, c_2, c_3, \ldots, c_m$. Where P_m is the probability that feature combination a$(w_1, w_2, \ldots, w_n)$ belongs to action class c_m. Finally, through MAX $(P_1, P_2, P_3, \ldots, P_m)$ gets the action class of feature combination a. MAX$(P_1, P_2, P_3, \ldots, P_m)$ represent the maximum value of $P_1, P_2, P_3, \ldots, P_m$.

The definition of Bayesian formula is shown in Eq.3.4. Let the event group $(A_1, A_2, \ldots, A_n)$ be a complete event group, B be any event, and P(A_i)>0(i=1,2,…,n), P(B)>0, then:

$$P(A_i/B) = \frac{P(A_i)P(B|A_i)}{\sum_{j=1}^{n} P(A_i)P(B|A_i)} \quad (3.4)$$

According to the definition of Bayesian formula, the denominator $\sum_{j=1}^{n} P(A_i)\,P(B|A_i)$ is a fixed value. To find MAX(P$(A_i$|B)):

Under B condition, the maximum value of the probability P$(A_i$|B) that A_i occurs, where A_i belongs to event group $(A_1, A_2, \ldots, A_n)$. So it only needs to calculate the molecular

$P(A_i)P(B|A_i)$. For action recognition classification, the formulas for finding P $(c_i|w_1, w_2, ..., w_n)$ is shown in Eq.3.5.

$$P(c_i/w_1, w_2, \dots, w_n)=\frac{P(w_1,w_2,\dots,w_n|c_i)P(c_i)}{\sum_{j=1}^{m} P(w_1,w_2,\dots,w_n|c_j)P(j)} \tag{3.5}$$

The definition of P(c_i): In the training set of feature vector, the probability that the feature vector belongs to the action classes c_i is called a priori probability. P $(w_1, w_2, \dots, w_n|c_i)$ implies that the conditional probability of generating the classified feature vector a (w_1, w_2, ..., w_n,) from the class c_i. where the denominator $\sum_{j=1}^{m} P(w_1, w_2, \dots, w_n|c_j)P(\text{j})$ is the joint probability of all action class. According to the previous analysis of Bayesian formula, denominator is a fixed value for all given classes.

It is required to find out which class is the feature vectors to be categorised belong to, and ultimately convert them into solving c_i, $\in(c_1, c_2, c_3,..., c_m)$. The maximum value of formula P $(w_1, w_2, \dots, w_n|c_i)\ P(c_i)$ as shown in Eq.3.6.

$$\max_{c_i \in C} P\ (w_1, w_2, \dots, w_n|c_i)\ P(c_i) \tag{3.6}$$

Therefore, assuming that feature words are independent of each other, formula P $(w_1, w_2, \dots, w_n|c_i)\ P(c_i)$ can be expressed as Eq.3.7

$$P\ (w_1, w_2, \dots, w_n|c_i)\ P(c_i)=\prod_{k=1}^{n} P(w_k|c_i)P(c_i) \tag{3.7}$$

Finally, Eq.3.6 is converted to Eq.3.8

$$\max_{c_i \in C} \prod_{k=1}^{n} P(w_k|c_i)P(c_i) \tag{3.8}$$

In the process of continuous multiplication calculation, because the probability value of each feature is relatively small, the dimension of feature is relatively high, and even have more stages of multiplication. So there will be underflow. In order to prevent the above-mentioned situation, the logarithmic operation of Eq. 3.8 is carried out, and Eq.3.8 is converted to Eq.3.9.

$$\max_{c_i \in C}\{\ln P(c_i) + \sum_{k=1}^{n} \ln P(w_k|c_i)\} \tag{3.9}$$

In action classification, each feature records **a** consists of feature **w**, which is expressed as a $(w_1, w_2, \dots, w_n)$. The specific flow chart is shown in Fig.3.5.

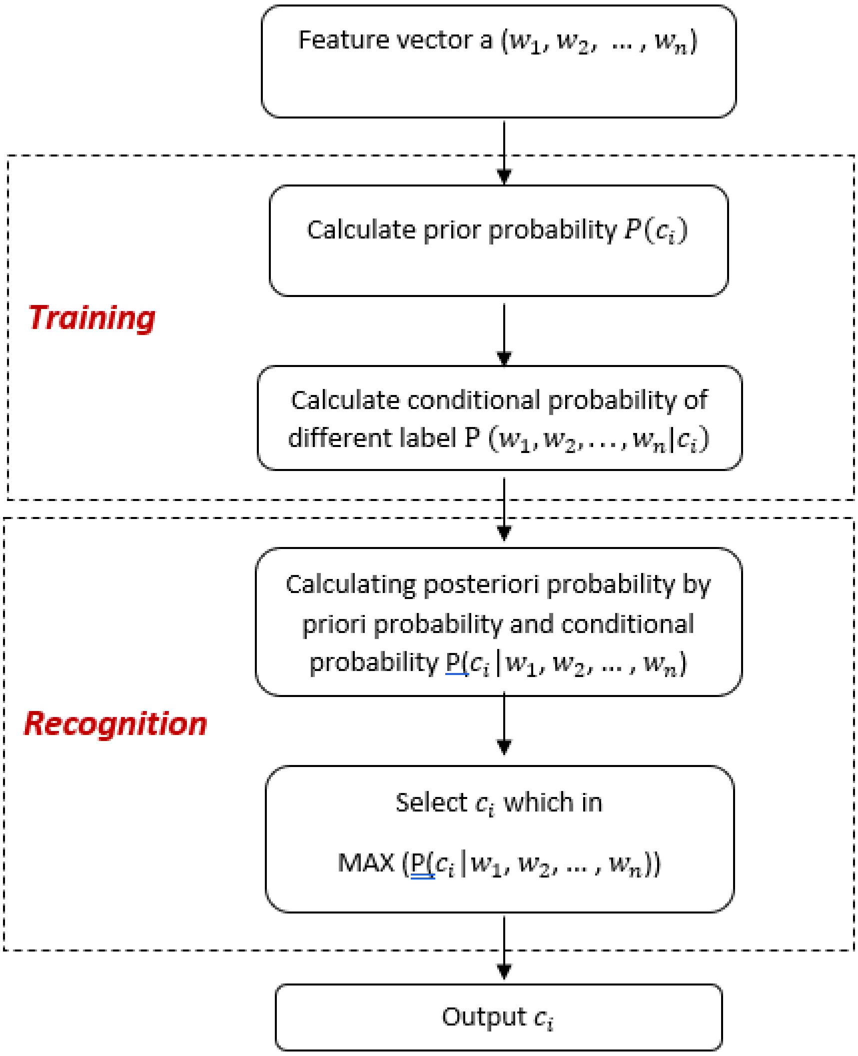


Fig.3.5 Naïve Bayes workflow in activity recognition

Crandall and Cook et al. (Crandall & Cook, 2008) applied raw data to Naïve Bayes model for data association, but achieved low performance. Through subsequent studies, we know that is due to the imbalance of some training datasets, Naïve Bayes usually allocates activities to residents who generate most of the sensor time during operation. Later in the process, Naïve Bayes classification ability is significantly improved after adding specific time features such as "one hour in a day".

3.4.3 Support Vector Machine

Support Vector Machine (SVM) is based on statistical learning theory VC dimension theory and structural risk minimization principle. According to the limited sample information, it seeks the best compromise between learning ability and model complexity to obtain the best generalisation ability. Support Vector Machine (SVM) seeks the global optimal solution, so it has more advantages than other statistical learning algorithms. It is aimed at the problem of two label classes classifications and has been extended to multilabel classifications. For the two-label class classification problem, the algorithm process is as follows:

Assuming that the training set can be partitioned by a hyperplane, there exists (w, b) such that

$$(wx_i + b) > 0,\ y_i = 1,$$
$$(wx_i + b) < 0,\ y_i = -1;$$

The ultimate goal of classification is to find the suitable (w,b) to acquire a best accuracy. As shown in Fig.3.6.

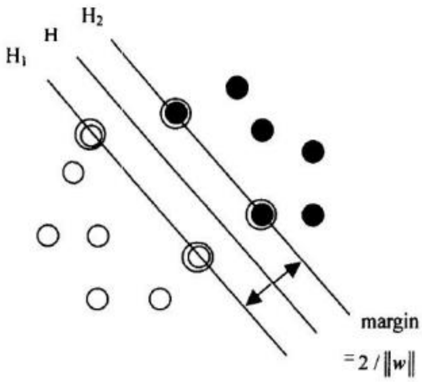


Fig.3.6 Classification diagram of SVM

The black solid point and the white hollow point represent two types, respectively, H is a separate type of separated hyperplane, H1 and H2 are the samples closest to the separated hyperplane and parallel to the plane separating the hyperplanes. The distance between them is called the classification interval. The purpose of the support vector machine is to find the optimal hyperplane, so that the sample classification interval is largest. There are also $y_i(\mathrm{w}x_i + b) \geq 1$, i=1, 2,..,n, and the classification interval is 2/||w||. To maximise the classification interval, then ||**w**|| is the smallest, which is equivalent to finding 1/2||w||2, so the following optimisation problem is obtained in (Yu & Kim, 2012) Eq.3.10:

$$\underset{w,b}{Min}\frac{1}{2}||w||^2 s.t.\ y_i(wx_i + b \geq 1, i=1,2,...n \qquad (3.10)$$

By applying Lagrangian duality, the optimal solution of the above problem can be obtained by solving the dual problem. The Lagrange function is expressed as follows in Eq.3.11:

$$L(w,b,\alpha) = \frac{1}{2}||w||^2 - \sum_{i=1}^{n} \alpha_i y_i (wx_i + b) + \sum_{i=1}^{n} \alpha_i \qquad (3.11)$$

Where α_i>0 is the Langeland coefficient. The training process of the SVM is shown in Fig.3.7.

Extending SVM from two label classes classifications to multilabel classifications is to construct a series of two label classes classification machines, in which each of the two classification machines can classify one of the remaining types and then classify them. The key of SVM is the selection of kernel function. Vector sets in low-dimensional space are usually difficult to partition and need to be mapped to high-dimensional space, but that will increase the computational complexity. In order to solve this problem, we usually need to introduce the kernel function. The commonly used kernels are:

(1) Linear Kernel function: k(x, x_i) = <x, x_i>

(2) Polymerization Kernel function：K(x, x_i) = $(\gamma < x, x_i > +r)^d$, d represents degree，r represents coef().

(3) Gauss Kernel function: K(x_i,x_j)= $\exp(-\gamma||x_i, x_j||)^2$, γ represents the keyword gamma and greater than 0.

(4) Sigmoid Kernel function: K(x, x_i)=tanh $(\gamma < x, x_i > +r)$, r represents coef（）

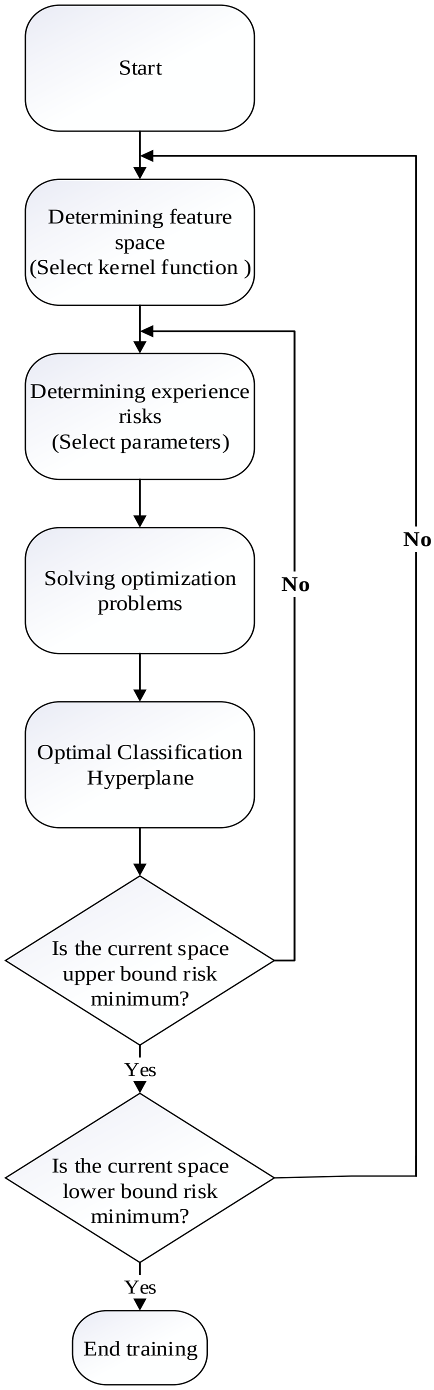


Fig.3.7 Training flow of SVM

3.4.4 Decision Tree

As a statistical model, the decision tree is generally responsible for supervising the classification and prediction of training data (Trevor Hastie, 2009). It is a model to display decision rules and classification results with tree data structure. As an inductive learning algorithm, it focuses on transforming known instances that seem to be disordered and disorderly into a tree model that can predict unknown instances by applying consistent rules.

Fig.3.8 shows the overall structure found in a decision tree, it consists of three parts: decision node, branch node and leaf node. Each node represents a feature and corresponding decision rules. The top node is the root node. At this time, all the samples are together, and after passing through the node, they are divided into each sub-node. Each sub-node uses new features to make further decisions until the final leaf node. A directed acyclic tree consists of decision nodes, branches and leaves. Decision nodes are usually used to represent a problem or decision. In classification problems, they usually correspond to a characteristic attribute of the object to be classified. Branches often correspond to a new decision node or leaf node of a tree. Leaf nodes are often used to represent the classification results. The process of traversing the decision tree from top to bottom will correspond to a test output at the node. Different test outputs of the node often correspond to different branches until they reach the leaf node. The leaf node stores the decision result (Melo & Lee, 2018). The above description is the classification process of the decision tree, which uses several characteristic attributes to decide the category of the sample.

The core idea of a decision tree algorithm is to construct the decision tree; this is constructed recursively from top to bottom by a greedy algorithm. To avoid over-fitting caused by noise points, pruning is needed. Pruning can be divided into two types: pre-pruning and post-pruning. Pre-pruning uses stopping the growth of decision tree in advance to prune, and post-pruning is to remove some subtrees through specific criteria in the generated over-fitting decision tree, and to finally generate a simplified version of the decision tree.

Decision trees are applicable to both numerical and nominal types (discrete data, where the results of variables are only valued in a finite target set). They can read data sets and extract rules contained in several columns of data. Especially in classification problems, the decision tree model has many advantages, such as low computational complexity, convenience and efficiency. The decision tree can process data with irrelevant characteristics and easily construct rules that are easy to understand and which are usually easy to explain and understand. The advantage makes the decision tree more capable of implementation in real scenarios, because it saves a lot of time for in training (Carolin Strobl, 2009; Trevor Hastie, 2009).

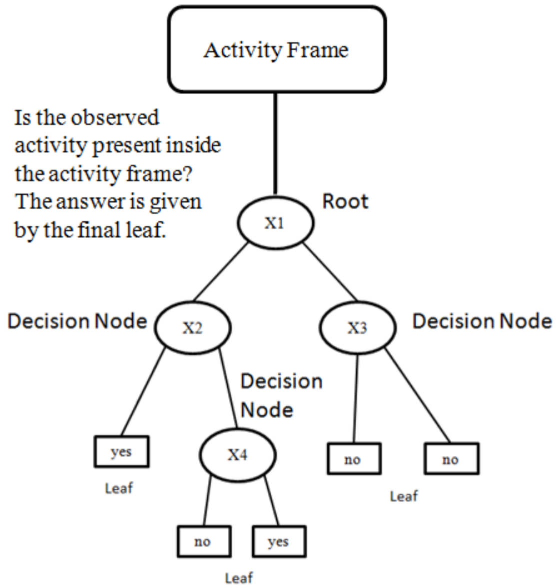


Fig.3.8 Overall structure of decision tree (Chiu, Yu, Liaw, & Chun-Hao, 2016)

There are many decision tree algorithms, such as CLS algorithm, SLIQ algorithm, ID3 algorithm, CART algorithm, C4.5 algorithm, SPRINT algorithm, etc. Among them, ID3 and C4.5 are commonly used. ID3 and C4.5 are based on information theory. The difference is that ID3 takes information entropy as the measurement standard, while C4.5 takes information gain (T. Guettari) as the measurement standard. C4.5 algorithm is improved based on ID3 algorithm, which has higher recognition accuracy and faster calculation speed. Therefore, the C4.5 algorithm is chosen.

The C4.5 algorithm uses information gain rate to select decision attributes, overcomes the shortcoming of ID3 when using information gain to select attributes with more values, and can process non-discrete data as well as incomplete data. Information gain rate is defined as:

$$GainRatio(S,A)= \frac{Gain(S,A)}{SplitInfo(S,A)}$$

Among them, Gain (S, A) is the information gain, and the calculation formula is as follows in (Anyanwu & Shiva, 2009; Safavian & Landgrebe, 1991) Eq.3.12:

$$Gain(S,A)=Entropy(S)- \sum_{v\in Values(A)} \frac{|S_V|}{S} Entropy(S_V) \qquad (3.12)$$

In Eq.3.9, **v** is one of the values of characteristic attribute A, and $|S_V|$ is the number of training samples when attribute A = v. Entropy (S) is the information entropy, and the calculation formula is as follows in Eq.3.13:

$$Entropy(S)=\sum_{i\in C} -p_i log_2 \, p_i \qquad (3.13)$$

In Eq.3.10, S is the total number of samples, C is the set of all actions of S, and p_i is the ratio of training set with action i to training set S.

SplitInfo(S, A) denotes the breadth and uniformity of the sample set S split according to attribute A. The calculation formulas are as follows in Eq.3.14:

$$SplitInfo(S,A) = -\sum_{v \in Values(A)} \frac{|S_V|}{S} log_2 \frac{|S_V|}{S} \quad (3.14)$$

In Eq.3.14, S is the total number of training samples, v is a value of feature attribute A, and the number of samples when feature attribute A is a value of v. Each time, the attribute with the largest information gain rate is selected as the decision node until the pruning condition is satisfied, and finally the decision tree model is constructed. Pseudo-code is shown in Fig.3.9.

Algorithm: C4.5

Input: Training samples: collection of candidate attributes Attributelist
S represents the current sample set, and the current candidate attribute set is represented by A.
Output: A Decision Tree
Step:
1. Create the root node N;
2. IF S belongs to the same class C, then N is returned as leaf node and marked as class C;
3. IF Attributelist is empty or the number of samples remaining in S is less than a given value;
4. Then N is returned as leaf node, mark N is the most frequent class in S;
5. FOR each Attributelist
6. Calculate the information gain ratio;
7. The test attribute of N test.attribute = attributelist has the highest information gain ratio attribute;
8. IF test attribute is continuity
9. Then find the segmentation threshold of this attribute;
10. For each new leaf node of root node N
{
11. IF the sample subset S' corresponding to the leaf node is empty;
12. the leaf node is split to generate a new leaf node, which is marked as the most frequent class in S;
13. Else perform C4.5 formtree (S', S'.attrbutelist) on the leaf node and continue to split it;
}
14. Calculate the classification errors of each node and prune them.
15. Returns the root node N

Fig.3.9 Pseudo code of C4.5 algorithm (Prossegger & Bouchachia, 2014)

3.4.5 Random Forest

The RF classifier belongs to one of the newest algorithms. It is a non-probabilistic decision tree-based classifier. The RF algorithm is employing a large ensemble of decision trees to deal with both regression and classification tasks. The strengths of the RF algorithm are remarkable classification performance along with relatively simple training and tuning. The RF algorithm uses a set of classification trees to solve the classification problem, which is based on a recursive binary tree and only considers a random subset of available datasets. For each node of each tree, we use the method of randomly selecting function subset to find the best segmentation point. The optimal segmentation point is determined using GINI index value. To classify the new datasets, they are randomly entered into each tree, so that the tree's majority of voting rights determine which tags are assigned. RF algorithms are also used in feature selection. When a particular feature is removed from the feature set in the tree, the average accuracy is reduced. If the accuracy of a feature is not significantly decreased when it is excluded, the feature does not play an important role in the whole dataset. In this project we used the function "RandomForestClassifier" in the WEKA learn package for constructing the RF classifier.

3.4.6 Hidden Markov Models

The hidden markov Model (HMM) is a generative model. It is used probabilistic to deal with sequence problem. The algorithm consists of hidden states and observation variables. Fig.3.10 shown the structure of the HMM model. At each time point t, a hidden state y_t related to an observable variable x_t. There are two dependency assumption in the HMM. Firstly, each hidden state y_t only depends on the previous hidden state y_{t-1}. Second assumption is each observable variable x_t only depends on a hidden state that observable variable corresponded at time point t (Eddy, 1996).

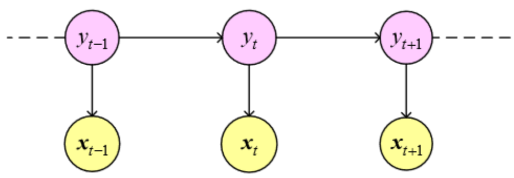


Fig.3.10 Graph structure of HMM

Therefore, HMM can be modeled using three probability distributions which are transition probability distribution p($y_t|y_{t-1}$), initial state p(y_1), and the last one is observation distribution p($x_t|y_t$). The joint probability of the HMM model as following equation:

$$p(y_{1:T}, x_{1:T}) = p(y_1)P(x_1|y_1)\sum_{t=2}^{T} p(y_t|y_{t-1})p(x_t|y_t)$$

When a labelled dataset $\{(x_t\ \ , y_t)\}$ is given, t = 1, 2, …, T, the distribution of the initial state $\pi_{\ i} = p(y_1\ \ = i), i = 1, 2, ..., K$ can be calculated, which represents the state of the HMM when the first sensor event is appeared. For a state (activity) a, the instance ratio of activity label a is calculated. The transition probability $a_{ij} = p(y_t\ \ = j|\ \ y_{t-1} = i), i, j = 1, 2, ..., K$, represents the possibility of changing from a given state to any other state in the model while capturing temperature. In any two states a and b, the transition probability calculation from state a and state b is the ratio of the number of instances with active labels a and b to the total number of instances. The observation distribution is factorized as:

$$p(x_t|y_t = i) = \prod_{n}^{N} p(x_t^n|y_t = i)$$

each sensor observation is modeled as an independent Bernoulli distribution as follows:

$$p\left(x_t^n\middle|y_t = i\right) = (u_{ni})^{x_t^n}(1 - u_{ni})^{1-x_t^n}$$

where $u_{ni} = p(xnx^n / y = i)\ (i = 1, 2, ..., K, n = 1, 2, ..., N)$ is calculated by finding the frequency of the nth sensor event observed for each activity.

It is possible to separate activity data when the activities are sequential, and then the HMM model can be created for each activity. However, the HMM method cannot reflect the interlacement of the activities. Thus the method is not suitable for interleaved activities. Another problem of HMM method is how to find the optimum number of hidden states for an HMM model corresponding to an activity. Creating an HMM for each activity would lead to having the same sensor model for each activity, but the number of hidden states for each activity is unknown. In fact, the authors in Khanetal (Khanetal, 2012) proposed to find the optimum number of hidden states through accuracy. Also they suggested using techniques applied for Hierarchical Dirichlet Process HMM (HDP-HMM) and infinite HMMs.

In some cases, even if the complex activity is decomposed, its sub activities are still very complicated. Thus, Sub activities cannot directly observable (hidden). For example, the activity “Prepare a dinner” can be divided into the activity “prepare drink” and the activity “cook” and even sub-activities can be divided into smaller one. Therefore, each individually trained HMM model can be combined to build a global HMM to model the entire activity. Therefore, a hierarchical graphical models (e.g., Hierarchical HMM or Abstract HMM) seems to be more suitable for this situation (Lee & Cho, 2011). In general, we are able to separate the activity and establish an HMM model for each activity when activity is sequential. However, HMM is not appropriate for the situation if the activity is of the interleaved or collaborative type because the interlacement of activity is possibly ignored. In some situation, there are some sub-activities still not observable directly when the complex activity is disintegrated (Khan, Karg, Hoey, & Kulic, 2012). For example, the activity "tidy the room" can be divided into two activities "mop the floor" and "take out the trash" and each of them also includes some sub-activities. Therefore, different activities trained using HMMs individually could be integrated to establish a global HMM and these hierarchical HMM are more suitable in these situations. To overcome these limitations, some variants of HMM have been proposed to recognise multi-resident activities (A. Benmansour et al., 2015).

3.5 Evaluation Method

Generally speaking, our choice of HAR classification algorithm is supported by previous empirical evidence. Most studies use cross-validation or statistical testing to compare the performance of classifiers against specific datasets. Assuming that for the classification of n classes, we can organize all the classification results in a confusion matrix $M_{n\times n}$. In this matrix, the element M_{ij} is the number of instances of class **i** but it is actually classified as class **j**.

We can get the following values from the confusion matrix in the binary classification problem:

- True Positives (TP): The number of positive instances that were classified as positive.
- True Negatives (TN): The number of negative instances that were classified as negative.
- False Positives (FP): The number of negative instances that were classified as positive.
- False Negatives (FN): The number of positive instances that were classified as negative.

The accuracy is the indicator we often use to summarise classified performance, which is defined as:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN}$$

The precision which is positive predictive value. It is the ratio of correctly classified positive instances to the total number of classified positive instances:

$$Precision = \frac{TP}{TP+FP}$$

The recall, also referred to true positive rate. The formula of recall value as follows:

$$Recall = \frac{TP}{TP+FN}$$

When we get the value of precision and recall, then combines them in a single value then get the F-measure:

$$F-measure = 2\frac{Precision*Recall}{Precision+Recall}$$

We calculate the precision, recall and f-measure for each activity separately and compute the average over of all classes according to the definition given in (van Kasteren et al., 2011a). For HAR datasets, activities are imbalance because there are many repetitive in every day. For example, sleeping, cooking and so on. Therefore, if the classification and recognition effect of the dominant class is better, the overall recognition performance will also be improved, but the recall rate is very low. In order to demonstrate the performance of each HAR's different activities, we chose accuracy, recall, precision and F-measure at the same time.

3.6 Statistical Test

The process of data analysis is examining the data systematically with the aim of integrating useful information and evaluating experimental results. In general, when we collect sample data through observational studies or experiments, statistical inference becomes a powerful tool for researchers to evaluate results (Nigam,2018). The significance test plays a key role in experiments, allowing researchers to determine whether their data support or reject the null hypothesis, thus whether an alternative hypothesis is acceptable. Generally speaking, the inference method of statistical hypothesis test is as follows: first, we need to have a preliminary research hypothesis, calculate the test value of test statistic from the observation results, then calculate the p value, that is the probability of sampling test statistic under the null hypothesis. If and only if the p value is less than the threshold value (i.e. 0.05), the null hypothesis is rejected (Ryan, 1960).

There are many types of statistical tests. If the experimental data are normally distributed, parametric tests should be used; otherwise, non-parametric tests should be used. Currently, when we want to test correlations, usually use Pearson correlation or chi-square to find correlations among variables. When it comes to mean differences between variables, paired t-test, independent T-test, and ANOVA (Analysis of variance) are usually used. The independent sample t-test and ANOVA are used to compare the mean values of independent groups and to test whether the variances are equal. ANOVA usually used the ratio of the between group variance to the within variance to determine whether there was a significant statistical difference for 3 or more independent groups. The difference between ANOVA and T-test is T-test used to compare the means of two independent groups.

Regardless of which statistical method we choose, we make inferences from sample data through hypothesis testing. Some key terms, such as null hypothesis H_0, usually have no difference between groups and no correlation between variables. Alternative hypothesis H_1 is usually used to investigate the problem, depending on whether the test is one two tailed. The significance level is an important term, and when the hypothesis is true, we usually set the value to 5%, which means that invalid hypothesis is rejected. The last one is the p value, and when we calculate the p value using different probability distributions, the significant result is when p is less than 0.05 (Martyn Shuttleworth, 2008).

3.7 Summary

In this chapter, we first introduced the source and application of experimental data in detail and collated the information from two main datasets. Secondly, according to the situation of each data set, different methods were adopted to preprocess the data including the cleaning of raw data, data integration and data reduction. The last part is the classification part. We described different methods for the classification of behaviour recognition. These included Bayes Network, Naïve Bayes, SVM, Decision Tree, HMM and Random Forest. Each machine learning model was studied in depth and the experimental results were presented.

Chapter 4 Results and Discussion

In this chapter, we mainly show the experimental results and introduce the specific model and implementations of each activity recognition datasets. Each model mentioned in Chapter 3 is applied to different datasets. We show the best state applied to different models in different datasets, that is, to give the highest accuracy. We also evaluate the experimental results in turn from the perspectives of datasets and models. The experimental configuration and data arrangement for the experiment will be described in this chapter. We discuss the findings and different methods of evaluating four experimental results.

4.1 Introduction

Activity recognition is a process of recognising human behavior from data captured by different types of sensors. At present, it has been applied in many aspects, such as daily activity monitoring, medical care and assisted life of the elderly. Nowadays, there are lots of research on simple activity modeling. The research of complex activities has only begun in recent years. Based on pervasive computing and intelligent environment, we need to find appropriate models to solve data association problems. Appropriate sensors should be used to collect data so as to capture the activity details of all residents. Data association refers to the recognizing of residents who trigger sensors. It requires mapping the perceived data to the actual startup data. In multi-residential environment, if we can recognize who triggered the event, then we can track residents' activities effectively and accurately.

In order to discover and recognize active learning, many algorithms of machine learning have been designed, improved and rewritten. There are many challenges in building models of human activity recognition, such as monitoring which residents trigger sensor events in a multi-resident environment (i.e., resident identification problems).

In this Chapter, we will introduce the experiments and discuss their results in order to compare the various classification models described in the previous chapter for multi-resident HAR. We do experiments using each method and adjust the parameters to get the highest classification accuracy.

4.2 Experimental Environment

- Hardware experimental environment:

The experiment is run on a desktop with 4 core i5 CPU 3.0GHZ.

- Software experimental environment:

We used MATLAB and Weka for training and testing of the models. MATLAB used for modeling HMM and data pre-processing. For the other model we use Weka. The particular software version we used is shown below:

Software name and version	Function
MATLAB 2018b	HMM modelling, data pre-processing
WEKA 3.8.3	BN, NB, SVM, DT and RF modelling

4.3 Experiment Results

In our experiment, we used six different general machine learning methods to classify the activity recognition datasets. The optimum state of the model can be achieved by repeatedly adjusting parameters. There are statistical deviations in each dataset. Deviation is the tendency of statistical data to overestimate or underestimate parameters. Firstly, we adjusted the bias parameter before adjusting the model. In this experiment, we selected 5 bias parameters of 0, 0.25, 0.5, 0.75 and 1 with spacing of 0.25. On the basis of each model, after adjusting each data set, the best performance parameters are selected as shown in Table 4.1.

Referring to Table 4.1, we can easily see the bias parameter and the model's parameter after parameter tuning with Bayes Network. In the Bayes Network, we chose different algorithms which are TAN, K2, HillClimber and TabuSearch. When we use these four algorithms to process dataset separately, the accuracy is totally different. For dataset1, the optimum algorithm of Bayes Network is TAN. Without the change of bias parameter, the accuracy is lowest. Then, in the process of gradually increasing the bias parameter value, the results of different algorithms show slight changes. Among them, when the bias parameter is 0.5 and using TAN to classify dataset1, the accuracy is highest at 78.84%. When we set up the same condition with other three datasets, the accuracy is not the same with dataset1. So for dataset2, the optimum bias parameter is 0. The accuracy is lower than dataset1 and is 71.12%. As we can see, for all datasets, the optimum algorithm is TAN and K2 which outperforms the other two algorithms overall. In each row, the best of the classifier results is displayed in Fig.4.1. Specifically, for each dataset, the different parameters of classifiers with the highest performance is highlighted. The experimental results show that when we use the same parameter estimation procedure to compare the TAN and K2 algorithms, the classification accuracy is not affected greatly. But in the experimental process, TAN takes less time because of its low computational complexity.

Model	NameOfParameter	RangeOfParameter	BiasParameter	Dataset
Bayes Network	searchAlgorithm	TAN(-S BAYES)	0.5	1
		TAN(-S BAYES)	0	2
		TAN(-S BAYES)	0.75	3
		K2(-P 1 -S BAYES)	0.25	4

Table 4.1 Parameter tuning result of Bayes Network

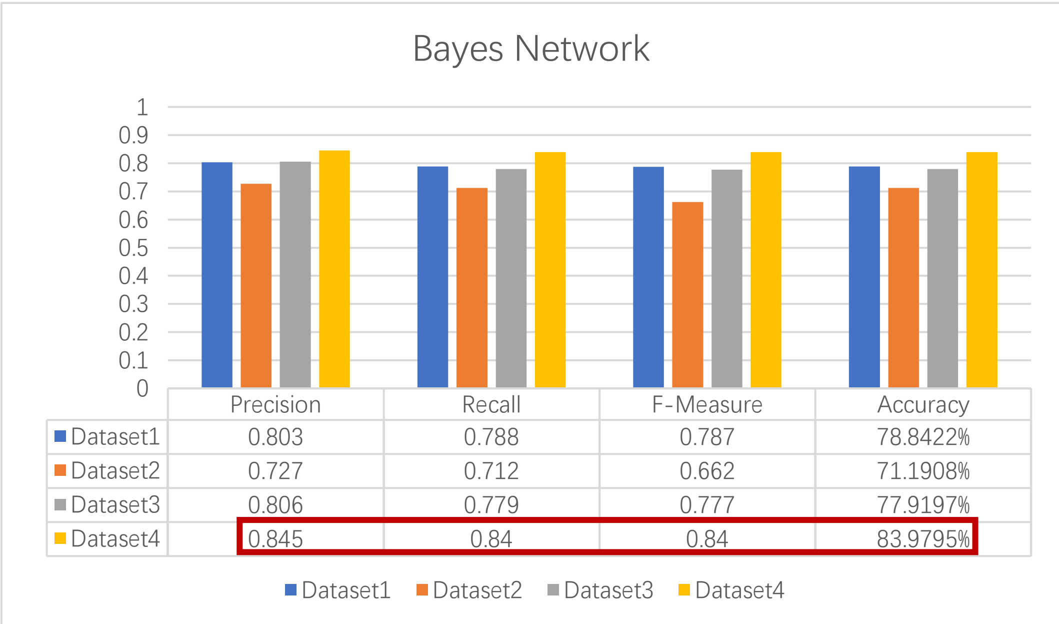


Fig.4.1 Result of Bayes Network

The second model is Naïve Bayes. Compared with other models, it is a simple and efficient machine learning classification model with high utilization rate. As a probability classifier, Naïve Bayes classification principle is maximum posteriori decision. In Bayesian environment, the classification model is suitable for high-dimensional situations. In many practical applications, even if the independence rules are not satisfied, NB algorithm can still show good performance, so it is a particularly popular classification method.

Model	NameOfParameter	RangeOfParameter	BiasParameter	Dataset
Naïve Bayes	useKernelEstimator	FALSE	0.25	1
	useSupervisedDiscretization	TRUE		
		TRUE	0	2
		FALSE		
		FALSE	0.75	3
		FALSE		
		FALSE	0.25	4
		TRUE		

Table 4.2 Parameter tuning result of Naïve Bayes

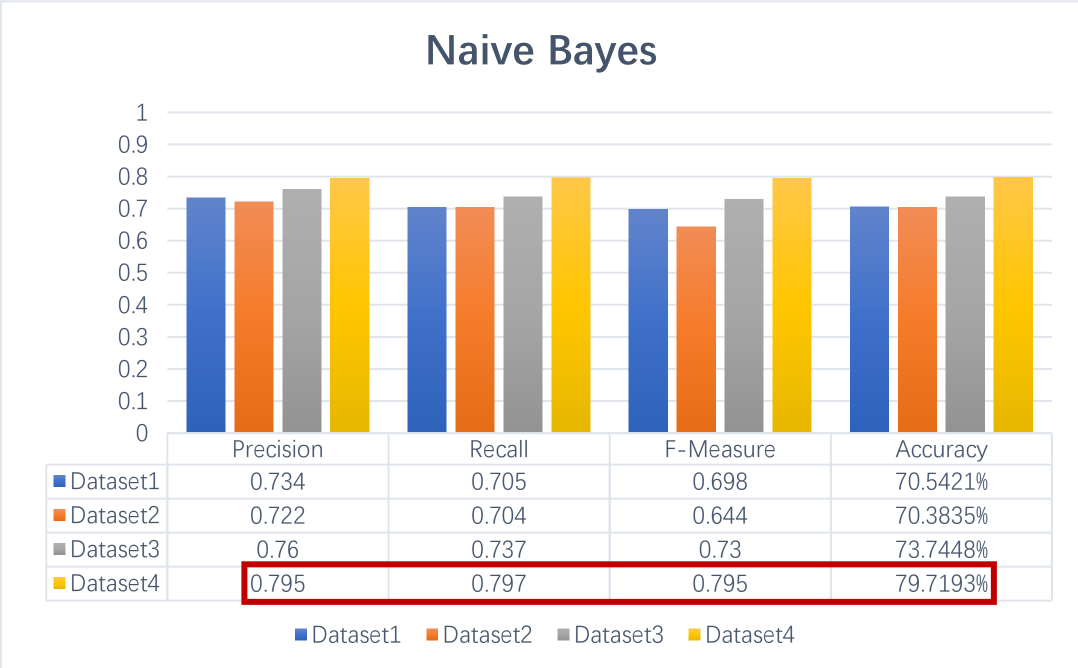


Fig.4.2 Result of Naïve Bayes

The result presented in Fig.4.1 and Fig.4.2 indicate that Bayes Network outperforms NB overall. The different models used in the experiment explain some of the disparities of experimental conclusion to some extent. We think that parameter estimation has significant effect. For Naïve Bayes and Bayes Network, we use parameter estimation to present different results. As expected, although parameter smoothing has little effect on the performance of NB, it does improve the performance of TAN algorithm to a certain extent because zero probability estimation may occur in more complex structures.

For the SVM model, we selected different kernel types; this is the Gaussian kernel which is special case of radial basis, linear, polynomial and sigmoid. All of the different kernel functions help map the data to a higher dimension where the data is separable. The best kernel type of these four datasets is a linear function. Linear kernel is usually used when data are linearly separable.

In other words, the selected data can be separated by a single line. Linear kernels, as one of the commonly used function, are mostly used for specific datasets with a large number of features. On the other hand, the training time of linear kernel function is faster than any other kernel of SVM. From Fig.4.3 we know the best performance of SVM is dataset1 which is 88.57%. And for bias parameter, each dataset has different value. But for dataset4, the accuracy is almost the same with dataset1.

Model	NameOfParameter	RangeOfParameter	BiasParameter	Dataset
SVM	kernalType	linear	0.5	1
		linear	0	2
		linear	0.75	3
		linear	1	4

Table 4.3 Parameter tuning result of SVM

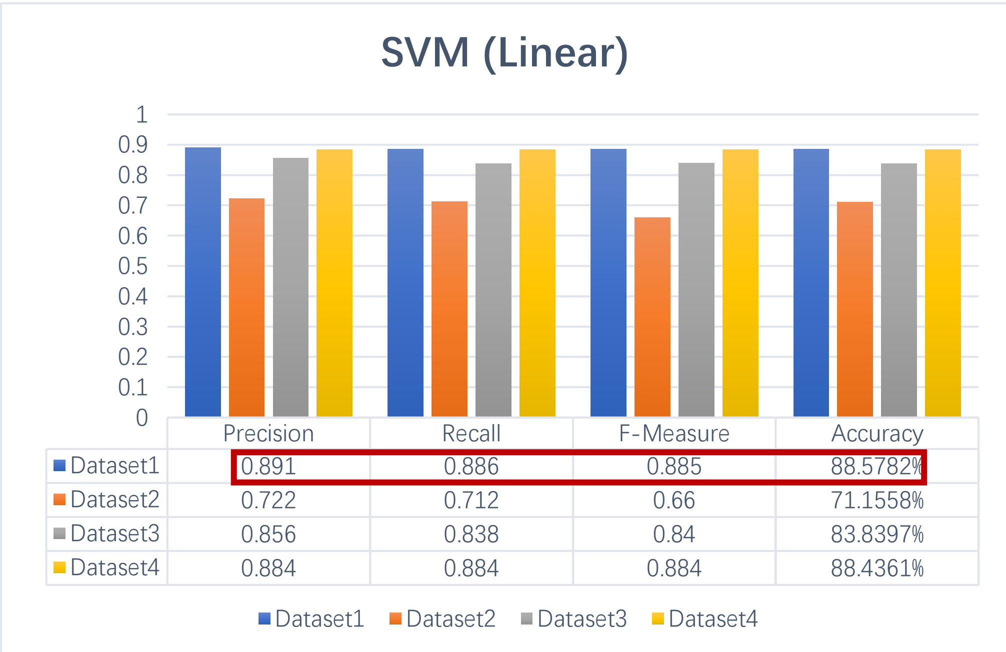


Fig.4.3 Result of SVM

As a supervised learning method for inductive reasoning, decision tree model can be used to approximate the objective function representing discrete values and represent it as a tree structure. It classifies an instance of a dataset from the root node to the leaf node, and then each node processes the value of an instance attribute. So, using the basic ideas from Chapter 3, we adjust the minimum number of objects and get the different results by experimenting from this model. As we can see in Fig.4.4, the highest accuracy is 83.47%, which is dataset4. The biggest difference between the two algorithms is that SVM uses the kernel trick to turn a linearly non separable problem into a linearly separable one, while decision trees (and forests based on them, and boosted trees, both to a lesser extent due to the nature of the ensemble algorithms) split the input space into hyper-rectangles according to the target.

Model	NameOfParameter	RangeOfParameter	BiasParameter	Dataset
Decision Tree (C4.5)	minNumberObject	2	0.5	1
		4	0	2
		2	0.75	3
		2	0.5	4

Table 4.4 Parameter tuning result of Decision Tree

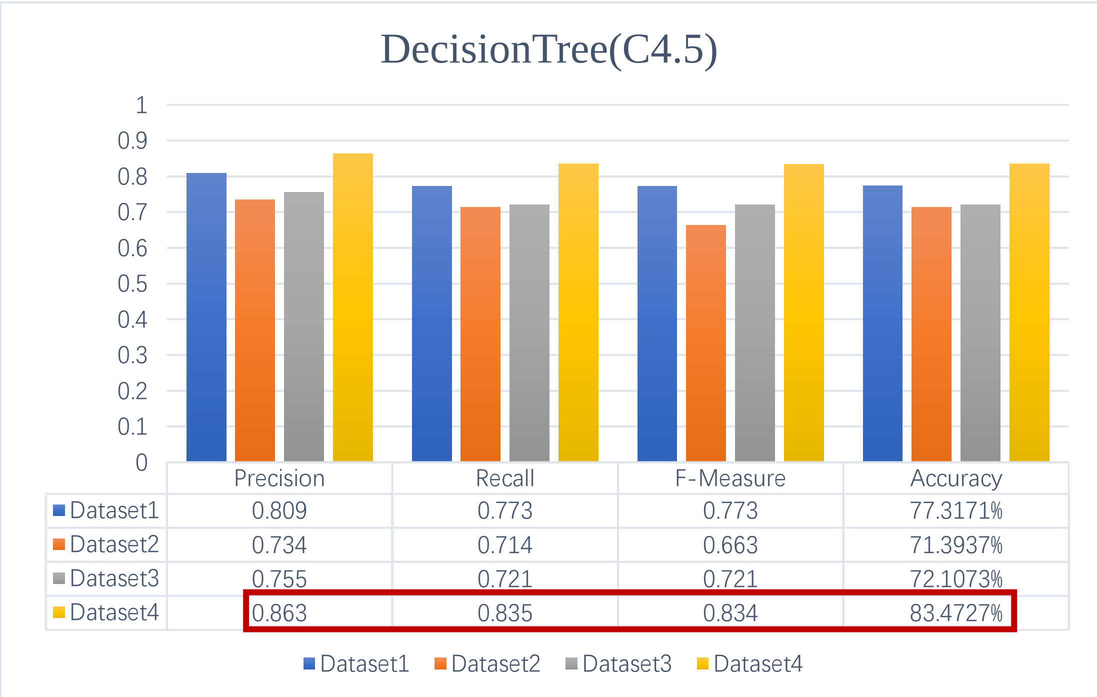


Fig.4.4 Result of Decision Tree

The advantages of Decision Trees are obvious. The first is the implicit performance of feature selection. And then discover non-linear relationships and interactions. In this way, the workload of data preparation and processing in the next decision tree will be reduced, and missing values can be handled reasonably without the influence of outliers. Finally, decision trees can generate rules to help researchers formalize knowledge.

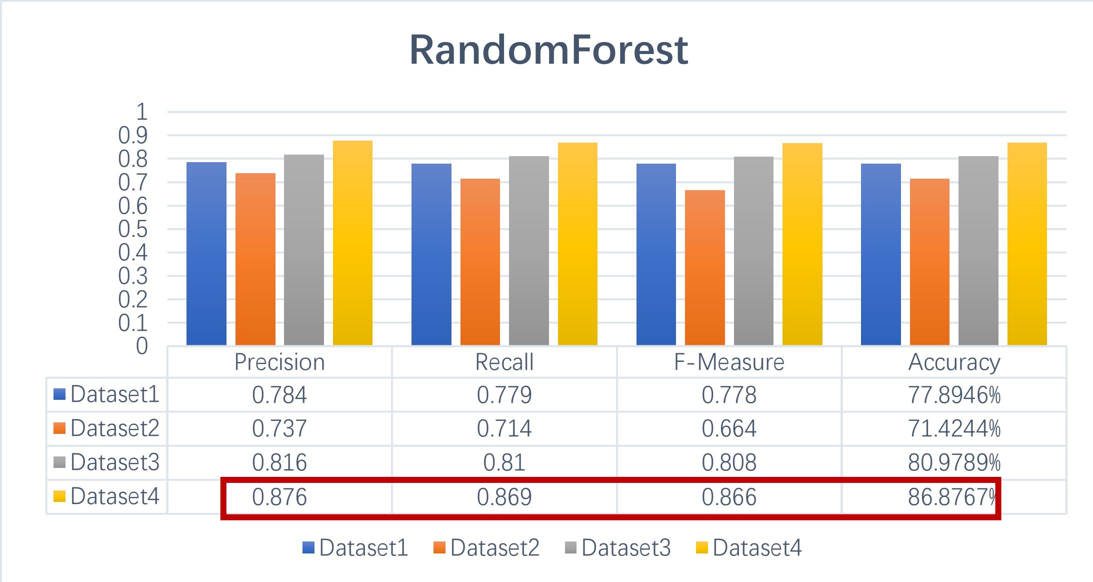


Fig.4.5 Result of Random Forest

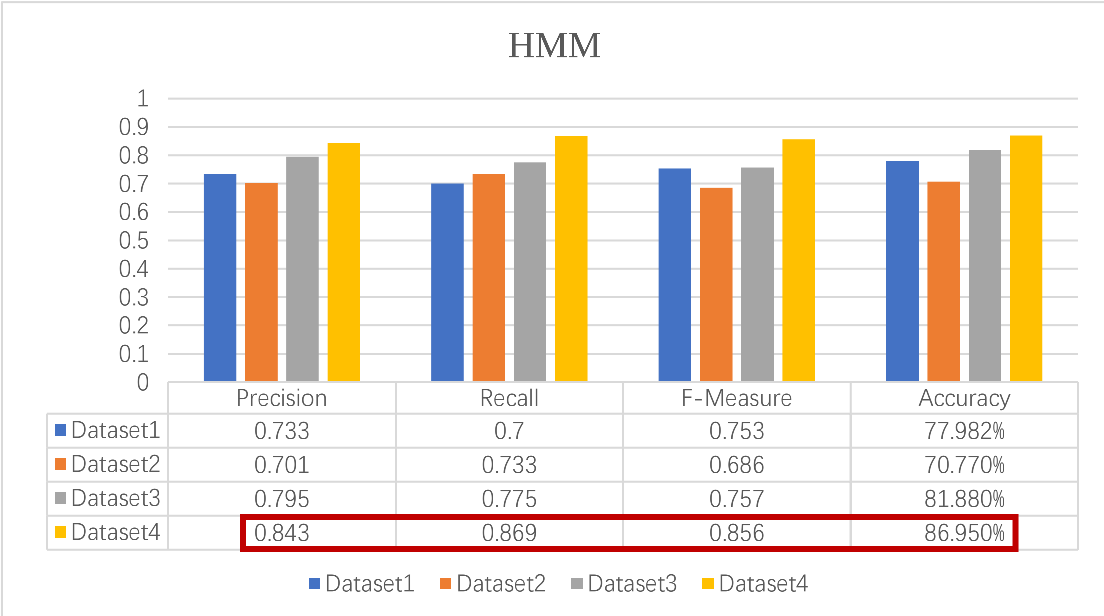


Fig.4.6 Result of HMM

For the RF algorithm, the only parameter to be tuned is the number of trees, which is adjusted by changing the number of trees from 1 to 100 and then determining the value that provides the best accuracy. It is an iterative process when I start at 1 and then 2, 3,4 etc. It was stopped once it was evident that peak performance was found at 20 and 18, 19, 21 and 22 gave worse accuracy than 20. Compared to the Naïve Bayes and Bayes Network, the HMM and Random Forest model gives better results in terms of precision, recall, f-measure and accuracy. The result can be interpreted as HMM model considering the temporal aspects of the data used in this experiment. The HMM model gives the best result followed by SVM with linear kernel, Naïve Bayes and decision tree. It is shown a significant improvement for some models.

When we select and extract different features, we can describe activities very well. Therefore, compared with the original data, the annotated data has a great improvement in classification accuracy. By comparing different classification models, we find that there are many factors affecting the performance of the model. For example, the types of sensors to be used, choosing different experimental schemes, the motivation of researching human activity recognition and the number and category of recognition activities, etc. By using different datasets to conduct experiments in different models, we discovered that the experimental results are influenced by the interaction between residents. Although the datasets contain two residents, it does not mean that they have cooperative activities all the time. Therefore, the interaction between residents is an indispensable factor in designing and adjusting the recognition model of multi-resident activity.

4.4 Boosting and Bagging

Boosting is a powerful method to improve the accuracy of a given base (weak) classifier. The approach consists of two steps. First, the weak classifier is learned by using the original data in an iterative way and a processing method is generated. Then, the weak classifier model is recombined with specific cost functions to obtain a strong classifier to enhance the performance of the model. The process of iteration is the process of re-calculation of the lifting algorithm. Meanwhile, the output results of the previous model should be considered in combination with the hypothesis of each learning, and the data points with wrong classification should be given greater weight. Thus final hypothesis learned can be given as in (Breiman, 1996):

$$\mathrm{F(x)} = \sum_{t=1}^{T} \alpha_t \, h_t(x)$$

Where α_t represent the coefficient with the hypothesis h_t is combined, both αt and h_t are learned during the process of boosting procedure.

At present, some researchers have designed many boosting algorithms. In this experiment we use Adaptive boosting algorithm (abbreviated as AdaBoost). Since the adjusted subsequences used to build classifiers are beneficial to the instances that are misclassified, they are called adaptive.

Bagging, also known as bootstrap aggregating. It is another simple meta-algorithm. Similar to boosting algorithm, multiple classifier models are combined to improve prediction accuracy. This algorithm is mostly used in Decision Tree method to train classifier by redistributing training set randomly. Therefore, the training set of each classifier is generated by randomly selecting **n** instances for replacement. **N** represents the size of the original training set. The final bagged estimator $h_{bag}(.)$ is the predicted expectation for each training hypothesis. If $\boldsymbol{h_k}(.)$ is the hypothesis learned for training sample k, then it has:

$$h_{bag}(.) = \frac{1}{M} \sum_{k=1}^{M} h_k(.) \quad \text{(Bühlmann \& Yu, 2000)}$$

The following figures show the result that each model after using bagging and boosting. Fig.4.6 and Fig.4.7 show the dataset1 bagging and boosting results. From the results we can see that, after using bagging, the accuracy has not improved greatly.

From the results we can see that simple boosting approaches almost always produces better performance than just training a single classifier. For some of datasets the gains in performance are quite significant. Such as dataset1 with Naïve Bayes, the result before boosting was 0.705, after boosting was 0.779. For dataset2 with Decision Tree, the result before boosting was 0.714, after boosting was 0.74. From dataset1 result, we can see after using boosting, the accuracy has improved greatly. But sometimes it produces worse result. However, most times it not only significantly outperforms using a single classifier, but significantly outperforms bagging. Boosting results are even more extreme. For certain datasets boosting produces a significant gain over any other method like dataset1. On other datasets like dataset 3, Boosting results that are even worse than using a single classifier. For both Boosting methods, it seems that when they work, they are extremely effective; on the other hand, when the Boosting methods fail they can often hinder performance. It is also interesting to note that both Boosting methods significantly outperform Bagging on the letter, segmentation, and vehicle domains, which suggests that their positive effects may be greatest when multiple classes are predicted.

Bagging and Boosting reduce the variance of the single estimate because they combine estimates from different models. The result show that the model has higher stability. If the performance of a single model is low, then bagging will rarely yield a better bias. However, it optimizes the advantages of a single model and reduces defects, a composite model with a small error can be generated. In contrast, if a single model is too difficult, bagging is the best option. Enhancement does not prevent overfitting. In fact, the technology itself faces this problem. So, to some extent, bagging is more effective than boosting.

			Precision	Recall	F-measure	Accuracy
Dataset1	BN	Before	0.803	0.788	0.787	0.789
		After	**0.732**	**0.713**	**0.703**	**0.713**
	NB	Before	0.734	0.705	0.698	0.705
		After	**0.648**	**0.605**	**0.585**	**0.605**
	SVM	Before	0.891	0.886	0.885	0.886
		After	**0.803**	**0.785**	**0.783**	**0.785**
	DT	Before	0.768	0.73	0.719	0.73
		After	**0.7**	**0.62**	**0.643**	**0.703**
	RF	Before	0.784	0.779	0.778	0.778
		After	**0.7**	**0.75**	**0.75**	**0.75**
	HMM	Before	0.733	0.762	0.753	0.78
		After	**0.6**	**0.664**	**0.6**	**0.7**
	BN	Before	0.727	0.712	0.662	0.712
		After	**0.71**	**0.72**	**0.6**	**0.7**

Dataset2	NB	Before	0.722	0.704	0.644	0.704
		After	**0.71**	**0.72**	**0.62**	**0.7**
	SVM	Before	0.722	0.712	0.66	0.712
		After	**0.71**	**0.70**	**0.64**	**0.70**
	DT	Before	0.733	0.714	0.663	0.714
		After	**0.71**	**0.70**	**0.62**	**0.70**
	RF	Before	0.727	0.712	0.662	0.712
		After	**0.71**	**0.70**	**0.6**	**0.70**
	HMM	Before	0.701	0.733	0.686	0.701
		After	**0.69**	**0.71**	**0.6**	**0.68**
Dataset3	BN	Before	0.806	0.779	0.777	0.779
		After	**0.792**	**0.771**	**0.766**	**0.78**
	NB	Before	0.76	0.737	0.73	0.737
		After	**0.75**	**0.728**	**0.72**	**0.728**
	SVM	Before	0.856	0.838	0.84	0.838
		After	**0.864**	**0.848**	**0.849**	**0.85**
	DT	Before	0.755	0.721	0.721	0.721
		After	**0.741**	**0.701**	**0.702**	**0.705**
	RF	Before	0.816	0.81	0.808	0.811
		After	**0.81**	**0.81**	**0.79**	**0.8**
	HMM	Before	0.795	0.775	0.757	0.819
		After	**0.78**	**0.77**	**0.754**	**0.785**
Dataset4	BN	Before	0.845	0.84	0.84	0.84
		After	0.832	0.83	0.83	0.83
	NB	Before	0.795	0.797	0.795	0.797
		After	**0.788**	**0.785**	**0.788**	**0.788**
	SVM	Before	0.884	0.884	0.884	0.884
		After	**0.88**	**0.88**	**0.88**	**0.88**
	DT	Before	0.863	0.835	0.834	0.834
		After	**0.84**	**0.82**	**0.82**	**0.82**
	RF	Before	0.876	0.869	0.866	0.869
		After	**0.77**	**0.75**	**0.73**	**0.79**
	HMM	Before	0.843	0.869	0.856	0.87
		After	**0.82**	**0.84**	**0.83**	**0.85**

Table 4.5 Summary of bagging results

			Precision	Recall	F-measure	Accuracy
Dataset1	BN	Before	0.803	0.788	0.787	0.789
		After	**0.81**	**0.778**	**0.785**	**0.778**
	NB	Before	0.734	0.705	0.698	0.705
		After	**0.782**	**0.779**	**0.779**	**0.779**
	SVM	Before	0.891	0.886	0.885	0.886
		After	**0.892**	**0.885**	**0.885**	**0.886**
	DT	Before	0.768	0.73	0.719	0.73
		After	**0.7**	**0.62**	**0.643**	**0.703**
	RF	Before	0.784	0.779	0.778	0.778
		After	**0.79**	**0.78**	**0.78**	**0.78**
	HMM	Before	0.733	0.762	0.753	0.78
		After	**0.75**	**0.76**	**0.75**	**0.78**
Dataset2	BN	Before	0.727	0.712	0.662	0.712
		After	**0.72**	**0.71**	**0.65**	**0.71**
	NB	Before	0.722	0.704	0.644	0.704
		After	**0.73**	**0.71**	**0.64**	**0.71**
	SVM	Before	0.722	0.712	0.66	0.712
		After	**0.74**	**0.73**	**0.68**	**0.73**
	DT	Before	0.733	0.714	0.663	0.714
		After	**0.75**	**0.72**	**0.67**	**0.74**
	RF	Before	0.727	0.712	0.662	0.712
		After	**0.73**	**0.72**	**0.65**	**0.72**
	HMM	Before	0.701	0.733	0.686	0.701
		After	**0.73**	**0.75**	**0.7**	**0.73**
Dataset3	BN	Before	0.806	0.779	0.777	0.779
		After	**0.805**	**0.77**	**0.77**	**0.76**
	NB	Before	0.76	0.737	0.73	0.737
		After	**0.73**	**0.72**	**0.70**	**0.72**
	SVM	Before	0.856	0.838	0.84	0.838
		After	**0.879**	**0.868**	**0.869**	**0.867**
	DT	Before	0.755	0.721	0.721	0.721
		After	**0.815**	**0.785**	**0.787**	**0.785**
	RF	Before	0.816	0.81	0.808	0.811
		After	**0.83**	**0.83**	**0.81**	**0.83**
	HMM	Before	0.795	0.775	0.757	0.819
		After	**0.85**	**0.83**	**0.81**	**0.83**
Dataset4	BN	Before	0.845	0.84	0.84	0.84
		After	**0.856**	**0.85**	**0.85**	**0.85**
	NB	Before	0.795	0.797	0.795	0.797
		After	**0.85**	**0.85**	**0.84**	**0.85**
	SVM	Before	0.884	0.884	0.884	0.884
		After	**0.895**	**0.895**	**0.895**	**0.895**
	DT	Before	0.863	0.835	0.834	0.834

		After	**0.88**	**0.85**	**0.85**	**0.87**
	RF	Before	0.876	0.869	0.866	0.869
		After	**0.89**	**0.87**	**0.87**	**0.87**
	HMM	Before	0.843	0.869	0.856	0.87
		After	**0.87**	**0.89**	**0.88**	**0.88**

Table 4.6 Summary of Boosting result

4.5 Discussion

In order to evaluate the performance of the BN, NB, SVM, DT, HMM and RF classifiers in human activity recognition, we conducted 10-fold cross-validation on four datasets. Metrics, such as recall, precision and the F-measure were used to evaluate the performance of the classification.

Dataset	Model	F-Measure (%)	Accuracy (%)
Dataset1	BN	78.77	78.84
	NB	69.85	70.54
	SVM (Linear)	**88.52**	**88.57**
	SVM (radial basis)	50.44	51.34
	SVM (polynomial)	50.43	51.40
	SVM (sigmoid)	50.21	51.2
	DT	77.34	77.31
	RF	55.61	60.24
	HMM	75.33	77.98
Dataset2	BN	66.23	71.19
	NB	64.44	70.38
	SVM (Linear)	66.61	71.36
	SVM (radial basis)	65.8	71.1
	SVM (polynomial)	46.4	55.61
	SVM (sigmoid)	65.4	70.1
	DT	66.33	71.16
	RF	**66.41**	**71.42**
	HMM	68.61	70.77
Dataset3	BN	77.77	77.92
	NB	73	73.75
	SVM (Linear)	74.82	74.57
	SVM (radial basis)	50.44	51.29
	SVM (polynomial)	50.53	51.43
	SVM (sigmoid)	50.45	51.53
	DT	72.11	72.11
	RF	80.8	80.98
	HMM	**75.77**	**81.88**

Dataset4	BN	84.1	83.98
	NB	79.51	79.72
	SVM (Linear)	**88.41**	**88.44**
	SVM (radial basis)	60.34	61.23
	SVM (polynomial)	60.45	61.45
	SVM (sigmoid)	60.24	61.32
	DT	83.41	83.47
	RF	86.61	86.88
	HMM	85.67	86.95

Table 4.7 Classification result for each dataset

All the experiments results are shown in Table 4.7. It is given for each of the proposed models. We cycled through all training sequences using 10-fold cross validation and reported the mean value of F-measure and accuracy. In fact, considering all the metrics, linear SVM performs better result than other models, especially in dataset4, its accuracy is as high as 88%. By comparing the variance of the accuracy of each model, we can know the variance value of the Random Forest is the largest, indicating that its data fluctuation is also the largest. The performance of the Random Forest in these four datasets is also the most unstable. On the contrary, Naïve Bayes performs better than other models and performs more stably when classifying four data sets which is expected. Moreover, when looking at the accuracies separated for datasets, it is clear that dataset4 has a good performance compared with other datasets.

By using the data collected by environmental sensors, the linear SVM classification algorithm was used to realise the accurate classification of ADL activities and the accuracy rate is 88.57%. Compared with other general classification models, this system is more effective in linear kernel classifier. The result can be applied to actual deployment, as there is no need to grasp the real-time ADL instances in the training process. In dataset activities, we still need to consider the factors of overlapping. For example, the resident is cooking while the other resident is bathing. This leads to a low sensitivity of the model. In fact, for these overlapping activities, the same type of sensor is triggered, and the model is processed according to the type of sensor triggered without considering the time characteristics. Naïve Bayes is greatly affected by this factor. In the process of classifying, the model classifies each slot separately without considering the duration of the activity. The HMM model will highlight its advantages, especially when dealing with Dataset3 with an accuracy rate of 88.44% accuracy.

As one of the simplest Bayesian classification methods, Naïve Bayes has stable classification efficiency. Generally speaking, it is suitable for multi-classification tasks with incremental learning methods. One prominent feature is that it is the sensitive to some redundant or uncorrelated features, so the model performs better when the correlations between properties are small. Han et al. (Yuan, Wang, Meng, Yan, & Xia, 2019) rationally applied the attributes of Naïve Bayes model with the same priority characteristics and proposed an adaptive multi-layer activity recognition framework, which achieved the real-time recognition of 15 kinds of activities. Therefore, on the other hand, Naïve Bayes model is still a popular classification algorithm due to its simple and fast modeling process.

Support Vector Machine (SVM) which has the worst classification accuracy among the three algorithms, and the fastest training speed, has three advantages. Firstly, SVM effectively maps non-linear data sets to high-dimensional vectors by using inner product and kernel function, so that it can be separable in high-dimensional linearity. Secondly, for geometric intervals, SVM simplifies the computational complexity by selecting support vectors, thus avoiding dimension disaster. Thirdly, when choosing the kernel function, RBF is regarded as a kind of neural network, which can better deal with the errors. On the other hand, the disadvantage of Support Vector Machine (SVM) is that when faced with large data sets, it has to map all data sets to high dimensions, which will take up a lot of memory and computing time to hide Markov model. The results of running speed and classification accuracy are between the two. The model based on Markov chain has two advantages. First, the Hidden Markov Model regards data sets as continuous actions and converts them into a sequence of special observation states, which makes these actions look more significant. No matter how large the quantity dimension of input is, the Hidden Markov Model has the ability to generate a sequence after simple pretreatment. This advantage is not found in the other two methods. The disadvantage of this model is that it may not converge to a real optimal parameter and thus lead to over-fitting. However, this is a problem common to many machine learning algorithms. In addition, one of the characteristics of this model is that the future nodes only depend on the present and have nothing to do with the past. Because human daily activities are a continuous movement, some mistakes may be avoided when we learn the past and present states simultaneously.

4.6 Statistical Analysis

In these experiments, we applied different data classification methods to different datasets, and assessed how varying the classification altered the performance of motion recognition. All approaches increased the performance of multi-resident AR, but the degree of improvement depended on the dataset.

Model	N	Mean Square	df	Sig.
Bayes Network	4	0.044	3	0.345
Naïve Bayes	4	0.003	3	0.440
SVM (linear)	4	0.344	2	0.011
DecisionTree	4	0.554	4	0.401
RandomForest	4	0.452	3	0.553
HMM	4	0.003	3	0.023

Table 4.8 Result of statistical test

To verify that the dataset also affects the classification results, we compared the results of the six classification methods by the ANOVA parametric test. The null hypothesis is that there are no differences between each method. In our experiment, if the calculated probability was lower than 0.05, the null hypothesis was rejected, meaning that at least two of the variables were significantly different. Conversely, the null hypothesis was accepted if the probability exceeded 0.05. The results are shown in Table 4.8. DF means "the degrees of freedom in the source." Sig. means "the P-value." The Sig values of the six methods in Table 4.8 were 0.345, 0.44, 0.011, 0.401, 0.553 and 0.023. As the values of the SVM (linear) and HMM were below 0.05, the null hypothesis was rejected, implying that the degrees of affected in these classification methods varied among the datasets. Meanwhile, the Sig value of other four methods was greater than 0.05, so we accepted the null hypothesis, namely the classification results of approaches do not depend on the datasets.

4.7 Summary

This chapter mainly discusses the experimental results and analysis. The main goal is to evaluate different activity recognition models by different metrics methods. Finally, the most optimal classifier is found in six classification models. In the experiment, we used 10-fold cross validation. The results show that linear SVM and HMM are more accurate than other models. When we analyses the experimental results from the perspective of statistical, we find that the statistical values of linear SVM and HMM are less than 0.05, which means that these classification methods have different degrees of impact on the datasets.

Chapter 5 Conclusion and Future Work

In this thesis, six different general classification machine learning models are evaluated by studying and analysing the classification problem of multi-resident activity recognition based on using sensors. In this chapter, we first summarise the contributions of this thesis and also answer the research questions in detail. Then, based on the current research, future work is proposed.

5.1 Summary of Contribution

Activity recognition has made great advances in recent years. Actually, in the last decade, much more attention has been paid to activity modelling based on sensor data and activity recognition. Because of the emergence of ubiquitous sensing technology, activity-aware applications have been introduced to the market, and anyone can buy them through the Internet. The combination of different types of sensors in machine learning can map our physical world into a digital world. In Chapter 1 we mainly discussed the motivation and the research question of this thesis.

In the related work of Chapter 2, we summarise around 70 research papers on activity recognition. Meanwhile, in order to establish the theoretical basis for the next stage of research, the effects of different classification models in this field are summarised. Among them, we have reviewed about 20 papers on activity recognition, including many overviews. Through sorting and analysis, we can not only understand the conceptual framework, research methods, development stages and trends of activity recognition at the present stage, but also grasp the challenges and limitations encountered in the development process. In this process, we analysed the existing experimental results and then applied them to our research. Next, we focused on research about different classification models in multi-resident activity recognition. Nearly 40 articles were relevant. In the process, we found that most of the studies adopted more than one classification model. The literature review provides arguments for the research and also explores new research methods and approaches used in our subsequent research. In real life, the number of residents living in the environment increases, as does the complexity of daily life. Multi-resident activities are often cooperative.

In Chapter 3, we discussed the processes of activity recognition. We focused on the data acquisition based on environmental sensors, then preprocessing the data, in order to improve the training model accuracy, we remove some noise data, and add a new label for sensor values and activity. Next segmenting all the data and extracting important features and, finally, classifying. Explained the six experimental methodology in detail. We discussed the state-of-art computational models used for classifying activities. In order to find the optimal model, we chose different evaluation criteria to measure it.

In Chapter 4, we tested the performance of the Bayes Network, Naïve Bayes, SVM (different kernel), Decision Tree, Random Forest and HMM classification methods on public datasets. The internal parameters of these methods have been optimised using the training data. Afterwards, the performance of the trained methods has been assessed using that test data. The results show that the linear SVM has high accuracy in each dataset, which indicates that the classification situation is ideal. After adjusting the different parameters, the final accuracy rate achieved is 88.57%. Secondly, the performance of HMM in each model is only inferior to that of linear SVM. Some activities' characteristics have a significant impact on the activity recognition performance. HMMs are robust to the noise in sensor readings and to the uncertainty while performing activities. Moreover, the HMM is capable of considering the sequential nature of activities.

The overall contribution of this thesis to the research is to explore the classification status of the general machine learning models in multi-resident activity recognition. We have integrated 6 general models in the application of this area. By comparing different models on different datasets, we conclude that the selection of the model will influence the classifying of results. Through statistical analysis, we deduce that the classification of linear SVM and HMM has significant differences in different datasets.

5.2 Limitation and Future Work

Throughout the whole research, the first limitation is about data. Due to the novelty of the field of multi-resident activity recognition, the data used in our experiments is binary sensor data and represent as a sequence of event. So the sensor state is changeable. From the current research, we know that the experimental datasets need a lot of human and financial resources to collect, most of researches prefer to use public datasets specially with more than two people did different types of activities datasets. So we didn’t train as many different dataset as possible to extend the accuracy of models. It is still difficult to compare the multi-resident activity recognition models quantitatively.

Another limitation is about the improvement and perfection for selection of classification algorithms. In the data mining area, selecting the correct data representation or feature is more important than selecting the correct technology. So we need spend more time to optimise the model structure. Investigate the best feature and classification techniques combination to solve the problem.

In the field of multi-resident activity recognition, there are many open issues with respect to sensory data pre-processing steps such as discretisation and feature representation. We plan to experiment with different lengths of time interval used for discretisation (e.g. 10 seconds, 20 seconds,...60 seconds). This would allow us to suggest a feature representation or to experiment with existing ones such as changepoint and last-fired representation. Transforming sensor data to a different feature representation sometimes does improve the recognition performance of a model significantly.

The last aspect we will investigate is real-word multi-resident activities in a more complex scenario because the available data hitherto used is rather scripted and does not reflect the real-world setting. Specifically, we plan to evaluate the proposed models on another set of data collected in real world situations. The dataset we used in our experiment is labeled as the data association variables are given. Thus, the performance of an activity recogniser is independent to that of a data associator. This allows us to compare the performance of activity recognition models in an objective manner. We can develop an approach which is based on contextual information (i.e. activity performed by each resident, properties of the activity performed at time, who trigged active sensor, locations of active sensors in the living space, etc.). Then we could apply our proposed research model. However, the activity recognition performance of each model would be immediately affected by the performance of the data association recogniser (i.e. when the sensor data is incorrectly associated with the resident).

In this study, we mainly focus on the application of general classification models of machine learning in multi-resident activity recognition. In this process, we also found that deep learning is also involved in the field of multi-resident activity recognition. But up to now, there are few researches on the ARAS and CASAS datasets. With the continuous development of deep learning, we can use different deep learning models for comparison in the future work.

References

Al-Nawashi, M., Al-Hazaimeh, O. M., & Saraee, M. (2017). A novel framework for intelligent surveillance system based on abnormal human activity detection in academic environments. *Neural Computing and Applications, 28*(1), 565-572.

Alaa, A., Vaidehi, M., Doreen, B., hnstedt, & Ralf, S. (2016). Activity Recognition in Multi-User Environments Using Techniques of Multi-label Classification *Proceedings of the 6th International Conference on the Internet of Things %@ 978-1-4503-4814-0* (pp. 15-23). Stuttgart, Germany: ACM.

Alam, M. A. U., Roy, N., Misra, A., & Taylor, J. (2016). CACE: Exploiting Behavioral Interactions for Improved Activity Recognition in Multi-Inhabitant Smart Homes. *Proceedings 2016 Ieee 36th International Conference on Distributed Computing Systems Icdcs 2016*, 539-548. https://doi.org/10.1109/Icdcs.2016.61

Alemdar, H., & Ersoy, C. (2017). Multi-resident activity tracking and recognition in smart environments. *Journal Of Ambient Intelligence And Humanized Computing, 8*(4), 513-529. https://doi.org/10.1007/s12652-016-0440-x

Álvarez de la Concepción, M. Á., Soria Morillo, L. M., Álvarez García, J. A., & González-Abril, L. (2017). Mobile activity recognition and fall detection system for elderly people using Ameva algorithm. *Pervasive and Mobile Computing, 34*, 3-13. https://doi.org/https://doi.org/10.1016/j.pmcj.2016.05.002

Amiribesheli, M., Benmansour, A., & Bouchachia, A. (2015). A review of smart homes in healthcare. *Journal Of Ambient Intelligence And Humanized Computing, 6*(4), 495-517. https://doi.org/10.1007/s12652-015-0270-2

Anthony Fleury, M. V., Norbert Noury. (2010). SVM-based multimodal classification of activities of daily living in health smart homes: sensors, algorithms, and first experimental results. *IEEE Transactions on Information Technology in Biomedicine - Special section on affective and pervasive computing for healthcare 14*(2), 251.

Anyanwu, M. N., & Shiva, S. G. (2009). Comparative analysis of serial decision tree classification algorithms. *International Journal of Computer Science and Security, 3*(3), 230-240.

ASMA BENMANSOUR, A. B., MOHAMMED FEHAM. (2016). Multioccupant Activity Recognition in Pervasive Smart Home Environments. *ACM Computing Surveys (CSUR) 48*(3), 0360-0300.

Barnan Das, N. C. K., Diane J. Cook. (2013). *Handling Class Overlap and Imbalance to Detect Prompt Situations in Smart Homes*. presented at the meeting of the 2013 IEEE 13th International Conference on Data Mining Workshops, Dallas, TX, USA.

Benmansour, A., Bouchachia, A., & Feham, M. (2015). Multioccupant Activity Recognition in Pervasive Smart Home Environments. *Acm Computing Surveys, 48*(3). https://doi.org/Artn 34

10.1145/2835372

Benmansour, A., Bouchachia, A., & Feham, M. (2017). Modeling interaction in multi-resident activities. *Neurocomputing, 230*(Supplement C), 133-142. https://doi.org/https://doi.org/10.1016/j.neucom.2016.05.110

Bouchachia, M. A. A. B. A. (2015). A review of smart homes in healthcare. *J Ambient Intell Human Comput*, 22.

Breiman, L. (1996). Bagging predictors. *Machine learning, 24*(2), 123-140.

Brownlee, J. (2011). *Clever algorithms: nature-inspired programming recipes*: Jason Brownlee.

Bühlmann, P. L., & Yu, B. (2000). Explaining bagging*Seminar für Statistik, Eidgenössische*

Technische Hochschule (ETH). Symposium conducted at the meeting of the Research report/Seminar für Statistik, Eidgenössische Technische Hochschule Zürich

Caldeira, J. M., Rodrigues, J. J., & Lorenz, P. (2012). Toward ubiquitous mobility solutions for body sensor networks on healthcare. *IEEE Communications Magazine, 50*(5), 108-115.

Carolin Strobl, J. M., Gerhard Tutz. (2009). An Introduction to Recursive Partitioning: Rationale, Application and Characteristics of Classification and Regression Trees, Bagging and Random Forests. *Psychol Methods.*, 323-348.

Chen, L., Hoey, J., Nugent, C. D., Cook, D. J., & Yu, Z. (2012). Sensor-based activity recognition. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), 42*(6), 790-808.

Chen, R., & Tong, Y. (2014). A two-stage method for solving multi-resident activity recognition in smart environments. *Entropy, 16*(4), 2184-2203.

Chiu, M.-H., Yu, Y.-R., Liaw, H. L., & Chun-Hao, L. (2016). The use of facial micro-expression state and Tree-Forest Model for predicting conceptual-conflict based conceptual change. *Chapter Title & Authors Page*, 184.

Cook, D. J. (2010). Learning setting-generalized activity models for smart spaces. *IEEE intelligent systems, 2010*(99), 1.

Cook, D. J., & Krishnan, N. C. (2015). *Activity learning: discovering, recognizing, and predicting human behavior from sensor data*: John Wiley & Sons.

Cook, N. R. A. M. D. (2016). Ambient and smartphone sensor assisted ADL recognition in multi-inhabitant smart environments. *J Ambient Intell Human Comput 7*, 19.

Cooper, G. F., & Herskovits, E. (1992). A Bayesian method for the induction of probabilistic networks from data. *Machine learning, 9*(4), 309-347.

Crandall, A. S., & Cook, D. (2008). Attributing events to individuals in multi-inhabitant environments.

Damla Arifoglu, A. B. (2017). Activity Recognition and Abnormal Behaviour Detection with Recurrent Neural Networks. *Procedia Computer Science 110*, 7.

Daniel Wilson, C. A. (2005). Simultaneous Tracking & Activity Recognition (STAR) Using Many Anonymous, Binary Sensors. *Pervasive Computing, 3468*, 62-79.

Daniele, R., Timo, S., Gabriele, C., & Heiner, S. (2016). Unsupervised recognition of interleaved activities of daily living through ontological and probabilistic reasoning *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing %@ 978-1-4503-4461-6* (pp. 1-12). Heidelberg, Germany: ACM.

Daniele Riboni, L. P., Laura Radaelli, Claudio Bettini. (2011). *Is ontology-based activity recognition really effective?* presented at the meeting of the 2011 IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops), Seattle, WA, USA.

Daniele Riboni, T. S., Gabriele Civitarese, Heiner Stuckenschmidt. (2016). Unsupervised recognition of interleaved activities of daily living through ontological and probabilistic reasoning. *ACM International Joint Conference on Pervasive and Ubiquitous Computing*, 12.

de la Concepción, M. Á. Á., Morillo, L. M. S., García, J. A. Á., & González-Abril, L. (2017). Mobile activity recognition and fall detection system for elderly people using Ameva algorithm. *Pervasive and Mobile Computing, 34*, 3-13.

Diane Cook, N. K., Parisa Rashidi. (2013). Activity Discovery and Activity Recognition: A New Partnership. *IEEE Trans Cybern*, 8.

Eddy, S. R. (1996). Hidden Markov models. *Current Opinion in Structural Biology, 6*(3), 361-365. https://doi.org/https://doi.org/10.1016/S0959-440X(96)80056-X

Emi, I. A., & Stankovic, J. A. (2015). SARRIMA: smart ADL recognizer and resident identifier in multi-resident accommodations*ACM.* Symposium conducted at the meeting of the Proceedings of the conference on Wireless Health

Emi, I. A., & Stankovic, J. A. (2015). *SARRIMA: smart ADL recognizer and resident identifier in multi-resident accommodations*. Retrieved from http://doi.acm.org/10.1145/2811780.2811916 https://doi.org/10.1145/2811780.2811916

Emmanuel Munguia Tapia, S. S. I., Kent Larson. (2004). Activity Recognition in the Home Using Simple and Ubiquitous Sensors. *International Conference on Pervasive Computing*, 17.

Eunju Kim, A. H., Diane J. Cook. (2010). Human Activity Recognition and Pattern Discovery. *IEEE Pervasive Computing 2010*, 5.

Fahad, L. G., Tahir, S. F., & Rajarajan, M. (2014). Activity recognition in smart homes using clustering based classification*IEEE.* Symposium conducted at the meeting of the 2014 22nd International Conference on Pattern Recognition

Fahad, L. G., Tahir, S. F., & Rajarajan, M. (2015). Feature selection and data balancing for activity recognition in smart homes*IEEE.* Symposium conducted at the meeting of the 2015 IEEE International Conference on Communications (ICC)

Forkan, A. R. M., Khalil, I., Tari, Z., Foufou, S., & Bouras, A. (2015). A context-aware approach for long-term behavioural change detection and abnormality prediction in ambient assisted living. *Pattern Recognition, 48*(3), 628-641. https://doi.org/10.1016/j.patcog.2014.07.007

Friedman, N., Geiger, D., & Goldszmidt, M. (1997). Bayesian network classifiers. *Machine learning, 29*(2-3), 131-163.

Galván-Tejada, C. E., Galván-Tejada, J. I., Celaya-Padilla, J. M., Delgado-Contreras, J. R., Magallanes-Quintanar, R., Martinez-Fierro, M. L., . . . Gamboa-Rosales, H. (2016). An Analysis of Audio Features to Develop a Human Activity Recognition Model Using Genetic Algorithms, Random Forests, and Neural Networks. *Mobile Information Systems, 2016*. https://doi.org/10.1155/2016/1784101

Garcia-Ceja, E., & Brena, R. F. (2018). An improved three-stage classifier for activity recognition. *International Journal of Pattern Recognition and Artificial Intelligence, 32*(01), 1860003.

Garcia-Ceja, E., Galván-Tejada, C. E., & Brena, R. (2018a). Multi-view stacking for activity recognition with sound and accelerometer data. *Information Fusion, 40*(Supplement C), 45-56. https://doi.org/https://doi.org/10.1016/j.inffus.2017.06.004

Garcia-Ceja, E., Galván-Tejada, C. E., & Brena, R. (2018b). Multi-view stacking for activity recognition with sound and accelerometer data. *Information Fusion, 40*, 45-56. https://doi.org/https://doi.org/10.1016/j.inffus.2017.06.004

García, S., Luengo, J., & Herrera, F. (2015). *Data preprocessing in data mining*: Springer.

Hadi Tabatabaee Malazi, M. D. (2018). Combining emerging patterns with random forest for complex activity recognition in smart homes. *Applied Intelligence, 48*(2), 15.

HaiderJanjua， RimHelaoui, D. C. G. Z. (2016). SmartFABER: Recognizing fine-grained abnormal behaviors for early detection of mild cognitive impairment. *Artificial Intelligence in Medicine, 67*, 17.

Hand, D. J. (2006). Data Mining. *Encyclopedia of Environmetrics, 2*.

Hande Alemdar, H. E., Ozlem Durmaz Incel, Cem Ersoy. (2013). *ARAS human activity datasets in multiple homes with multiple residents*. presented at the meeting of the 2013 7th

International Conference on Pervasive Computing Technologies for Healthcare and Workshops, Venice, Italy.

Hande Alemdar, Cem Ersoy. (2017). Multi-resident activity tracking and recognition in smart environments. *J Ambient Intell Human Comput 8*, 16.

Hsu, K.-C., Chiang, Y.-T., Lin, G.-Y., Lu, C.-H., Hsu, J. Y.-J., & Fu, L.-C. (2010). Strategies for inference mechanism of conditional random fields for multiple-resident activity recognition in a smart home*Springer.* Symposium conducted at the meeting of the International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems

Huang, X., & Dai, M. (2017). Indoor Device-Free Activity Recognition Based on Radio Signal. *IEEE Trans. Vehicular Technology, 66*(6), 5316-5329. https://doi.org/10.1109/TVT.2016.2616883

Jeremie Saives, C. P., Gregory Faraut. (2015). Activity Discovery and Detection of Behavioral Deviations of an Inhabitant From Binary Sensors. *IEEE Transactions on Automation Science and Engineering 12*, 13.

Jing Zhao, X. X., Xin Xu, Shiliang Sun. (2017). Multi-view learning overview: Recent progress and new challenges. *Information Fusion, 38*, 11.

Kashimoto, Y., Fujiwara, M., Fujimoto, M., Suwa, H., Arakawa, Y., & Yasumoto, K. (2017a). ALPAS: Analog-PIR-sensor-based activity recognition system in smarthome*IEEE.* Symposium conducted at the meeting of the 2017 IEEE 31st International Conference on Advanced Information Networking and Applications (AINA)

Kashimoto, Y., Fujiwara, M., Fujimoto, M., Suwa, H., Arakawa, Y., & Yasumoto, K. (2017b). *ALPAS: Analog-PIR-Sensor-Based Activity Recognition System in Smarthome*. Retrieved from https://doi.org/10.1109/AINA.2017.33 https://doi.org/10.1109/AINA.2017.33

Khan, S. S., Karg, M. E., Hoey, J., & Kulic, D. (2012). *Towards the detection of unusual temporal events during activities using HMMs*. Retrieved from http://doi.acm.org/10.1145/2370216.2370444 https://doi.org/10.1145/2370216.2370444

Korhonen, I., Parkka, J., & Van Gils, M. (2003). Health monitoring in the home of the future. *IEEE Engineering in medicine and biology magazine, 22*(3), 66-73.

Kotsiantis, S., Kanellopoulos, D., & Pintelas, P. (2006). Data preprocessing for supervised leaning. *International Journal of Computer Science, 1*(2), 111-117.

Kwon, M.-C., & Choi, S. (2018). Recognition of Daily Human Activity Using an Artificial Neural Network and Smartwatch. *Wireless Communications and Mobile Computing, 2018*.

Kwon, O., Shim, J. M., & Lim, G. (2012). Single activity sensor-based ensemble analysis for health monitoring of solitary elderly people. *Expert Systems with Applications, 39*(5), 5774-5783.

Lara, O. D., & Labrador, M. A. (2013). A Survey on Human Activity Recognition using Wearable Sensors. *IEEE Communications Surveys and Tutorials, 15*(3), 1192-1209. https://doi.org/10.1109/SURV.2012.110112.00192

Lee, Y.-S., & Cho, S.-B. (2011). Activity recognition using hierarchical hidden markov models on a smartphone with 3D accelerometer*Springer.* Symposium conducted at the meeting of the International conference on hybrid artificial intelligence systems

Lior Rokach, O. M. (2005). Decision Trees. *Data Mining and Knowledge Discovery Handbook*, 27.

Liu, Y., Ouyang, D., Liu, Y., & Chen, R. (2017). A novel approach based on time cluster for activity recognition of daily living in smart homes. *Symmetry, 9*(10), 212.

Malazi, H. T., & Davari, M. (2017). Combining emerging patterns with random forest for complex activity recognition in smart homes. *Appl Intell*. https://doi.org/10.1007/s10489-017-0976-2

Markus Prossegger, H. B. (2014). Multi-resident Activity Recognition Using Incremental Decision Trees. *Adaptive and Intelligent Systems: Third International Conference, ICAIS* 9.

Martyn Shuttleworth, L. T. W. (2008). *Significance Test*. Retrieved from https://explorable.com/significance-test

Melo, N., & Lee, J. (2018). Environment aware adl recognition system based on decision tree and activity frame. *Paladyn, Journal of Behavioral Robotics, 9*(1), 155-167.

Mitja Luštrek, B. K. (2008). Fall Detection and Activity Recognition with Machine Learning. *Informatica*, 7.

Nait Aicha, A., Englebienne, G., & Kröse, B. (2013). How lonely is your grandma?: detecting the visits to assisted living elderly from wireless sensor network data*ACM.* Symposium conducted at the meeting of the Proceedings of the 2013 ACM conference on Pervasive and ubiquitous computing adjunct publication

Nef, T., Urwyler, P., Büchler, M., Tarnanas, I., Stucki, R., Cazzoli, D., . . . Mosimann, U. (2015). Evaluation of three state-of-the-art classifiers for recognition of activities of daily living from smart home ambient data. *Sensors, 15*(5), 11725-11740.

Ni, Q., García Hernando, A., & de la Cruz, I. (2015). The elderly's independent living in smart homes: A characterization of activities and sensing infrastructure survey to facilitate services development. *Sensors, 15*(5), 11312-11362.

Nigam, V. (2018). *Statistical Tests — When to use Which?* Retrieved from https://towardsdatascience.com/statistical-tests-when-to-use-which-704557554740

Nishkam Ravi, N. D., Preetham Mysore, Michael L. Littman. (2010). Activity Recognition from Accelerometer Data. *Association for the advancement of artificial intelligence, 5*, 5.

Prossegger, M., & Bouchachia, A. (2014). Multi-resident activity recognition using incremental decision trees*Springer.* Symposium conducted at the meeting of the International Conference on Adaptive and Intelligent Systems

Qin Ni, A. B. G. H., Iván Pau de la Cruz. (2015). The Elderly's Independent Living in Smart Homes: A Characterization of Activities and Sensing Infrastructure Survey to Facilitate Services Development *Sensors*.

Rawat, U. Introduction to Hill Climbing | Artificial Intelligence. *Geeks for Geeks*.

Reisberg, B., Finkel, S., Overall, J., Schmidt-Gollas, N., Kanowski, S., Lehfeld, H., . . . Heininger, K. (2001). The Alzheimer's disease activities of daily living international scale (ADL-IS). *International Psychogeriatrics, 13*(2), 163-181.

Rodomagoulakis, I., Katsamanis, A., Potamianos, G., Giannoulis, P., Tsiami, A., & Maragos, P. (2017). Room-localized spoken command recognition in multi-room, multi-microphone environments. *Computer Speech & Language, 46*, 419-443. https://doi.org/10.1016/j.csl.2017.02.004

Ross Messing, C. P., Henry Kautz (2009). Activity recognition using the velocity histories of tracked keypoints. *IEEE 12th international conference on computer vision*, 7.

Roy, N., Misra, A., & Cook, D. (2016). Ambient and smartphone sensor assisted ADL recognition in multi-inhabitant smart environments. *Journal Of Ambient Intelligence And Humanized Computing, 7*(1), 1-19. https://doi.org/10.1007/s12652-015-0294-7

Ryan, T. H. (1960). Significance tests for multiple comparison of proportions, variances, and other statistics. *Psychological Bulletin, 57*(4), 318-328. https://doi.org/10.1037/h0044320

Safavian, S. R., & Landgrebe, D. (1991). A survey of decision tree classifier methodology. *IEEE transactions on systems, man, and cybernetics, 21*(3), 660-674.

Sarkar, A., Lee, Y.-K., & Lee, S. (2010). ARHMAM: an activity recognition system based on hidden Markov minded activity model*ACM*. Symposium conducted at the meeting of the Proceedings of the 4th International Conference on Uniquitous Information Management and Communication

Sarkar, A. J., & Khan, A. M. (2011). An active process for sensor-based activity data collection*IEEE*. Symposium conducted at the meeting of the 2011 24th Canadian Conference on Electrical and Computer Engineering (CCECE)

Sikder, F., & Sarkar, D. (2017). Log-Sum Distance Measures and Its Application to Human-Activity Monitoring and Recognition Using Data From Motion Sensors. *IEEE Sensors Journal, 17*(14), 4520-4533. https://doi.org/10.1109/JSEN.2017.2707921

Singla, G., Cook, D. J., & Schmitter-Edgecombe, M. (2010). Recognizing independent and joint activities among multiple residents in smart environments. *Journal of ambient intelligence and humanized computing, 1*(1), 57-63.

T. Guettari, J. B., B E. Benkelfat, G. Chollet, J L. Baldinger,P. Dore, D. Istrate. (2014). *Thermal signal analysis in smart home environment for detecting a human presence*. presented at the meeting of the 2014 1st International Conference on Advanced Technologies for Signal and Image Processing (ATSIP), Sousse, Tunisia.

T.L.M. van Kasteren, G. E., B.J.A. Kröse. (2010). Activity recognition using semi-Markov models on real world smart home datasets. *Journal of ambient intelligence and smart environments, 2|3* 41.

Tapia, E. M., Intille, S. S., & Larson, K. (2004). Activity recognition in the home using simple and ubiquitous sensors*Springer*. Symposium conducted at the meeting of the International conference on pervasive computing

Trevor Hastie, R. T., Jerome Friedman. (2009). *The Elements of Statistical Learning*: Springer, New York, NY.

Triboan, D., Chen, L., Chen, F., & Wang, Z. (2017). Semantic segmentation of real-time sensor data stream for complex activity recognition. *Personal and Ubiquitous Computing, 21*(3), 411-425.

trifacta. (2019). *Data Cleansing for Better Analysis & Business Insight*. Retrieved from https://www.trifacta.com/data-cleansing/

U. A. B. U. A. Bakar, H. G., S. F. Hasanm, S. C. Mukhopadhyay. (2015). Activity and Anomaly Detection in Smart Home: A Survey. *Next Generation Sensors and Systems*, 29.

Uddin, M. Z. (2017). Human activity recognition using segmented body part and body joint features with hidden Markov models. *Multimedia Tools and Applications, 76*(11 %@ 1573-7721), 13585-13614. Uddin2017. https://doi.org/10.1007/s11042-016-3742-2 %U https://doi.org/10.1007/s11042-016-3742-2

van Kasteren, T. L., Englebienne, G., & Kröse, B. J. (2011a). Hierarchical activity recognition using automatically clustered actions*Springer*. Symposium conducted at the meeting of the International Joint Conference on Ambient Intelligence

van Kasteren, T. L., Englebienne, G., & Kröse, B. J. (2011b). Human activity recognition from wireless sensor network data: Benchmark and software. In *Activity recognition in pervasive intelligent environments* (pp. 165-186): Springer.

Wang, J., Zhang, X., Gao, Q., Yue, H., & Wang, H. (2017). Device-Free Wireless Localization and Activity Recognition: A Deep Learning Approach. *IEEE Trans. Vehicular Technology, 66*(7), 6258-6267. https://doi.org/10.1109/TVT.2016.2635161

Wilson, D. H., & Atkeson, C. (2005). Simultaneous tracking and activity recognition (STAR) using many anonymous, binary sensors*Springer.* Symposium conducted at the meeting of the International Conference on Pervasive Computing

WSU CASAS Datasets. (2007). Retrieved from http://casas.wsu.edu/datasets/

Xu, L., Yang, W., Cao, Y., & Li, Q. (2017). Human activity recognition based on random forests*IEEE.* Symposium conducted at the meeting of the 2017 13th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD)

Yin, J., Fang, M., Mokhtari, G., & Zhang, Q. (2016a). *Multi-resident Location Tracking in Smart Home through Non-wearable Unobtrusive Sensors*. Retrieved from https://doi.org/10.1007/978-3-319-39601-9_1 https://doi.org/10.1007/978-3-319-39601-9_1

Yin, J., Fang, M., Mokhtari, G., & Zhang, Q. (2016b). Multi-resident location tracking in smart home through non-wearable unobtrusive sensors*Springer.* Symposium conducted at the meeting of the International Conference on Smart Homes and Health Telematics

Yu, H., & Kim, S. (2012). SVM tutorial—classification, regression and ranking. *Handbook of Natural computing*, 479-506.

Yuan, G., Wang, Z., Meng, F., Yan, Q., & Xia, S. (2019). An overview of human activity recognition based on smartphone. *Sensor Review, 39*(2), 288-306.

ZDRAVEVSKI, E., LAMESKI, P., & TRAJKOVIK, V. (2017). Improving Activity Recognition Accuracy in Ambient-Assisted Living Systems by Automated Feature Engineering. *Ieee Access, 5*, 5262-5280. https://doi.org/10.1109/ACCESS.2017.2684913

Zhao, C., Wang, J., & Lu, H. (2017). Learning discriminative context models for concurrent collective activity recognition. *Multimedia Tools and Applications, 76*(5 %@ 1573-7721), 7401-7420. Zhao2017. https://doi.org/10.1007/s11042-016-3393-3 %U https://doi.org/10.1007/s11042-016-3393-3