A SYSTEM FOR CREATING RICH AND INTERACTIVE HUMAN DIGITAL MEMORIES

By

Chelsea Dobbins, BSc (Hons)

A thesis submitted in partial fulfilment of the requirements of Liverpool John Moores University for the degree of Doctor of Philosophy

January 2014

This thesis is dedicated to my grandfather,

Harold Leonard Mentel 1927-2013

Contents

Acknowledgements

Firstly, I would like to thank my excellent supervisory team, Professor Madjid Merabti, Dr Paul Fergus and Dr David Llewellyn-Jones, for their continued support and encouragement throughout my PhD. I would like to thank Professor Madjid Merabti for giving me the opportunity to embark on this endeavour and for always taking the time to provide advice and invaluable knowledge. I would also like to thank Dr Paul Fergus for his constant support and guidance, throughout my PhD. I would like to thank him for always making the time to listen and for sharing his invaluable expertise. I would also like to thank Dr David Llewellyn-Jones for also sharing his knowledge and for providing guidance and invaluable feedback. I have truly been blessed to have such an outstanding supervisory team. Without them, this thesis would not have been possible

I would also like to thank my family and friends for their continued support throughout this process. In particular, I would like to thank my fiancé Henry for his words of reassurance, especially during the times when I doubted myself. I would also like to thank my granddad, Harold, for first encouraging me to have an interest in computing, from the age of three and my mum and dad for always believing in me.

Last but not least, I would like to thank everyone, within the School of Computing & Mathematical Sciences, for their continued support. In particular, I would like to thank William Hurst for being an excellent office mate throughout the years. I would also like to thank Áine MacDermott, Laura Pla Beltran, Tricia Waterson, Lucy Tweedle, Carol Oliver and the technician team for all of their help and support over the years.

Abstract

Memories are an important aspect of a person's life and experience. They shape our existence, influence every aspect of our lives and link our past with our future. Reminiscing, over past experiences, is a substantial part of life. It is a practice that has been performed over thousands of years and is what makes us who we are. Despite this, in our later years, the ability to remember information declines. For example, in the case of remembering a birthday party that happened last week, recalling attendees, location, feelings, temperature (if it was particularly hot or cold), what was eaten and other pieces of information are quite easily remembered. However, trying to remember a birthday party that happened thirty years ago, with the same level of detail, is a lot harder. Nevertheless, recent advances in technology can alleviate this problem, to a certain extent.

Human digital memories are formed as people become more interested in reliving their experiences, through their digital media. The fundamental challenge is to provide a platform that allows us to relive any period of our lives. The more information contained in a digital memory the more vivid the memory. However, this is a challenge as current systems focus on only collecting a specific set of data. Consequently, as more data is collected searching this set of big data is another challenge. An intelligent method of data analysis, which enables the execution of multi-dimensional queries, is essential. Once data has been found structuring this information into a human digital memory is another issue. Data needs to be brought together and presented in a succinct manor. Additionally, human digital memories evolve and grow alongside their human counterparts. Therefore, any human digital memory system needs to be able to withstand the evolution of technology so that a lifetime of data can be utilized in creating such memories.

This thesis presents the DigMem system, which has been developed to address these challenges. DigMem utilizes distributed mobile services, linked data and machine learning to create rich and interactive human digital memories. In this way, information is structured to create temporal memory boxes of human experiences. The system is also able to answer life questions about our human digital memory data. A prototype has been successfully developed, which demonstrates the approach. Furthermore, the thesis evaluates the use of machine learning algorithms, as an alternative to keyword searching methods. Supervised machine learning algorithms have been used to answer life questions about the user's data, such as "Have I been running?" Alternatively, an unsupervised approach has also been used to cluster data based on a different set of key questions, such as "When have I been the most active?" and "When have I been the least active?" The results, from both the supervised and unsupervised learning approaches, have been successful.

Publications Resulting From This Thesis

Journal Papers

1. **Chelsea Dobbins**, Madjid Merabti, Paul Fergus, and David Llewellyn-Jones, "Creating Human Digital Memories with the Aid of Pervasive Mobile Devices" in *Pervasive and Mobile Computing*, 2013 (In Press)

2. **Chelsea Dobbins**, Madjid Merabti, Paul Fergus, David Llewellyn-Jones and Faycal Bouhafs, "Exploiting Linked Data to Create Rich Human Digital Memories" in *Computer Communications*, vol. 36, no. 15-16, pp. 1639-1656, 2013

3. Paul Fergus, Pauline Cheung, Abir Hussain, Dhiya Al-Jumeily, **Chelsea Dobbins**, and Shamaila Iram, "Prediction of Preterm Deliveries from EHG Signals using Machine Learning" in *PLoS ONE*, vol. 8, no. 10, pp. e77154, 2013

4. **Chelsea Dobbins**, Paul Fergus, Gareth Stratton, Michael Rosenberg and Madjid Merabti, "Monitoring and Reducing Sedentary Behaviour in the Elderly with the Aid of Human Digital Memories" in *Telemedicine and e-Health*, vol. 19, no. 3, pp. 173-185, 2013

5. Paul Fergus, Andrew Attwood, **Chelsea Dobbins**, Gareth Stratton, Abir Hussain, Dhiya Al-Jumeily, and Martin Randles, "Monitoring and Measuring Physical Activity and Sedentary Behaviour for Supporting Medical Outcome Assessments" in *International Journal of Healthcare Technology and Management (IJHTM)*, vol. 13, no. 5/6, pp. 283-303, 2012

Conference Papers

1. **Chelsea Dobbins**, Madjid Merabti, Paul Fergus, and David Llewellyn-Jones, "The Big Data Obstacle of Lifelogging" in *Proceedings of the 28th IEEE International Conference on Advanced Information Networking and Application Workshops s (WAINA'14), Victoria, Canada*, 2014 (Accepted)

2. **Chelsea Dobbins**, Madjid Merabti, Paul Fergus, and David Llewellyn-Jones, "A User-Centred Approach to Reducing Sedentary Behaviour" in *Proceedings of the* 11^{th} *Annual IEEE Consumer Communications & Networking Conference (CCNC'14), Las Vegas, NV, USA*, 2014 (In Press)

3. **Chelsea Dobbins**, Madjid Merabti, Paul Fergus, and David Llewellyn-Jones, "Creating Human Digital Memories for a Richer Recall of Life Experiences" in *Proceedings of the* 10^{th} *IEEE International Conference on Networking, Sensing and Control (ICNSC'13), Evry, France*, 2013, pp. 246-251

4. Asaad Sabah Hadi, Paul Fergus, **Chelsea Dobbins**, Abbas Muhsin Al-Bakry, "A Machine Learning Algorithm for Searching Vectorised RDF Data" in *Proceedings of the* 27^{th} *IEEE International Conference on Advanced Information Networking and Applications Workshops (WAINA'13), Barcelona, Spain*, 2013, pp. 613-618

5. **Chelsea Dobbins**, Madjid Merabti, Paul Fergus, and David Llewellyn-Jones, "Capturing and Sharing Human Digital Memories with the Aid of Ubiquitous Peer-to-Peer Mobile Services," in *Proceedings of the* 10^{th} *Annual IEEE International Conference on Pervasive Computing and Communications (PerCom'12), Lugano, Switzerland,* 2012, pp. 64-69

6. **Chelsea Dobbins**, Paul Fergus, Madjid Merabti, and David Llewellyn-Jones, "Monitoring and Measuring Sedentary Behaviour with the Aid of Human Digital Memories," in *Proceedings of the* 9^{th} *Annual IEEE Consumer Communications & Networking Conference (CCNC'12), Las Vegas, NV, USA,* 2012, pp. 395-398

7. **Chelsea Dobbins**, Paul Fergus, Madjid Merabti, and David Llewellyn-Jones, "Remotely Monitoring and Preventing the Development of Pressure Ulcers with the Aid of Human Digital Memories," in *Proceedings of the* 47^{th} *Annual IEEE International Conference on Communications (ICC'12), Ottawa, Canada*, 2012, pp. 6101-6105

8. **Chelsea Dobbins**, Madjid Merabti, Paul Fergus, and David Llewellyn-Jones, "Augmenting Human Digital Memories with Physiological Data" in *Proceedings of the 3rd IEEE International Conference on Network Embedded Systems for Every Application (NESEA'12)*, *Liverpool, UK*, 2012, pp. 35-41

9. **Chelsea Dobbins**, Madjid Merabti, Paul Fergus, and David Llewellyn-Jones, "Towards a Framework for Capturing and Distributing Rich Interactive Human Digital Memories," in *Proceedings of the 12th Annual PostGraduate Symposium on the Convergence of Telecommunications, Networking and Broadcasting (PGNet'11), Liverpool, UK*, 2011

Book Chapter

1. **Chelsea Dobbins**, Madjid Merabti, Paul Fergus, and David Llewellyn-Jones, "Capturing Human Digital Memories for Assisting Memory Recall" in *Advances in Physiological Computing*, Springer, 2014 (In Press)

List of Figures

List of Tables

Chapter 1

Introduction

The turn of the 21st century has seen a fundamental advancement in technology. Computing devices are, nowadays, capable of capturing a multitude of personal data and can store a phenomenal amount of information. We are now living in a data rich society, where the ability to generate and access a number of different data sources is feasible. The combination of the Internet and emerging technologies, such as near-field communications, real-time localization, and embedded sensors, lets us transform everyday objects into "smart objects" that can understand and react to their environment (Kortuem et al. 2010). Any object, embedded with a sensor, is capable of providing us with information. This is known as pervasive computing and can be defined as the, "*Integration of computing power (micro-processors) and sensing (sensors) into anything, including not only traditional computers, personal digital assistants (PDAs), printers, etc., but also everyday objects like white goods, toys, houses, furniture, or even paint ("smart dust")*" (Korhonen & Bardram 2004). Through unique addressing schemes, these pervasive devices are able to interact with each other and cooperate with their neighbours, to reach common goals (Atzori et al. 2010). This revolution is known as the Internet of Things (IoT) and can be defined as "*a worldwide network of uniquely addressable interconnected objects, based on shared communication protocols*" (Mainetti et al. 2011). The proliferation of these devices, within our environment, is becoming more abundant. Currently, according to The NPD Group (NPD Group 2013), there are 425 million devices connected to the Internet in U.S. homes; whilst computers were still the primary connected devices, numerous other devices are close behind, such as smartphones, games consoles, Blu-ray Disc players and Internet connected high-definition televisions (HDTVs). Furthermore, by 2016, Cisco predicts that there will be more than 10 billion mobile Internet-connected devices, which exceeds the worlds projected population, at that time, of 7.3 billion (Cisco 2012). These machines now fit seamlessly into our world, instead of forcing users to enter their environment, a concept first envisioned by Weiser (Weiser 1999).

As more and more devices become part of this global data space, unquestionably, the main strength of the IoT is the high impact it will have on several aspects of everyday-life and behaviour of potential users (Atzori et al. 2010). For example, as observed by Guo *et al.* (Guo et al. 2011), these include:

1. *Mobile Social Networking – Mobile social networking (MSN) aims to improve social connectivity in physical communities, by leveraging information about people, places, and interpersonal interactions.*

2. *Real World Search – A real world search system can identify the real-time location, status and profile information of real world entities.*

3. *Lifelogging – Numerous digital lifelogging systems aim to argument human memory through capturing, storing, and accessing our daily life experiences.*

4. *Enterprise Computing and Groupware – Deploying and using smart things in enterprises can facilitate the communication and collaboration among co-located or non-co-located employees.*

5. *Urban Mobility Systems – Understanding human movement in urban environments has direct implications for the design of future urban public transport systems*.

In addition to smart objects, mobile phones are also becoming a constant fixture in our lives. Such devices are becoming smarter, smaller and more capable of gathering an enormous amount of information, such as temperature, photos, location and videos, to name but a few. As such, 91% of UK adults own a mobile phone, with 81.6 million UK mobile subscriptions being held (Ofcom 2012; Ofcom 2011). As well as using mobile phones to capture and share content, as people develop more of an interest in monitoring their health, body sensors are also becoming more widespread. For example, devices such as FitBit[1], AliveCor[2] and Nike+ Fuelband[3] have been attached to people and connected to smartphones to access real time physiological data, in order to track activity (Fergus, Iram, et al. 2012). These devices are capable of capturing such data as step count,

[1] http://www.fitbit.com/uk
[2] http://www.alivecor.com/
[3] http://www.nike.com/cdp/fuelband/us/en_us/

heart rate, calorie expenditure, distance travelled and even sleeping patterns. Additionally, environmental sensors can gather temperature, humidity and atmospheric readings. As these devices become more prevalent, within our environment, a vast array of information about us and our surroundings can be captured.

Within today's society, it is a common practice to capture, store and share almost every moment of our lives. For example, many people take photos, video family events and use fitness devices to monitor their health and general wellbeing. Sharing such information, using mobile devices, is a popular method to disseminate data. According to a recent report by Cisco (Cisco 2013), in 2012, mobile data traffic grew by 70%, compared to 2011. In 2012, this type of traffic reached 885 petabytes per month and was nearly twelve times greater than the total global Internet traffic in 2000 (75 petabytes per month). Additionally, over 6 billion hours of video are watched each month on YouTube, with 100 hours of video being uploaded every minute (YouTube 2012). Moreover, more than 25% of global views come from mobile devices (more than one billion views a day) (YouTube 2012). As well as sharing experiences, the advent of increasing storage capacities is also enabling a lifetime of data to be feasibly stored. This vast amount of data is growing daily. Consequently, with all of this data readily available, people have become interested in reliving experiences through their collected digital media. Our entire lives can be reconstructed from these digital artefacts, thus creating human digital memories of life experiences. With all of this data at our fingertips, bringing it all together, succinctly, presents new and exciting ways to create digital memories. This interest has led to the task of managing, and using, human digital memories, over a lifetime, being declared a grand challenge in computing (Fitzgibbon & Reiter 2005).

1.1 Human Memories

Time is physically irreversible. The unidirectionality of time is one of nature's most fundamental laws and as long as the universe has existed governs all occurrences; there is no return to yesterday (Tulving 2002). Although it is impossible to physically go back in time, mental time travel occurs every day. As stated by Tulving (Tulving 2002), "*Time's flow is irreversible. The singular exception is provided by the human ability to remember past happenings. When one thinks today about what one did yesterday, time's arrow is bent into a loop. The rememberer has mentally travelled back into her past and thus violated the law of the irreversibility of the flow of time.*" This unique ability resides within all of us and occurs on a daily basis, without hesitation. As such, human memory is considered to be the most basic and important operation of the brain, with very few cognitive processes (recognition, language, planning, etc.) being able to operate effectively without a contribution from it (Tranel & Damasio 2003).

Within our brain, memory encompasses many “systems”. Tulving (Tulving 1993) states that there are five major human memory systems – procedural, perceptual representation, short-term, episodic and semantic memory. In particular, episodic memory is related to memories about personally experienced occasions. The information concerned is mainly about the subject’s experiences of temporally dated episodes or events, and the temporal-spatial relations between them (Tulving 1993; Tulving 1984). When events are remembered, they are usually composed of not only the people that you interacted with and the places you visited, but also the feelings that surrounded those times. These memories are a significant part of our existence that can be shared anywhere and at any time. As such, reminiscing, over past experiences, is a substantial part of life. It is a practice that has been performed over thousands of years and is what makes us who we are.

Memories shape our existence, influence every aspect of our lives and link our past with our future. Remembering the past helps people to re-examine their lives, recalling previous activities and accomplishments, and seek personal validation (Kikhia et al. 2010). As such, retaining information and reconstructing past experiences is one of the most important ways by which a person’s histories animate their current actions and experiences (Sutton 2010). Constructing stories, from our memories, defines family identities and is an integral part of most cultures (McCarthy et al. 2007). Supporting our memory, through cave paintings, storytelling, books and personal diaries, has been done for thousands of years and has been a way to preserve memories over a lifetime (Doherty 2005). However, losing the ability to recollect memories is not only disadvantageous, but can prove quite detrimental, particularly to many older people (McCarthy et al. 2007). As such, retaining every aspect of our lives, for example, how we felt or what we did on a specific day is virtually impossible, especially if we are recollecting events from ten years ago. As people get older, the ability to remember this information declines (Prull et al. 2006). However, recent advances in technology can alleviate this problem, to a certain extent. As previously stated, devices are now capable of capturing an enormous amount of personal information. This has led to people creating extensive digital collections and reflection of those items has become an active part of people’s lives (Kalnikaite & Whittaker 2011).

1.2 The Concept of Human Digital Memories

The idea of storing all of ones accumulated digital items was first proposed in 1945, by Vannevar Bush (Bush 1945), with the concept of the *Memex*. This idea is described as being, “*a device in which an individual stores all his books, records, and communications, and which is mechanized so that it may be consulted with exceeding speed and flexibility. It is an enlarged intimate supplement to memory*”. Since that time, the notion of storing all of ones accumulated digital items has been a topic of interest, for many researchers. In today’s society, this practice is known as 'lifelogging' and refers to the process of automatically recording aspects of one's life in digital form (Doherty, Caprani, et al. 2011). As described by Dodge and Kitchin (Dodge & Kitchin 2007), "*A life-log is conceived as a form of pervasive computing consisting of a unified digital record of the totality of an individual's experiences, captured multimodally through digital sensors and stored permanently as a personal multimedia archive*". Lifelogging has many benefits, Sellen and Whittaker (Sellen & Whittaker 2010) summarize these as "*the five Rs*":

1. *Recollecting (mentally re-living specific life experiences),*
2. *Reminiscing (re-living past experiences for emotional or sentimental reasons, either individually or social in groups),*
3. *Retrieving (recovering specific digital information we've encountered over the years, for example, documents, email, and Web pages),*
4. *Reflecting (the reviewing of past experiences that may include examining patterns of about one's behaviour over time),*
5. *Remembering intentions (remembering prospective events in one's life)*

Whilst the practise of recording this information is known as lifelogging, the outcome can also be referred to as a human digital memory (HDM), as well as a lifelog. As defined by Kelly (Kelly 2007), "*A HDM is typically a combination of many types of media, audio, video and images*". These personal archives are constructed from a wide range of data sources, across various media types (Gurrin et al. 2008). As technology advances, and sensors become more prevalent within our environment, the range of data that we have access to is increasing.

Human digital memories are a digital representation of ourselves that evolve and grow alongside us and are seen as a window into our past. As observed by Bush (Bush 1945), "*The human mind operates by association. With one item in its grasp, it snaps instantly to the next that is suggested by the association of thoughts, in accordance with some intricate web of trails carried by the cells of the brain. It has other characteristics, of course; trails that are not frequently followed are prone to fade, items are not fully permanent, and memory is transitory. Yet the speed of action, the intricacy of trails, the detail of mental pictures, is awe-inspiring beyond all else in nature*". The area of human digital memories does not aim to replicate this process, but merely learn from it and emulate it as close as possible. Simply arranging photos, videos or documents into a human digital memory is not an accurate reflection of ourselves. The changes that our bodies were going through should also be incorporated into a human digital memory. As people collect more and more data there is a danger of "information overload" and inadvertently, significant mementos are being lost and forgotten. Harnessing this information into an accurate description of a person's life is a significant challenge.

Emotion is also an important component of our mental life; therefore, lifelogging tools should be able to capture these feelings (Ivonin et al. 2012). Recent research into the area of affective computing has demonstrated that the emotional states of people can be recognized from their physiological signals (Ivonin et al. 2012; van den Broek et al. 2009; McDuff et al. 2012). This data is highly personalized and unique to every individual. Its incorporation into a human digital memory allows us to reflect on how we were feeling at any stage of our lives. The development of smaller sensing devices and wireless communications is revolutionising the way in which this data can be obtained, ubiquitously (Pantelopoulos & Bourbakis 2010a). However, these devices are primarily used within wearable health-monitoring systems (WHMS). As Pantelopoulos and Bourbakis (Pantelopoulos & Bourbakis 2010b) observe, "*These systems represent the new generation of healthcare by providing real-time, unobtrusive, monitoring of patients' physiological parameters, through the deployment of several on-body and even intra-body biosensors*". As well as monitoring our health and wellbeing, the data generated from WHMS can also be used to enhance a human digital memory, by providing us with physiological data, which can be reasoned upon. The evolution in short-range communication technology, such as ultra-wideband radio technology (Hirt 2003), Bluetooth (Haartsen 1998) and ZigBee (Wang & Wang 2010) have allowed these systems to become more sophisticated (Bonato 2010). WHMS, composed of a number of different sensors, can measure a variety of parameters, including electrocardiogram (ECG), blood pressure, respiration, body and/or skin temperature etc. (Pantelopoulos & Bourbakis 2010a). Incorporating this multitude of data, into a human digital memory, allows memories to become more dynamic and personal to the user and enables a richer understanding about our health, level of activity and wellbeing to emerge. Physiological data, obtained through the use of wearable sensors, provides a wealth of information about affective, cognitive and physical state and is able to record long-term trends (Gilleade & Fairclough 2010).

Using pervasive devices, such as mobile phones, body sensor networks and embedded environmental sensors, a greater level of detail can be incorporated into the creation of a human digital memory. New possibilities will allow content about us, family and friends to be clustered and linked together, based upon a multitude of factors. This will include, but is not limited to, information from mobile and physiological computing. This data can even provide information on how we made others feel at that time. A human digital memory is comprised of many items and aims to record our entire lives. However, successfully bringing together these fragmented pieces of information, and reasoning over a lifetime's worth of data raises many challenges.

1.3 Research Challenges

As it can be seen, data is being generated at a tremendous rate. The advent of smart environments, pervasive and mobile computing has enabled users, and their surroundings, to become active participants in generating information. However, this data is very disjointed. As people become more interested in reliving their experiences, bringing together such information together, to create rich and interactive human digital memories raises many challenges. The fundamental challenge of human digital memory research is to enable us to relive anytime of our lives, through our digital data. Information amassed over a lifetime needs to be efficiently searched and composed so that any moment of our lives can be extracted, recalled and replayed. This is a significant challenge that raises many other secondary challenges, which are detailed below.

1. The components of a human digital memory

A human memory is not made up of a finite set of variables. However, it is safe to assume that the more information we can remember, the greater the level of detail, and the more vivid, a memory is. It is this idea that is of particular interest. Currently, devices, such as the *SenseCam* (Hodges et al. 2006) or *SenseWear Armband (SWA)* (Andre et al. 2006), focus on only collecting specific information, such as photos or physiological information. This method is limited because restricting the information that is being collected, results in a digital memory that is also limited. Instead of focusing on collecting a specific set of data items, the process of capturing human digital memory data needs to be flexible enough to adapt to the user's current situation, utilizing a variety of information sources. As stated above, the proliferation of pervasive devices has enabled any object, embedded with a sensor, to provide us with data. The culmination of this advancement is a world awash with sensing devices, which can record our every move and the environment around us. Each environment that we occupy offers a different set of devices and information. As such, an adaptive approach is required that can collect data using such devices.

2. Searching a lifetime of data

The vision of the *Memories for Life: managing information over a human lifetime* grand challenge is to help people manage and use their digital memories across their *entire* lifetime (Fitzgibbon & Reiter 2005). Collecting data over this extensive period of time yields a phenomenal amount of information. When human digital memories are created this enormous amount of data needs to be intelligently searched and the associated information succinctly brought together. As stated by Ranpura (Ranpura 2000), "*Memories are rich because they are formed through associations. When we experience an event, our brains tie the sights, smells, sounds, and our own impressions together into a relationship. That relationship itself is the memory of the event*". Whilst humans can do this type of processing, subconsciously, in a matter of nanoseconds, creating these associations, digitally, poses a greater challenge. The complex and heterogeneous nature of a human digital memory means that the simple ranked retrieval of information is unlikely to support many of the user's information searching tasks (Kelly & Jones 2007). Furthermore, queries that require sophisticated interpretation need to be efficiently handled (Fitzgibbon & Reiter 2005). For example, queries such as, "Does this location make me happy?" or "When have I been active?" require an intelligent method of data analysis that enables multi–dimensional queries to be executed across a vast amount of data. Consequently, the system needs to learn about its user. Semantic web technology and machine learning is seen as a way to overcome this challenge. Intelligent search, instead of keyword matching, and query answering is facilitated and provides a way to search data from distributed sources, irrespective of its format (Fensel et al. 2011). Using a matrix representation of the data, allows the searching of this information to be treated as a machine learning problem, based on the similarities in a vector object. Consequently, a wider range of information can be included in the memory; the user is not limited by needing to have a pre-existing knowledge of the information.

3. Structuring data into a representation of a human digital memory

Structuring lifelogs is a major challenge in lifelogging systems, any system needs to present these logs in a concise and meaningful way to the user (Kikhia et al. 2011). Past research into the elements of human memory can be drawn upon so that human digital memory data can be structured into a reasonable representation of a memory. Episodic memory is a form of memory that relates to memories about personally experienced occasions. The information concerned is mainly about the subject's experiences of temporally dated episodes or events, and the temporal-spatial relations between them (Tulving 1993; Tulving 1984). This type of memory is concerned about happenings, in particular, places at particular times, or about "*what*," "*where*," and "*when*" (Tulving 2002). When an event or "episode" is remembered we usually answer these three points subconsciously. Primarily, we tend to remember where we were, the time of the event and what happened. This temporal episode, which has been brought to the forefront of our mind, can be imagined as a "box" of an event. All the information, from a specific time, is searched and grouped together in one place. In this instance, an interactive viewer is essential. By allowing users to "go into" their memories and to see numerous data items, such as temperature, location and emotions, could lead to the augmentation of group memories and can benefit various aspects of people's lives.

4. The longevity of human digital memories

As previously stated, the goal of the *Memories for Life* grand challenge is to help people manage and use their digital memories across their *entire* lifetime (Fitzgibbon & Reiter 2005). As new technologies emerge, data types become obsolete. Data needs to be accessible, regardless of time. This is an important challenge and one that Fitzgibbon and Reiter (Fitzgibbon & Reiter 2005) reiterate, "*How can we ensure that data is still accessible in 50 years time, despite inevitable changes in software, hardware and* [data] *formats?*" The use of linked data, semantic web principles and the Resource Description Framework (RDF) is seen as a way to alleviate this difficulty. The generic structure of RDF makes data interoperability and evolution easier to handle as different types of data can be represented using the common graph model (Cheung et al. 2005). The use of RDF enables data to be incorporated into a memory, irrespective of its format. As time goes on and new devices and formats emerge they can, nevertheless, be incorporated into a human digital memory, using this method. This is reiterated by the W3C (W3C 2004a), who comment that, "*RDF has features that facilitate data merging even if the underlying schemas differ, and it specifically supports the evolution of schemas over time without requiring all the data consumers to be changed*". As new standards become available their data, as well as data collected ten years ago, for instance, need to still be incorporated in a human digital memory.

Addressing the challenges, raised above, is important so that a more realistic representation of a memory can be constructed and content–based searches can be performed. Processes need to be developed in order to store and share memories in a more flexible way. These challenges have helped to formulate the reasoning that human digital memories should not be "tied-down" to a fixed number of information sources. The more data that can be accessed the more detailed the memory will be. Since this data is highly heterogeneous, an intelligent way of searching this information is also needed. Semantic web principles, RDF and using a similarity approach, are seen as a way to alleviate this problem. As stated above, RDF has features that facilitates data merging even if the underlying schemas differ, and it specifically supports the evolution of schemas over time without requiring all the data consumers to be changed (W3C 2004a). Therefore, as long as the raw data is transformed into a RDF model, any piece of information can be included in a human digital memory. Treating the searching of these RDF documents as a clustering problem also eliminates the need to produce complex and exact queries. This enables a wider range of information to be utilized in the creation human digital memories.

1.4 Novel Contributions

Addressing the challenges above, this project presents a new and novel framework that builds on the nomadic nature of people, ubiquitous computing, physiological computing, machine learning, cloud computing and ad-hoc networking (Dobbins, Merabti, Fergus, Llewellyn-Jones, et al. 2013). As such, the system creates rich and interactive digital memories, as well as learning about its user. In achieving this the DigMem system is presented, which is composed of three components – Mobile DigMem (MoDM), the DigMem Server and the DigMem Web Application. The project offers a new insight into how digital memories can be created, as well as proposing a way for us to learn about ourselves, through our data.

As sensors, smart objects and smart environments become more prevalent the ability to produce and retrieve information, from a variety of objects, is growing. Information about ourselves and the world around us can be captured and used to create human digital memories. As we move through different environments, the data sources that we have access to change. However, current lifelogging systems only collect a limited amount of information, such as photos, videos or emails, from a fixed number of devices. For instance, implementations such as *SenseCam*[4], *SenseWear Armband*[5] *and ActivPAL*[6], are very static in the variety of data that is collected. This method is limited, as other sources of information are left out of the memory. Human digital memories need to adapt, depending on the information available at the time. The Mobile DigMem (MoDM) (Dobbins, Merabti, et al. 2012b) middleware platform has been developed to address this issue by providing an open, flexible and extendable method of obtaining the use of device-specific services for the purpose of collecting HDM data. In this sense, any MoDM compliant device is capable of providing us with information, thus not limiting the number of devices that can be used to collect data. This is beneficial as the user is not limited to a single device to collect data, thus a more detailed memory is created, as the variety of information that is accessible increases. This development addresses the first research challenge (identified in section 1.3 Research Challenges) of flexibility by using pervasive devices, within the user's current environment, to collect memory data; as such, devices present in one environment might not exist in another, thus each memory is different. For example, a memory created in the home uses different devices than that of a memory created in the city centre.

[4] http://research.microsoft.com/en-us/um/cambridge/projects/sensecam/
[5] http://sensewear.bodymedia.com/
[6] http://www.paltech.plus.com/products.htm

This approach provides a unique method of building digital memories, from multiple distributed data sources and offers a much richer insight into our captured memories.

Once data has been collected it is transformed into human digital memory vectors and then into RDF, which can then be further processed into other formats, depending on current standards (Dobbins, Merabti, Fergus, Llewellyn-Jones, et al. 2013). This is another flexible aspect of the system that addresses the fourth research challenge (identified in section 1.3 Research Challenges) of the longevity of memories. The system is not "tied-down" to specific file formats. This is achieved because RDF can support the evolution of schemas over time, without requiring all the data consumers to be changed (W3C 2004a). This is beneficial because as new standards become available this data, as well as data collected twenty years ago, for instance, can all still be used in the creation of human digital memories. This system has the strength to create human digital memories for an extended period of time. Memories can be built from a variety of information, from different resources, which have been collected from different times of our lives.

Any human digital memory system needs to allow people to use and create memories "*across their entire lifetime*" (Fitzgibbon & Reiter 2005). As such, as more data is collected over a period of years and decades searching becomes problematic, due to the volume of information. Current search methods are limited as searching is based on keywords, as is the case in Microsoft's *MyLifeBits* (Gemmell et al. 2002). Furthermore, searching data by executing complex queries is also problematic as the user requires a precise knowledge of the data. For example, SPARQL (W3C 2008) is a very complex language, and if the queries are not constructed precisely, then false results can occur. Finding information becomes more difficult as more data is accumulated. In comparison, DigMem uses probabilistic approaches and analyses human digital memory data by utilizing a number of machine learning methods. This is beneficial as the user does not have to be precise in defining their search criteria and an exact query match is not required. In achieving this a single search space has been developed, composed of human digital memory feature vectors (Dobbins, Merabti, Fergus & Llewellyn-Jones 2013b). Raw datasets can become extremely large; therefore, extracting features (such as mean, median, standard deviation, etc.) enables information to be extracted so that the dimensions of these datasets are reduced. Numerous features can be extracted from a variety of signals, as such the bigger the feature space, the more detailed the resulting human digital memory is. The creation of human digital memory vectors can become very large. In spite of this, machine learning algorithms are able to deal with these sets of big data quite easily (Dobbins, Merabti, Fergus & Llewellyn-Jones 2013b). This approach addresses the limitations of searching memory logs, as recognized in the second research challenge (identified in section 1.3 Research Challenges).

Utilizing various machine learning techniques, DigMem is able to answer specific questions about the user's data so that this information can either be classified or clustered. For instance, in the case of classification, questions such as, "Have I been running?" or "Does this location make me happy?" can be answered. This is beneficial as the information can then be used to gain a greater insight into ourselves. For instance, a certain location might invoke subconscious physiological responses that could only be known by questioning our data.

In terms of clustering information, another set of questions, such as, "When was I active?" or "When was I least active?" can be answered. This approach enables related pieces of information to be brought together so that memory boxes can then be created. In this context, a memory box is a graphical way of presenting data. Instead of presenting the user with all of their data simultaneously, initially the data types that DigMem supports are displayed and upon opening a window the related information is displayed. For example, opening the photograph window displays photographs or opening the map window plots the user's movements on a map. This method does not overload the user with data and enables them to explore their information. They are a method of condensing a massive amount of raw information into graphical items (Dobbins, Merabti, Fergus & Llewellyn-Jones 2013a). Memory boxes also address the third research challenge (identified in section 1.3 Research Challenges) as data is structured concisely. Interaction with our memories is fundamental. Users are able to explore their memories to see various pieces of information, such as photos, location, accelerometer and heart rate readings. Our actions and how our bodies have changed over time are vividly portrayed (Dobbins, Merabti, et al. 2012a). Users are able to reason over their behaviour by retrieving any time of their lives, which is then displayed in this succinct manner. The ability to answer such abstract questions is a unique feature that has not been seen before.

The novel contributions, outlined above, demonstrate how memories can be created across diverse environments and how a lifetime of information can be searched and brought together. The composition of memories is a widely debatable subject. However, a fundamental view, which can be agreed upon, is that the more information contained in a memory the more vivid, and useful, it is. Capturing memory data should not be limited to a fixed number of devices, as is the case with current applications, for example, (Doherty, Moulin, et al. 2011; Doherty, Caprani, et al. 2011; Chennuru et al. 2010; Lee & Dey 2009; Kikhia et al. 2010). Human digital memories need to be reflective of our present environment so that we can gain a deeper understanding of our surroundings and ourselves. For example, certain places might invoke certain physiological responses that we are unaware are happening. More specifically, the collection of memory-related information (images, video, physiological data and so on) needs to occur using ubiquitous ad-hoc services, prevalent within the environments, we occupy. This is likely to happen without us necessarily being aware that memories are being created. This will remove the need to manage the growing number of information sources that require conventional tools to achieve this, for example, a camera to take stills and video.

1.5 Thesis Structure

The research presented in this thesis aims to create a framework that allows rich and interactive human digital memories (memory boxes) to be created, whilst also producing a system that learns about its user. In order to explore and explain this idea, the thesis is divided into eight chapters.

Chapter two explores the background of human digital memory research. More specifically, focusing on wearable systems, mobile, physiological and smart devices, the methods that are used to capture human digital memory data are presented. Once it has been established how data can be collected, a more in-depth look at how digital memories are created is also explored. This section looks at earlier applications that have been developed and illustrates how previous research has structured human digital memory data. Next, we examine how data can be linked, to form human digital memories. This section explores the area of the semantic web and previous research within that area. This chapter is then concluded with a short summary of the background of the area.

Chapter three focuses on the design of the system. In this section, the design specifications are described, as well as the system requirements. An overview of how DigMem captures human digital memory data and creates a memory is also presented. Furthermore, this chapter describes the design of the three components of the system – Mobile DigMem (MoDM), the DigMem Server and the DigMem Web Application.

Chapter four describes the implementation that has been undertaken. This includes describing the development of the three components of the system (MoDM, the DigMem Server and the DigMem Web Application). In order to gather human digital memory data a mobile application, MoDM, has been developed (Dobbins, Merabti, et al. 2012b). This application finds device-specific services, gathers data from a number of personal devices and sends it back to the user’s mobile device. The data is then transferred to the DigMem Server (Dobbins, Merabti, et al. 2012a), which processes the information. Afterwards, the web application is used to search the human digital memory data and transform this information into a memory box (Dobbins, Merabti, Fergus & Llewellyn-Jones 2013a). The web application also enables the user to ask the system questions about their data.

Chapter five discusses a case study that has been devised to demonstrate the design and implementation of the system. It also expands on the idea of a memory box and the information that it contains. This chapter also discusses other application areas that the system can be used for.

Chapter six discusses the evaluation methodology that has been used to process raw human digital memory data. This data is then used to form a feature dataset, which is used in the evaluation, in chapter seven. This chapter also includes a discussion on the supervised and unsupervised machine learning algorithms that have been chosen for the evaluation.

Chapter seven evaluates the system and presents the results from the various machine learning algorithms that have been used to search the created feature dataset from chapter six. An assessment between supervised and unsupervised machine learning has also been made. This chapter also presents a comparison of DigMem against existing approaches.

The thesis is concluded in chapter eight. A summary of the project, and future work to be undertaken, are discussed here. As well as these points, the chapter also discusses how this work has contributed to knowledge and provides concluding remarks about the project.

Chapter 2

Background

In the previous chapter, an overview of human memory and the concept of human digital memories have been introduced. Furthermore, the research challenges of the area have been identified, and the novel contributions that the project provides have also been discussed. This chapter explores the background of lifelogging and presents an overview of current work in the area. As such, the areas of capturing, searching and organising human digital memory data are examined.

Research into capturing and creating human digital memories has received a great deal of attention, from researchers, over the last few decades. Since the *Memex* (Bush 1945) in 1945, research into how aspects of our lives can be captured and organised, have been investigated. Over time, this vision of storing accumulated items has evolved into digitally capturing information about ourselves and our environment. The culmination of this practise has been to continually capture content, with the aid of wearable systems. Since the 1980s, Steve Mann has been the pioneer of such lifelogging systems (Doherty, Caprani, et al. 2011). Mann's *EyeTap* device (Mann et al. 2005; Mann 2004; Mann 1997) has evolved over the last thirty years, and, with the progression of smaller components, has "*become more feasible for commercialisation and mass production*" (Mann 2004). The system consists of eyeglasses, with a camera, display and diverter, and enables precise lifetime personal experiences to be captured (Mann et al. 2005). From 1994 to 1996, Mann continuously broadcast full-motion video footage of his everyday life to the Internet, through his head-mounted wearcomp system (Crete-Nishihata et al. 2012).

Mann paved the way for wearable lifelogging systems. Many other projects have taken this vision further, to produce their own implementations. For instance, Healey and Picard's (Healey & Picard 1998) *StartleCam*, uses a wearable camera and sensors to "*Capture events that are likely to get the user's attention and to be remembered*". Whilst, Dickie *et al.*'s (Dickie et al. 2004) *eyeBlog* captures video streams based on eye contact with the user. A revolutionary device, which has dominated the area of lifelogging, in the last few years, has been Microsoft's *SenseCam* (Hodges et al. 2006) – a small wearable digital camera, with built-in sensors, that is designed to take photographs automatically, without user intervention (Hodges et al. 2006). This device has been worn continuously, by Cathal Gurrin (Gurrin 2012; Doherty et al. 2009). He has been wearing the device since June 2006, from morning until night (Gurrin 2012). Between May 06 – Dec 08 he recorded 2,579,455 images (3,080/day) = 29,301 events (35/day). During this time; the device was worn for an average duration of 14 hours 22 minutes per day and managed to capture 846 days (90.2%) worth of information; 92 days had missing data (9.8%) due to missing sensor files (Doherty et al. 2009). The *SenseCam* has also been worn, continuously, by Gordon Bell (Gemmell & Bell 2009), from Microsoft. His collected information was then incorporated into their *MyLifeBits* project (Gemmell et al. 2002; Gemmell et al. 2004).

Most recently, Google Glasses (Google 2012) takes this concept further with their Project Glass. These glasses project information and entertainment onto the lenses (Bilton 2012). Although this concept is geared toward augmented reality, their potential, within the lifelogging field, is clear. These technical developments have opened the possibility for users to passively capture and store comprehensive representations of their personal experiences (Crete-Nishihata et al. 2012). Continuous lifelogging is slowly becoming socially acceptable and, as Mann observes, "*Recent prototypes have been gaining acceptance in social situations. This can be attributed partly to miniaturized smaller units, but also to dramatic changes in the attitude toward personal electronics*" (Mann et al. 2005). As these devices become smaller, lifelogging will only increase. The following is an overview of the current work within this area.

2.1 Capturing Human Digital Memory Data

When thinking of capturing memories, the first thing that usually comes to mind is purposely taking photos or videos of significant events. However, in recent years, there has been a shift in the data that can be obtained and the fashion in which it is being done. This shift has resulted in the effortful selective capturing of moments being replaced with lifelogging, which seeks to be effortless and all-encompassing in terms of data capture (Sellen & Whittaker 2010). In other words, content is being captured constantly and with minimal user involvement (i.e. with the use of automatic, wearable, devices). Mobile devices and sensor equipment are now able to capture a more comprehensive record of everyday life, as continuously as possible (Sellen & Whittaker 2010). These devices offer an innovative, and less obtrusive, method into capturing content ubiquitously and can be used to document our entire lives. A vast collection of information can be recorded about ourselves, at any time. Automatically recording this data, and quantifying how a given aspect of our body changes over time, provides an insight into our underlying behaviours (Gilleade & Fairclough 2010; Doherty, Caprani, et al. 2011). This increasing trend is one that will only strengthen over time; thus presenting us with a more diverse range of data that is accessible and new ways to capture such information.

2.1.1 Wearable Lifelogging Systems

Microsoft’s *SenseCam* (see Figure 1), as stated above, is a revolutionary lifelogging device, which is capable of storing up to 30,000 images, in total (Hodges et al. 2006). By default, photos are captured and stored every thirty seconds, usually resulting in 1,500 – 2,500 photos per day (Lee et al. 2008). The device contains a digital camera, with a fisheye lens, and multiple sensors, including sensors to detect changes in light levels and a passive infrared sensor to detect the presence of people (Hodges et al. 2006). A log file is also created that records the data from the sensors every few seconds and the reason for taking each photograph (for example, manual shutter press, timed capture, or significant change in sensor readings). Each day, the photos can be uploaded, via Universal Serial Bus (USB), onto a computer (Hodges et al. 2011) and viewed using Microsoft’s simple viewer application (Microsoft Research 2011).

Figure 1 Microsoft's *SenseCam*

[Permission to reproduce Figure 1 has been granted by Taylor & Francis Group (www.tandfonline.com) from the paper (Hodges et al. 2011)]

Since 2010, over fifty research institutions and labs worldwide have used the *SenseCam* in their research (Hodges et al. 2011). The device was originally developed as a retrospective memory aid (Hodges et al. 2006) and various studies have used it as a memory aid device and to monitor behaviour (Lee et al. 2008; Doherty, Moulin, et al. 2011; Sellen et al. 2007; Doherty, Caprani, et al. 2011; Lindley et al. 2009; Kalnikaite et al. 2010; Kelly et al. 2011; Byrne & Jones 2008; Kikhia et al. 2010; Hodges et al. 2011; Browne et al. 2011; Pauly-Takacs et al. 2011; Crete-Nishihata et al. 2012; Brindley et al. 2011; Kelly et al. 2012; Qiu et al. 2011; Wang & Smeaton 2013). Browne *et al.*’s (Browne et al. 2011) study highlights its effectiveness as a memory aid tool. As stated by the authors, in this investigation, the *SenseCam* was used to generate images for rehearsal, promoting consolidation and retrieval of memories for significant events in a patient with memory retrieval deficits. The results indicated that regularly reviewing the *SenseCam* images resulted in superior recall of recent events, compared to reviewing a written diary, and this effect was maintained in the long term. This study also supports Crete-Nishihata *et al.*'s (Crete-Nishihata et al. 2012) work that used the device with patients that had Alzheimer's or mild cognitive impairment. The results, from this work, suggest that reviewing *SenseCam* images, of personal events, can support episodic recollection of the experiences over time (Crete-Nishihata et al. 2012). Whether it is used for private reflection or as a memory-aid device repeated viewings of *SenseCam* images play an important role in the long-term retention of personally experienced events (Pauly-Takacs et al. 2011).

Although this device is paving the way for lifelogging technologies, its application within behavioural studies, is also gaining momentum. These investigations are opening up a completely new way in which human digital memories can impact our physical health. One such approach has been Lindley *et al.'s* (Lindley et al. 2009) study on creating 'small stories' based around the *SenseCam* images. By reflecting upon the images and discussing and re-creating memories, associated with the photos, these helped the users to reflect upon daily life and to identify periods of sedentary behaviour. This study was supported by Doherty *et al.* (Doherty, Caprani, et al. 2011) who stated that "*after participants looked at their images, they were prompted to change their lifestyle by, for example, cycling instead of driving, taking up exercise, and spending more time interacting with their children*". This study also emphasizes the importance that visual illustrations of behaviour play on changing our habits and has been particularly useful for developing automatic classifiers for graphical life–logs to infer different lifestyle traits or characteristics (Doherty, Caprani, et al. 2011). It has also been beneficial because of the amount of data that was collected and used. The study included 95,000 manually annotated images and 3 million lifelog images, obtained from 33 individuals sporadically over a period of 3.5 years. These images were later used to identify 22 different lifestyle traits, which were then analysed to inform individuals about their future wellbeing (Doherty, Caprani, et al. 2011).

In contrast, the *SenseCam* has also been used within the area of travel research and to monitor sedentary behaviour. Kelly *et al.* (Kelly et al. 2011; Kelly et al. 2012) have used it within the area of travel research, in two studies. In their pilot study, the device was used to investigate its effectiveness in tracking the journeys that adult participants made. Participants were required to wear the device for one full day and to also keep a diary of that day. Initial results indicated that when users recollected their journey times, in self-reported diaries, they tended to over-report them. They found that in order to accurately track sedentary behaviour the *SenseCam* was a useful tool, as it provided the visual "back-up" data that was needed to identify errors in the diaries. The second study (Kelly et al. 2012), again involved measuring self-reported journeys; however, this time school journeys were studied in participants aged 13-15 years. For one week, the participants wore the *SenseCam* to and from school. The authors reported that there was very little bias on self-reporting. From the 135 journey stages, 31.4 hours of travel were reported, whilst the wearable camera recorded 31.1 hours. At the group mean level, self-reported journey stage durations were ten seconds longer per journey (Kelly et al. 2012). This is in contrast to the previous study (Kelly et al. 2011), where over-reported journeys by adults was much higher, with 70% of journeys being over reported, compared to 59% for the adolescents. Nonetheless, as the authors conclude, the results still indicate that the *SenseCam* is a useful tool, and that self-reporting is a poor measure of individual journey behaviour; the two methods should not be used interchangeably.

The *SenseCam* is seen as the leading technology in capturing memories. However, other projects have also been undertaken, for the purposes of lifelogging and monitoring behaviour. Blum *et al.'s* (Blum et al. 2006) *inSense* is one such system. It uses acceleration, audio and visual sensing equipment, to perform real–time context recognition (Blum et al. 2006). This method is quite interesting, in terms of the device setup. It is not too bulky and looks like it can be easily worn. However, the approach relies on the user annotating the data, at a later time. A task often overlooked and neglected by users, especially when the data is a "*minute-by-minute*" (Blum et al. 2006) account of their activities. This can be a tedious and laborious task. In contrast, Belimpasakis *et al.* (Belimpasakis et al. 2009) have implemented a "*client–server platform that enables lifelogging, via mobile context collection, and processes the data so that meaningful higher-level context can be derived*". However, their system focuses on emphasising social and group features by providing a way for shared content to be browsed, via associations and context-based content retrieval (Belimpasakis et al. 2009). The data collected is also limited to photos and location information. Chennuru *et al.* (Chennuru et al. 2010) have also used a mobile device for lifelogging. Their system comprises a helmet, mounted with a Nokia N95 phone, which records accelerometer, location, photo, sound, rotation and Wi-Fi signal strength data (Chennuru et al. 2010). Unlike the *inSense* (Blum et al. 2006) system, this set up is less practical to wear daily. It also relies on the user annotating their data and the presentation of the data in a calendar format that could make it harder to review memories, especially ones that occurred many years ago. Furthermore, more than one memory could be captured on any one day; this lay out would make it quite difficult to sort through multiple memories that occurred on the same day.

Meanwhile, *MemoryLane* (Kalnikaite & Whittaker 2011; Kikhia et al. 2009), "*allows people to capture, actively organize and reflect on digital representations of mementos relating to people, places and objects*". This work centred on distributing a Sony Visual IC Recorder™ to 31 participants, which they used to take photos and record audio. For three days, the users captured pictures and audio narratives about significant people, places and objects in their lives (Kalnikaite & Whittaker 2011). This study illustrated that human digital memories are important for reflection and that people are interested in reflecting upon their digital artefacts. However, the system relies on the user actively taking pictures and recording audio data. In order to construct a truly reflective digital memory this data, along with other pieces of information, needs to be collected with as little user interaction as possible.

Whilst the technologies and methods discussed are a useful starting point, within the area of lifelogging, novel solutions are needed that include much richer data capturing capabilities and require a less obtrusive and expensive, approach. Although the *SenseCam* has been hailed as a "*revolutionary pervasive device*" (McCarthy et al. 2007) it has its limitations. One drawback is that the data that is captured is limited, as only photos and a very small amount of sensor data are recorded. Whilst photos are a good place to start, memories are made up of so much more than that. Emotions and environmental factors also contribute to the composition of memories. These are important factors that need to be factored into creating an accurate human digital memory. Another drawback is in relation to accessing the data. Images have to be manually downloaded periodically, which can be very time-consuming and mundane for the user. The device is also very expensive. Exploiting the services of devices, already present within our environment, is a far less-expensive approach. One way that this is envisioned is with the use of mobile and peer-to-peer (P2P) technologies. The explosion of mobile computing and the sharing of content ubiquitously have enabled users to create and share memories instantly. Access to different data sources, such as location, movement, and physiology, has helped to create a data rich society where new and enhanced memories will form part of everyday life. P2P systems have also increased in popularity over the years, due to their ad hoc and decentralized nature. Mobile devices are "smarter" and are increasingly becoming part of P2P systems; opening up a whole new dimension for capturing, sharing and interacting with enhanced human digital memories, as we shall see in the following section.

2.1.2 Mobile Technologies

The mobile era is upon us and with these devices increasing in sophistication and decreasing in size, they are the ideal candidate to capture and share life memories. Memories are often impulsive events and are better suited to being captured and shared on a portable device. These devices are adequately compact to be carried around and are sophisticated enough for sharing content amongst users. Olsson *et al.* (Olsson et al. 2007) reiterates this point by stating that "*Mobile phones offer natural opportunities for collecting instant digital pictures and videos because of their immediate availability to users*". Mobile computing, along with P2P networking and the notion of collecting digital life memories, has begun to generate a great deal of interest. Furthermore, an extensive amount can be learnt from existing work in this area.

One such approach is *JMobiPeer*, a P2P middleware for mobile ad hoc networks (MANETs). It has been designed to work on Java 2 Platform Micro Edition (J2ME) enabled portable devices and is compatible with the JXTA (Juxtapose) P2P protocols (Bisignano et al. 2005). The *JMobiPeer* system has obtained good results, in relation to the discovery time and bytes exchanged. However, as Wang and Motzfeldt (Wang & Motzfeldt 2007) observe, it "*has only been tested on emulators on standard PCs. This is likely due to high requirements on CPU and memory from running the framework*". Therefore, real-world testing on actual mobile devices would be required to determine whether the application could be executed on devices that are far less capable.

Taking an opposing view, Tsai *et al.* (Tsai et al. 2009) created a *Mobile Social Software (MoSoSo)* application that runs on mobile devices. The application "*allows users to discover, communicate and share resources with each other*". It is a P2P social networking application in which users can view friends, share files, message each other, add new friends and edit their own profiles. Taking a similar approach, Park and Cho (Park & Cho 2010) discuss how a mobile social network can be constructed by obtaining the life–logs of users and how this network can be used to share information. In this work, lifelogs were collected using a Samsung smart phone and a mobile social network was built with semantic relations using a Bayesian network model (Park & Cho 2010).

Meanwhile, Palazzi *et al.* (Palazzi et al. 2009) created the *P2PBluetooth* platform, a "*proof–of–concept file sharing application for mobile phones that works through Bluetooth connectivity*". In particular, their work focuses on creating a P2P network using J2ME. An interesting aspect to their work is file sharing and the idea of proximity. Bluetooth includes several power conserving features (Shorey & Miller 2000), thus it is a very useful wireless technology for use with mobile devices. However, since Bluetooth has a limited range, communication is limited to very short distances and is better suited to applications that open sporadic short-lived connections (Riva & Kangasharju 2008), which is not in alignment with human digital memory applications that collect data constantly. Our work aims to be less restrictive, by creating ad-hoc P2P networks to share information. In other works, Ismail *et al.* (Ismail et al. 2009) have designed a framework to identify personal memories, through photograph image analysis and a reporting system. As the authors have described, their system uses the JXTA

P2P networking architecture to create a virtual network of peers who share their serendipitous moments among themselves. This work is particularly interesting due to the technologies used in the implementation of the P2P system.

In other works, Hamm *et al.* (Hamm et al. 2013) have used smartphones as a platform to facilitate the collection of data. Using multiple annotation tags, the user reviewed these daily logs and segmented the information into meaningful events. After this, the *bag-of-words representation*, along with state-of-the-art classifiers, was used in order to predict the tags. This work is interesting due to the algorithms that have been used and the results that were obtained. However, the system only uses one device for data collection, which limits the types of data that are used. Memories are not composed of a definitive set of data items and benefit from multiple sources of information.

In terms of capturing data, mobile devices fit well with our natural ability to move within our environment. These tools and standards provide mechanisms for interconnecting device functionality as independent network discoverable services. However, this alone does not support the memory structures required. There is a need to build additional middleware services to achieve this, and this will be the focus of future work. Memories are not isolated static events, but rather a continuous sequence of experiences contextually linked and created within and across different geographical areas within the environments we occupy. This will be a key requirement in our work, along with building additional middleware services, to achieve the memory structures required. In this way, the service's scheme provides a plug-and-play platform for memory data sources, which can be exploited by any digital life memory middleware.

As well as utilising mobile devices, which most people carry around with them, another source of information lays within our own bodies and the surrounding environment. As people become more interested in monitoring their health, physiological devices are becoming smaller and more practical to wear on a daily basis. These devices can record a range of information about ourselves, and coupled with other collected data items, can offer a way to see how we were feeling at any point in our lives. As well as using information from ourselves the advent of smart devices has enabled any physical object to provide us with information. Utilizing the information that these devices provide adds another dimension to our human digital memories. These devices are capable of providing us with a range of information, as we shall see in the next section.

2.1.3 Physiological and Smart Devices

In terms of obtaining other types of, more personal, information sensor-based systems are emerging as a new way to capture our every move and to monitor our health and wellbeing. The development of smaller sensing devices, and wireless communications, is revolutionising the way in which a subject can be ubiquitously observed (Pantelopoulos & Bourbakis 2010a). These devices offer a new generation of inexpensive, unobtrusive wearable or implanted devices (Pantelopoulos & Bourbakis 2010b), which are capable of capturing content over a lifetime. As well as monitoring the wellbeing of the wearer, this data can also be used to enhance human digital memories. These systems provide highly personalized data and are another source of information, which can be reasoned upon. In order to add more depth and detail, about ourselves and the surrounding environment to memories, incorporating this technology is essential.

One such system is *activPAL*, which houses a single–axis accelerometer (Taraldsen et al. 2011). The *activPAL* accelerometer has also been used to determine habitual behaviour, whilst also determining the interplay between sedentary behaviour and periods of physical activity (Lord et al. 2011). In terms of creating human digital memories, this system can be used to illustrate the movements of a user throughout the day. Nevertheless, the context in which those movements occurred is unknown, without the use of a visual aid.

Taking a similar approach, Lee *et al.* (M. W. Lee et al. 2011) have created a real-time single tri-axial accelerometer-based personal life log (PLL) system. The system is capable of recognizing human activity and generating exercise information. It recognizes the occurrence of activities, based on the statistical and spectral features of the accelerometer signals. As the authors have described, once an activity has been recognized the system further estimates exercise information that includes energy expenditure based on metabolic equivalents, stride length, step count, walking distance, and walking speed. In this study, the system has been tested against six daily activities (lying, standing, walking, going-upstairs, going-downstairs, and driving), via subject-independent and subject-dependent recognition, on a total of twenty subjects. As described by the authors, an average recognition accuracy of 94.43 and 96.61%, respectively has been achieved. In order to ascertain the subject's true activities, voice annotation was also used and the number of steps was counted using a pedometer (M. W. Lee et al. 2011). Whilst this study is capable of recognising activities, annotating those times with voice notes is an impractical method of validating activity. The use of a visual aid would provide a better illustration of the activities. For instance, visual diaries would have provided a better depiction of such times. These items would have been more useful to help validate the classifiers accuracy, rather than vocal descriptions. Furthermore, the collection of this data can occur automatically, through the use of a wearable camera, which is more convenient for the user.

A more sophisticated device is the development of armbands, which house several sensors so that a variety of physiological data can be collected. The *SenseWear Armband (SWA)* collects data from a bi-axial accelerometer, galvanic skin resistance (levels of sweat), heat flux (heat dissipated from the body), and skin and near body temperature, to estimate energy expenditure (EE) and step count (Dwyer et al. 2009). The device has been used within Dwyer *et al.*’s (Dwyer et al. 2009) work to determine its accuracy for estimating step count during treadmill walking. The authors reported that in this instance, the *SWA* provided a reasonably accurate measure of step count compared to manual counting during treadmill walking. The *SWA* has also been used to monitor its practicality in measuring physical activity in women with rheumatoid arthritis, in a similar way to measuring decreases in sedentary behaviour (Almeida et al. 2011). The results from this study concluded that 89% of the participants felt comfortable wearing the device and that the *SWA* is a viable and practical method of quantifying physical activity and may be useful to monitor effectiveness of interventions to increase activity in people with rheumatoid arthritis. In relation to creating memories, this system can also be used to illustrate the movements of a user throughout the day. However, like the *activPAL* system (Taraldsen et al. 2011), the context in which those movements occurred is unknown, without the use of a visual aid. For instance, visual diaries would have provided a better depiction of such times. These items would have been more useful to help validate the classifiers accuracy, rather than vocal descriptions. Furthermore, the collection of this data can occur automatically, through the use of a wearable camera, which is more convenient for the user.

Whilst armbands are capable of measuring levels of physical activity, and energy expenditure, wearable systems (Paradiso et al. 2005; Taraldsen et al. 2011; López et al. 2010; Figueiredo et al. 2010; Pantelopoulos & Bourbakis 2010b; Pacelli et al. 2006; Pandian et al. 2008; Coyle et al. 2010; Curone et al. 2010; Langereis et al. 2007; Matthews et al. 2007) have been designed to measure a user’s biological signals. One such approach has been Matthews *et al.*’s (Matthews et al. 2007) “*Physiological Sensor Suite (PSS)*”, which gathers electrocardiogram (ECG), electromyogram (EMG), electrooculogram (EOG) and through-hair electroencephalogram (EEG) data. These types of data gather a variety of electrical activities from various parts of the body. ECG measures the heartbeat; EMG measures this activity in muscle tissue; EOG measures the activity of the eye, whilst EEG measures the activity of the brain. The information is then communicated, wirelessly, to a data logger, which the user is wearing. In terms of gathering information, this suite is particularly interesting because the sensors are quite small and “*no modification of the skin’s outer layer is required*” (Matthews et al. 2007). In other words, the user does not have to prepare their skin with special gel (as is common practise amongst other systems). In other works, the *BIOTEX* project (Coyle et al. 2010) consists of a textile-based system that collects and analyses sweat, by using a textile-based sensor capable of performing chemical measurements. The system monitors a number of physiological parameters, together with sweat composition, in real-time (Coyle et al. 2010). This work is interesting because the components are quite small; a 1 cm × 3 cm piece of superabsorbent (Absorbtex) has been used to collect the sweat. Furthermore, monitoring a number of physiological parameters together with sweat composition in real time is beneficial. For example, in the case of athletes who partake in endurance sports, analysis of sweat can give information on hydration levels (Coyle et al. 2010). However, since the system has been designed to analyse sweat, the analysis of non-strenuous physiological signals, for example, sitting or lying down would not be possible. Therefore monitoring a subject for a sustained period of time is not possible.

Whilst these systems can gather a range of information, they are quite cumbersome to wear for a lengthy period of time. Integrating sensors into everyday clothing, using "*smart fabrics and interactive textiles (SFIT)*" (Lymberis & Paradiso 2008), is a more practical solution. Sensors do not have to be placed on the body by a professional; therefore, the user can be monitored at any time (Lymberis & Paradiso 2008). One such approach, in this area, has been López *et al.*'s (López et al. 2010) *LOBIN* project. A combination of e-textile and wireless sensor networks have been used to provide an efficient way to support non-invasive and pervasive services (López et al. 2010). The system consists of a set of "*smart shirts*" that monitor heart rate (ECG), angle of inclination, activity index, and body temperature and a location subsystem, which monitors the patient's location (López et al. 2010). The information is then sent to the management subsystem, which processes and stores the data. This system is quite interesting due to the parameters that can be measured and the location system, particularly as patients are tracked indoors.

Meanwhile, Niazmand *et al.* (Niazmand et al. 2010) have integrated acceleration sensors into a pullover jumper, to detect falls. This garment is washable and houses eight acceleration sensors and an electronics unit. The electronics unit reads in and analyses the data from the sensors, and sends an alarm to the base station, if a fall is detected (Niazmand et al. 2010). This system is interesting because it is washable and can monitor a user for a long period of time. In addition, the alarm feature is useful because a number of different behaviours could be analysed and flagged for concern, not just falling. In theory, if minimal acceleration was detected, i.e. the subject was being sedentary for too long, this could trigger an alarm to a healthcare professional. In other works, Tyrer *et al.*'s (Tyrer et al. 2011) smart carpet has been developed to also monitor the elderly and to detect falls. Although smart carpets are primarily used in this manner (Tyrer et al. 2011; Aud et al. 2010; Fukui et al. 2012; Klack et al. 2010; Grossi et al. 2008), they could also be used to detect if a user has gained or lost weight and can be seen as a way to monitor weight over a significant period of time.

As well as measuring physiological signals, smart objects offer a way for inanimate objects, in our environment, to provide us with information, which can be included in our human digital memories. These devices link everyday items with information technology by augmenting ordinary objects with small sensor-based computing platforms (Siegemund 2004). By providing such context-aware services, these objects have the potential to revolutionize the way in which people deal with objects in their daily environment (Siegemund 2004). For example, Luo *et al.*'s (Luo et al. 2009) smart fridge application has been designed to perform dietary control, nutrition monitoring and eating habit analysis using sensors to monitor what food is entering and exiting the fridge

In other works, Yousefi *et al.*'s (Yousefi et al. 2011) proposed smart bed platform collects information from various sensors incorporated into the bed. The data is analysed to create a time-stamped, whole-body pressure distribution map. It then commands the bed's actuators to adjust its surface profile periodically to redistribute pressure over the entire body (Yousefi et al. 2011). In a similar approach, Yip *et al.* (Yip et al. 2009) have developed a flexible pressure monitoring system, consisting of 99 pressure sensors on a 17-cm x 22-cm sheet. The results from both studies indicate that the use of sensing technologies can accurately measure the pressure that is being sustained on the body. In other works, Manohar and Bhatia (Manohar & Bhatia 2008) have used ZigBee and bend sensors to create a low power, micro-controller based patient bed monitoring system. The goal of their work is to "*periodically estimate the pressure points and to maintain a database*". However, this approach is not flexible enough and is deemed "*mattress specific*". In a similar study, Wang *et al.* (Wang et al. 2011) developed a remote monitoring and caution system. Like Manohar and Bhatia (Manohar & Bhatia 2008), their system also uses the ZigBee network infrastructure. They have used sensors to monitor pressured positions for mobility-impaired persons on the bed (Wang et al. 2011). Four sensors are attached to potential ulcer areas, and a ZigBee sensor is integrated with several of the sensors. If a certain position is sustained over a 30-minute period, an alarm is triggered, and a message is sent to the caregiver's mobile device (Wang et al. 2011). These devices are mainly used to monitor elderly patients, who are susceptible to developing pressure ulcers. However, the information can still be incorporated into anyone's human digital memories to measure the amount of time spent in bed.

Taking a different approach, Frohlich and Murphy's *Memory Box* (Frohlich & Murphy 2000) and the *Living Memory Box* project (Stevens et al. 2002; Stevens et al. 2003) are physical boxes where radio-frequency identification (RFID) tagged items are either placed or stored, which trigger the replay of associated audio commentaries about the objects (Banks et al. 2012). More specifically, the *Living Memory Box* is "*a device and* *service to assist families in preserving memories in a variety of media forms*" (Stevens et al. 2003). The box records the appearance of physical objects that have been placed into it, together with audio narratives and metadata (for example, date, time and place), to support later retrieval (Bowen & Petrelli 2011). User feedback about the initial prototype was positive; however, users felt they wanted to be able to annotate their own mementos instead of having automatic annotation (Kalnikaite & Whittaker 2012).

Another approach that combines physical objects with the digital world is Kawamura *et al.*'s (Kawamura et al. 2007) *Ubiquitous Memories* project. RFID tags are embedded in objects, as well as in a central device that is attached to the user. This study assumes that every object in the real-world is implanted or attached to an RFID tag, when the user touches it the object's information is saved (Kawamura et al. 2007). This is similar to *MEMENTO* (West et al. 2007), a digital-physical scrapbook for memory sharing, where users move between digital and physical forms of a scrapbook. Users are able to create a paper scrapbook, which is automatically synchronized with a web-based version (West et al. 2007). However, the practicality of these approaches is limited. None of these projects has ever been fully developed to the point where it could be evaluated in ecologically valid deployments (Banks et al. 2012).

As it can be seen, many signals can be captured from the user. However, these methods are used in very separate fields and have rarely been combined. In order to form a more rounded snapshot of our lives these technologies need to work together. So that not only can a visual representation of experiences be recapped, but also the feelings and changes our bodies were experiencing when those events were occurring. Subsequently, by incorporating even more data, for instance, from smart objects and our environment, would reduce ambiguity further. For example, a higher heart rate than normal and an increase in sweat production could be attributed to many things. Presenting only this information as a memory is insufficient. However, if it was known that it was a hot day, by incorporating a temperature reading and that a photograph of the user doing physical activity was also obtained, then the context of the physiological data is known. This information could also be used to establish how different environments affect us. Bringing together data, from separate sources, enables a finer level of detail to be achieved, as the range of accessible information is increased. In order to integrate this data into human digital memories, advanced solutions are needed. A significant drawback is the ambiguity of physiological data, which can require extensive data analysis. Automatic analysis of this data would have to be performed in order to discern meaningful information to enhance our memories.

Current work aims to address these limitations by automatically gathering and linking a variety of data, from distributed data sources, which ordinarily would not be associated with each other. This data would then be semantically linked and used to form a memory box. In this perspective, a memory box would contain a number of items, including photos and location information, as well as several sensor readings, ranging from the temperature of the room to changes in physiological data. Various boxes are constructed and linked together, forming an endless stream of memories that can be searched through. However, constructing and linking these data items and subsequently, the boxes presents a significant problem. To address this problem semantic web technologies offer an ideal solution, as it shall be seen in the following section.

2.2 Searching Human Digital Memory Data

The abundance of data that we have access to is phenomenal. The way in which information is generated, searched and accessed is also changing. When the World Wide Web (WWW) was invented in the early 90s (Berners-Lee 1996) it was composed of mainly static read-only HTML, content and provided little user interaction (Chang et al. 2009). This was known as “Web 1.0”. However, over time, user-generated content began to emerge as the new way to produce and access information. The inception of the “Web 2.0” revolution redefined the browser as a vehicle for delivering richer media content and interactivity through a fusion of existing technologies, most notably Asynchronous JavaScript and XML (AJAX) (Chang et al. 2009). This second incarnation of the Web has been called the ‘social Web’, because, in contrast to Web 1.0, its content can be more easily generated and published by users, and the collective intelligence of users encourages more democratic use (Kamel Boulos & Wheeler 2007). Web 2.0 encourages a more human approach to interactivity on the Web and better supports group interaction and fosters a greater sense of community in a potentially ‘cold’ social environment (Kamel Boulos & Wheeler 2007). The development of “Web 2.0” user–centred applications, such as Facebook, YouTube and Flickr (Hotho & Stumme 2010), has allowed the role of the user to shift. Users are no longer an observer of but a participant, in the web. The scope of just how Web 2.0 applications have influenced our lives is evident by Facebook’s popularity. Since its launch, in 2004, 140.3 billion friend connections have been made, 219 billion photos upload (excluding deleted photos) and on September 14th 2012, it had one billion monthly active users (Facebook.com 2012). The sophistication of mobile devices, and their presence within daily life, provides the perfect platform to distribute these types of applications, and bring these two areas together. Hotho and Stumme (Hotho & Stumme 2010) reiterate this point by commenting that the amalgamation of Web 2.0 applications and mobiles has led to the development of the “*Ubiquitous Web*”. However, the web is still evolving. The Semantic Web, also known as “Web 3.0”, will simplify human-computer interfaces by attaching machine-readable metadata (information about information) to web content to enable computers to ‘understand’ the actual / intended meanings of this content as they process it (Kamel

Boulos & Wheeler 2007). This is reiterated by Berners-Lee *et al.* (Berners-Lee et al. 2001), who comment that "*Most of the Web's content today is designed for humans to read, not for computer programs to manipulate meaningfully. Computers can adeptly parse web pages for layout and routine processing, but in general, computers have no reliable way to process the semantics*". The Semantic Web brings structure to the meaningful content of Web pages. It is not a separate web but an extension of the current one, in which information is given well-defined meaning, better enabling computers and people to work in cooperation (Berners-Lee et al. 2001).

In terms of human digital memories, applying the principles of the semantic web enables the machine to "understand" our memory related data. This has the potential for the human digital memory applications to know which information to link together to create a human digital memory. The complex and heterogeneous nature of a human digital memory means that the simple ranked retrieval of information is unlikely to support many of the user's information searching tasks (Kelly & Jones 2007). As we move through different environments, and the data sources that we have access to change, so will the types of data we are collecting. In order to successfully search, and link, this diverse set of data together an intelligent method is required.

2.2.1 Linked Data

Linked Data (also known as the Semantic Web) provides a common framework that allows data to be shared and reused across application, enterprise, and community boundaries (W3C 2011b). It enables intelligent search instead of keyword matching, query answering instead of information retrieval, document exchange between departments via ontology mappings, and definitions of views on documents (Fensel et al. 2011). The heart of the Semantic Web lies in linking data together from different sources. The term Linked Data refers to a set of best practices for publishing and connecting structured data on the Web (Bizer, Heath, et al. 2009) and is essential in connecting data across the semantic web (Berners-Lee 2009). Linked Data relies on documents containing data in the RDF format (Bizer, Heath, et al. 2009), a model for describing resources (Miller 1998).

RDF describes data in the form of triples, which consists of a subject, predicate and object. The subject and objects are Uniform Resource Identifier (URI) references, whilst the predicate denotes a relationship. A set of such triples is called an RDF graph. This is illustrated by a node and directed-arc diagram, in which each triple is represented as a node-arc-node link. RDF enables a number of data items to be linked together. This is achieved from the triples, which denote that a relationship, indicated by the predicate, is held between the subject and object of the triple (W3C 2004b). Once the data is in RDF form, the use of URIs for merging and mapping data from different resources facilitates the development of mash-ups (Hendler 2009). The structure of

RDF is interesting as diverse pieces of data can be associated with each other. This is particularly interesting for memories as links between distinctive pieces of information and memories could be set. Periods of our lives could be divided and the relationships between those times seen.

The Linking Open Data community project (W3C 2011a) is the most noticeable example of the implementation of the semantic web. Founded in January 2007, the project's aim is to bootstrap the Web of Linked Data, by identifying existing data sets that are available, converting them to RDF, and publishing them on the Web (Bizer, Heath, et al. 2009; Bizer 2009). The data sets are distributed as RDF and RDF links are set between data items from different data sources (W3C 2011a). This project has been incredibly successful. As of September 2011, there were, collectively, 295 data sets, consisting of over 31 billion RDF triples, interlinked by approximately 504 million RDF links (W3C 2011a). Figure 2, below, illustrates the datasets (as of September 2011) that have been published in the Linked Data format, by contributors to the Linking Open Data community project and other individuals and organisations (Cyganiak & Jentzsch 2011). The arcs indicate links between items in two data sets, whilst heavier arcs correspond to a greater number of links, and bidirectional arcs indicate that each data set contains outward links to the other (Bizer 2009).

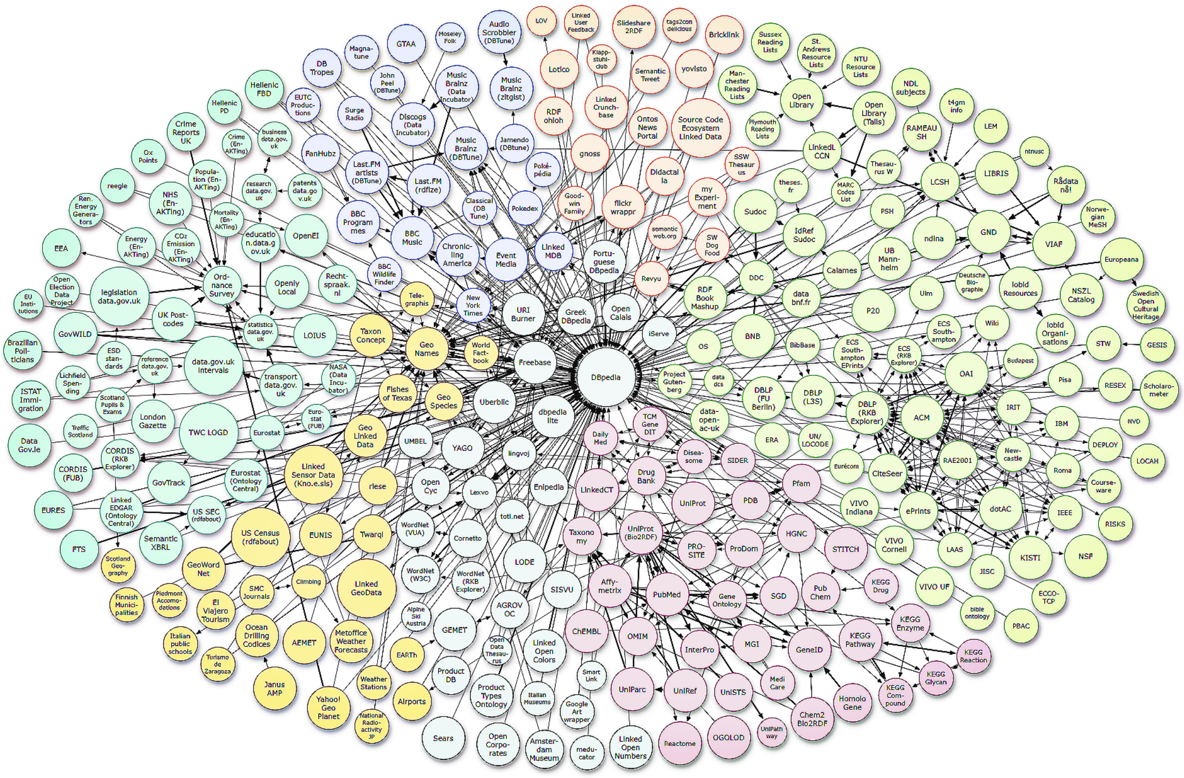


Figure 2 Linking Open Data Cloud Diagram

[Permission to reproduce Figure 2 has been granted by (Cyganiak & Jentzsch 2011) under a CC-BY-SA licence (http://lod-cloud.net/)]

The content of the cloud is diverse in nature, comprising data about geographic locations, people, companies, books, scientific publications, films, music, television and radio programmes, genes, proteins, drugs and clinical trials, online communities, statistical data, census results, and reviews (Bizer, Heath, et al. 2009).

One such application that has come out of the Linking Open Data community project has been *DBpedia* (Auer et al. 2007). This project "*focuses on the task of converting Wikipedia content into structured knowledge, such that Semantic Web techniques can be employed against it*". *DBpedia* has been very successful, with 4.7 billion interlinked RDF triples (Bizer, Lehmann, et al. 2009). This project has also been extended with the implementation of *DBpedia Mobile* (Becker & Bizer 2008). The mobile version "*allows users to access information about DBpedia resources located in their physical vicinity, from where they can explore links to other resources on the Semantic Web*" (Becker & Bizer 2008). This work is of particular interest because of its success in linking data from varied resources together and that data is presented that is in the same proximity as the user.

Taking a different approach, Tummarello *et al.*'s (Tummarello et al. 2010) implementation, *Sig.ma*, uses a holistic approach in which large scale semantic web indexing, logic reasoning, data aggregation heuristics, ad-hoc ontology consolidation, external services and responsive user interaction all play together to create rich entity descriptions. This work is of particular interest due to its focus on combining Semantic Web querying, rules, machine learning and user interaction to effectively operate in real-world Semantic Web data conditions (Tummarello et al. 2010). The system "*Proposes to aggregate heterogeneous data gathered on the Web of Data into a single entity profile using Semantic Web data consolidation techniques*" (Tummarello et al. 2010). However, the weakness of this system is that keyword or structured queries are still needed to search the data and display an entity profile. Data can also be found by following hyperlinks from one profile to another, by accessing permalinks to other profiles or by viewing a web page where embedded JavaScript tags are linked to profiles, via a permalink (Tummarello et al. 2010).

Meanwhile, Smith *et al*.'s work focuses on using open linked data to annotate the life logs of a user (Smith et al. 2011). Their project, *Imouto*, collects and organises lifelogs and life annotations, as well as combining them with both external contextual data and open linked data from the web (Smith et al. 2011). This work is of particular interest due to the similar nature of present research. However, current research aims to improve on this idea by using linked data to connect data from distributed data sources together, instead of obtaining the information from the Internet and to provide intelligent searching of this data.

In comparison, Araújo and Houben’s (Araujo & Houben 2010) work takes the idea of building digital memories in a different direction, by envisioning the web as a “*living system*”. The use of Semantic Web technology and Information Extraction is used for weaving the users’ personal data into a web of concepts. This work is closely related to current research, however, it is only limited to linking data from the web and from online social communities. For a true representation of a memory to be built, data also needs to be incorporated from real–world devices, for example, from sensors. Current research is focussing on the idea of building a memory that is composed of linked data from a number of distributed devices.

In contrast, *SPITFIRE* (Pfisterer et al. 2011), takes the idea of the semantic web further. In this work, Pfisterer *et al.* (Pfisterer et al. 2011) describe their vision of a “*Semantic Web of Things*”. This idea is focused on integrating Internet–connected sensors into the Semantic Web (Pfisterer et al. 2011). Their work, “*Provides abstractions for things, fundamental services for search and annotation, as well as integrating sensors and things into the linked open data cloud*” (Pfisterer et al. 2011). In this context “things” refers to real-world entities, such as meeting rooms and parking spots (Pfisterer et al. 2011). This work is of significant importance because, when building a memory, incorporating as much data from the surrounding environment and smart objects, is vital. Sensor data tends to be ambiguous; therefore overcoming this challenge is a big step into integrating data from a variety of sources into human digital memories. Incorporating this type of data would produce richer human digital memories, as the level of detail is increased.

The benefit of RDF and Linked Data is its ability to represent any object as a triple. With these standardised tools, billions of objects can be represented and linked. One such example is The Linking Open Data community project, which has collected over 31 billion RDF triples, interlinked by approximately 504 million RDF links (W3C 2011a). As everyday objects become connected to the web (smart objects), data is being generated at a faster rate than ever before. As the rate of data that we are producing increases, this standardised format enables user-generated content to become part of the Semantic Web. A consequence of this is that, as more data becomes available, the repositories, which house the information, will become increasingly harder to search. SPARQL is a complex language, and if the queries are not constructed precisely then false results can occur (Hadi et al. 2013). However, a probabilistic approach would allow us to treat the searching of RDF documents as a machine learning problem. This approach enables similar pieces of data to be grouped together, thus creating smaller subsets of data that can then be processed to extract relevant pieces of data. This has an enormous potential to transform the range of data that can be incorporated into a human digital memory. Nevertheless, whilst many systems can capture information, and create human digital memories, and there are many approaches that are used within the Semantic Web, very little work exists that brings together these two areas. In order for a better representation of our lives to be digitally encapsulated enabling these separate disciplines to work simultaneously is fundamental, as a diverse set of data can be searched and linked. As the amount of data that we collect increases searching this amount of information and creating human digital memories still poses a great challenge.

2.2.2 Supervised Learning

As previously stated, the amount of information that is being created, every day, is growing rapidly. This increase in both the volume and the variety of data requires advances in methodology, which are used to automatically understand, process, and summarize the data (Jain 2010). The use of computer algorithms, and visualization techniques, are considered fundamental to support the analysis of such datasets, commonly referred to as *Big Data* (Trelles et al. 2011). One approach is the use of supervised learning algorithms (classification). Using labelled data, this approach is concerned with predictive modelling, (i.e. given some training data, we want to predict the behaviour of the unseen test data) (Jain 2010).

One such implementation has been the *data mining by evolutionary learning (DMEL)* algorithm. This approach, as stated by Au *et al.* (Au et al. 2003), "*Is capable of mining rules in large databases without any need for user-defined thresholds or mapping of quantitative into binary attributes*". This algorithm has been evaluated against several real-world databases and compared against *Decision Tree Learning* (*C4.5*) (Quinlan 1996), *Simple Classifier System* (*SCS*) (Goldberg 1989) and *Genetic Algorithm Based Learning* (*GABL*) (De Jong et al. 1993). The accuracy for *DMEL*, averaged over ten trials for each experiment, was far better than those of the other algorithms.

In other works, the *Apriori* algorithm (Agrawal & Srikant 1994) is, "*A seminal algorithm for finding frequent itemsets using candidate generation. It is characterized as a level-wise complete search algorithm using anti-monotonicity of itemsets*" (Wu et al. 2007). As the algorithm is quite simple and easy to implement, experimenting with *Apriori*-like algorithms is the first thing that data miners try to do (Wu et al. 2007). One approach, which takes this idea further, is Morishita and Sese's (Morishita & Sese 2000) *AprioriSMP* algorithm. This method uses a combination of the *Apriori* algorithm with pruning, via statistical metrics. The advantage of this solution is "*its avoidance of the use of higher support thresholds that has been believed to be requisite for the application of Apriori*" (Morishita & Sese 2000).

More recently, such techniques have been used extensively within the medical domain. One example of this is the *Common Spatial Patterns (CSP)* algorithm. This was proposed by Woon *et al.* and has been successfully used to study Alzheimer's (Woon et al. 2007). In other studies, Latchoumane *et al.*, analyse the brain's electrical activity (EEG signals), using *Multi-way Array Decomposition (MAD)*. This is a supervised learning process for evaluating multidimensional and multivariate data, like EEG (Latchoumane et al. 2012). *Multi-Layer Perceptrons (MLP)* (Pal & Mitra 1992) and *Probabilistic Neural Networks (PNNs)* (Holmes et al. 2001) have featured widely in research to process and analyse such medical datasets. *MLP*s are feed-forward networks that work with back-propagation learning rules. *PNNs* are similar to *MLP*s, in this way, and consist of three layers; an input layer, radial basis layer, and a competitive layer. This type of feed-forward network operates using the *Parzen's Probabilistic Density Function (PDF)* (Parzen 1962). In terms of overall performance, *PNN* networks perform slightly better than *Perfectly Matched Layer* (*PML*) networks (Bakar et al. 2012).

The primary goal of such algorithms is to extract meaning from potentially huge amounts of data. Features, associated with particular data, such as datasets that contain data about neurodegenerative diseases, are characterized. This has led to a great deal of work in feature extraction, within datasets. One example of this is the *Discrete Cosine Transform (DCT)* (Ahmed et al. 1974) algorithm that decreases the number of features and the computation time when processing signals. *DCT* is used to calculate the trapped zone, under the curve, in special bands (Whitwell et al. 2012). Similar algorithms have been used to predict heart disease using *Decision Trees* (Quinlan 1986), *Naïve Bayes* (Lewis 1998) and *Neural Networks* (Holmes et al. 2001). In Palaniappan and Awang's study (Palaniappan & Awang 2008), the results indicate that, using the Lift Chart to graphically present the results for prediction and non-prediction, the *Naïve Bayes* algorithm predicted more heart disease patients than both the *Neural Network* and *Decision Tree* approaches. Using data, collected from patients suffering with Alzheimer's, Joshi *et al.*, were able to identify the various stages of Alzheimer's. This was achieved using *neural networks*, *multilayer perceptrons*, including the *coactive neuro-fuzzy inference system (CANFIS)* and *Genetic Algorithms* (Sandhya et al. 2010). The results showed that *CANFIS* produced the best classification accuracy result (99.55%), as compared to the *C4.5* decision tree algorithm. Other algorithms, such as *dissimilarity based classification techniques*, have also proven to be very useful. For example, algorithms, such as the *k-nearest neighbour classifier (k-NN)*, and *Linear* and *Quadratic normal density based classifiers*, have been extensively used to classify seismic signals (Orozco et al. 2006). Nonetheless, the results have shown that *Bayesian* (normal density based) classifiers outperform the *k-NN* classifier, when a large number of prototypes are provided.

Activity recognition and classifying behaviour can also be achieved using machine learning algorithms (Lee & Cho 2012; Qiu et al. 2011; Phan 2012). In one such approach, Lee and Cho (Lee & Cho 2012) have developed a multi-modal context-aware system, which collects data from various wearable sensors, including accelerometers, gyroscopes, physiological sensors, and data gloves. The system is able to recognize the user's activities using probabilistic models, which are built by using an evolutionary algorithm (Lee & Cho 2012). The optimal probabilistic models, one context versus-all (OVA) dynamic Bayesian networks (*DBNs*), deal with the uncertain, incomplete, and temporal observations from the sensors (Lee & Cho 2012). The results indicate that different activities can be recognised, using this method. However, whilst activities can be recognised other supporting data is missing, such as photos or location. In similar work, Qiu *et al.* (Qiu et al. 2011) use accelerometer data, from a *SenseCam*, and machine learning tools to automatically identify user activity. In their approach, a Support Vector Machine (*SVM*) is trained to automatically classify accelerometer features into user activities (sitting or standing, driving, walking or lying down) (Qiu et al. 2011). The accuracy of each activity ranged from 90% to 98%. This work illustrates how machine learning, and a wearable accelerometer can be used to identify the activities of a wearer to a very high accuracy (Qiu et al. 2011). Nevertheless, whilst these results are encouraging, the location of these movements is unknown and the devices used are expensive and proprietary in nature.

As more data becomes available for public use, encountering large data sets will be more common. A great deal can be learnt from the research efforts carried out on medical dataset analysis, as dealing with big datasets is not unusual. For example, in pharmocogentics five terabyte (TB) files are often used. However, as big data becomes a part of everyday life, dealing with a file that is this big remains a significant challenge.

2.2.3 Unsupervised Learning

As well as classifying data to predict behaviour, another approach, used in the analysis of big data, is to group data, based on similarities. This is known as unsupervised learning (cluster analysis). In this instance, given a set of unlabelled data, this method aims to discover the natural grouping(s) of a set of patterns, points, or objects (Jain 2010). As stated by Wilkin and Huang (Wilkin & Huang 2007), "*An easy abstraction for clustering data is based on a proximity relationship. Data that are close to each other tend to share some external relationship. This relationship can be established to group the data into clusters.*"

Due to its ease of implementation, simplicity, efficiency, and empirical success, the most popular algorithm for clustering is *K-means* (Jain 2010). It is a simple iterative method that is used to partition n observations into a user-specified number of clusters, k (Wu et al. 2007; Kikhia et al. 2011). As defined by

Kikhia *et al.* (Kikhia et al. 2011), "*This method assigns k initial means randomly, and then it goes over all the observations and assigns them with the nearest mean, which results in having k clusters.*" The data objects are grouped together into "compact" clusters with the assumption that all objects, within one group, are either mutually similar to each other or they are similar with respect to a common representative or centroid (Fischer & Buhmann 2003). A requirement of the algorithm is that the user needs to specify the number of clusters (*K*) that they require. This is the most critical user-specified parameter, with no perfect mathematical criteria (Jain 2010). Therefore, defining *K* can be challenging and may be seen as a drawback (Kikhia et al. 2011), as the best number of clusters can be difficult to distinguish. However, a silhouette plot can overcome this. This graphical display illustrates which objects lie well within their cluster, and which ones are in the incorrect clusters (Rousseeuw 1987). It is very useful for selecting the 'appropriate' number of clusters, as it gives an idea of how well separated the clusters are (Rousseeuw 1987; Forero et al. 2004). In conjunction with the algorithm, the appropriate number of clusters can easily be defined. There have been many implementations of the algorithm, across multiple domains. In one such approach, data has been collected, via social networking sites, and clustered to analyse crowd behaviour (Wakamiya et al. 2011). In this work, geo-tagged information from Twitter[7] has been collected for one month, between a specified latitude and longitude range, around Japan. Using *K-means*, crowd behaviour from 300 urban areas have been analysed (Wakamiya et al. 2011). In this approach, such areas have been successfully categorized, based on the characteristics of the collected data.

Another implementation has been the *Self-Organising Tree Algorithm (SOTA)* (Herrero et al. 2001). This approach is a hierarchical clustering algorithm, which is described as being, "*A growing, self-organizing neural network that expands itself by following the taxonomic relationships that exist among the sequences being classified*" (Dopazo & Carazo 1997). In one such implementation, it has been used to analyse gene expression in DNA array experiments (Herrero et al. 2001). This method while based on the *Self-Organizing Map (SOM)* (Kohonen 2013; Kohonen 1990) algorithm provides several advantages over *SOM* and classical hierarchical clustering methods, which use a bottom-up approach (agglomerative) (Herrero et al. 2001). As stated by Herrero *et al.* (Herrero et al. 2001), "*SOTA has two crucial advantages: the topology is that of a hierarchical tree, and the clustering obtained is proportional to the heterogeneity of the data, instead of to the number of items.*" Furthermore, the binary topology produces a nested structure in which nodes at each level are averages of the items below them (items that can be nodes or in the case of terminal nodes, genes) (Herrero et al. 2001). This

[7] https://twitter.com/

makes it straightforward to compare average patterns of gene expression at different hierarchical levels even for large data sets (Herrero et al. 2001).

Other algorithms, such as *Density-Based Spatial Clustering of Applications with Noise* (*DBSCAN*) and *Ordering Points to Identify the Clustering Structure* (*OPTICS*), have proven to be useful for analysing spatial data (Birant & Kut 2007; McArdle et al. 2012). In one such approach, using *DBSCAN* together with *OPTICS*, location points have been clustered into points of interest, with *SenseCam* images have then been linked to these places (Kikhia et al. 2011). The advantages of using *DBSCAN*, over *K-means*, is that *DBSCAN* is less sensitive to noise and allows clusters of arbitrary shape, whilst providing deterministic results (Kikhia et al. 2011). However, as stated by Birant and Kut (Birant & Kut 2007) a drawback of density-based clustering algorithms, such as *DBSCAN* and *OPTICS*, is that "*They capture only certain kinds of noise points when clusters of different densities exist. Furthermore, they are adequate if the clusters are distant from each other, but not satisfactory when clusters are adjacent to each other.*" Whilst this is a good start, the approach is limited due to the inclusion of only location and photograph data. A memory is composed of more information, such as physiological changes, temperature etc. Clustering techniques need to be able to handle this diverse set of data.

Meanwhile, *PhotoTOC*, proposed by Platt *et al.* (Platt et al. 2003), is "*A browser for personal digital photographs that uses a clustering algorithm to automatically generate a table of contents of a user's personal photograph collection*". In this implementation, time-based clustering has been used to choose one photograph from a cluster, which is the most representative of that cluster. These photographs then provide an overview of the entire collection. While Harada *et al.* (Harada et al. 2004) developed a timeline browser for PDA's that uses a time–based clustering algorithm to organise related photos together. Similarly, Harada *et al.*'s (Harada et al. 2004) algorithm has been based on previous work by Graham *et al.* (Graham et al. 2002) in which their original system uses the recursive way in which photographs are taken, in bursts, and represents this using a tree of clusters where photos are stored only at the leaf nodes (Graham et al. 2002). Whilst these developments are interesting, in terms of organisation and the way in which data is retrieved, a memory is composed of more information than just photographs. As previously stated, more information is needed so that detailed human digital memories can be created. Information such as physiological changes, temperature, location, etc. would provide more context about such captured times.

As the accumulation of data increases, the need to provide efficient ways of analysing this information is evident. As these datasets grow in size, executing specific queries to find information becomes less practical, as this requires a precise knowledge of the data. As these datasets increase in size, it is easy to forget to include pieces of information in such queries. However, by using a probabilistic approach, an exact query match is not required. Data is grouped together based on similarities. Therefore, a precise knowledge of the information is not required. This is an interesting approach, which will be explored further so that dynamic and detailed memory boxes can be created.

2.3 Organising Human Digital Memory Data

Capturing human digital memory data can be an easy and enjoyable activity. Nevertheless, searching and organising this information can take a considerable amount of time and is a task that is often neglected by the user. If data is not structured correctly the risk of it becoming useless and unmanageable is greater. Whittaker *et al.*'s (Whittaker et al. 2009) study on people's ability to retrieve photos, which were over a year old, reinforces the idea that without proper structure, data can be inaccessible. The study concluded that for many people, the ability to collect more digital information is not matched by a similar ability to organize and maintain such information (Whittaker et al. 2009). However, human digital memories are not limited to photographs. The data sources that we have access to are increasing; therefore, lifelogs will not only be composed of photos but of a variety of assorted data. This is reiterated by Teraoka (Teraoka 2011) who state, "*Interactive visual interfaces are essential for exploring heterogeneous data*". It is important to structure this data into events, as this is how episodic memory functions within our own mind (Tulving 1993; Tulving 1984). This is echoed by Doherty and Smeaton (Doherty & Smeaton 2010), who state, "*While there are many technologies, which can be used to generate lifelogs, perhaps the most important of the key dimensions for accessing lifelogs, is to structure our lives into events or happenings, corresponding to the happenings which occur in our lives*". However, this remains a significant challenge as the data within lifelogs increases.

The use of timelines to organise visual data is not a new concept and has been the subject of many projects. Microsoft's *MyLifeBits* (Gemmell et al. 2002; Gemmell et al. 2004) is a "*system for storing a lifetime's worth of media, with a database at its heart*" and is seen as a 21st century interpretation of Bush's original idea (Bush 1945). This scheme addresses a user's need to store all of their personal files more easily, as well as having effortless access to them. Sensor readings are displayed in a graph, with black dots for every picture that was taken (Gemmell et al. 2004). *SenseCam* data is displayed based on time and text searches that have been performed (Gemmell et al. 2002). Fuller *et al.* (Fuller et al. 2008) argue that this is the best way to remember content: "*Standard forms of context data, such as time, date, number of accesses, etc. have proven beneficial in retrieval from various collections*". Doherty *et al.* (Doherty, Moulin, et al. 2011) also utilize the timeline concept in their *SenseCam browser* and structure the information as "events". They have created a technique "*to* *automatically structure and organise SenseCam data and later facilitate quick retrieval of desired events with a lower cognitive load on the user, through returning them fewer candidate events that are more relevant to the information they seek*" (Doherty, Moulin, et al. 2011). Lee *et al.* (Lee et al. 2008) also use the *SenseCam* to capture the user's daily routine and present the images in a timeline format, similar to the approach used in Microsoft's *MyLifeBits* (Gemmell et al. 2002) project.

In a similar approach, Plaisant *et al.* (Plaisant et al. 1996) propose *LifeLines*, which uses the timeline concept to map a user's own history. In a related way, Kumar *et al.* (Kumar et al. 1998) use the same format to visualise historical events. Using timelines has become a very popular way to organize file systems. Picault *et al.* (Picault et al. 2010) also present some interesting ideas on how to structure and arrange a user's personal information so that it is easily accessible and can be more effectively retrieved. Their work focuses on structuring data into a timeline format, as "*recalling a piece of information is easier when the user can remind themselves about events in time and space*" (Picault et al. 2010). In similar work, Gozali *et al.* (Gozali et al. 2012) have "*developed a photo browser called Chaptrs that helps users organize their event photos by automatically grouping photos in each event into smaller groups of photos*". Using *Chaptrs*, they conducted an exploratory study involving 23 college students, with a total of 8,096 personal photos, from 92 events. This study was conducted in order to understand how different spatial organization strategies were used in performing storytelling, photograph search and photograph set interpretation tasks (Gozali et al. 2012). The results indicated that subjects valued the chronological order of the chapter's more than maximizing screen space usage, and that they valued chapter consistency more than the chronological order of the photos (Gozali et al. 2012).

Taking a different approach, the *Affective Diary* (Lindström et al. 2006; Ståhl et al. 2009) system is a digital diary where "*users can scribble their notes, but that also allows for bodily memorabilia to be recorded from body sensors*" (Ståhl et al. 2009). This system has been used to explore the emotional aspect to creating diaries and is designed to support self-reflection (Machajdik et al. 2011). However, a drawback of the system is that the user is required to manually upload any logged data into the application and the data is, "*Presented as somewhat ambiguously shaped and coloured figures mapped out along a timeline*" (Ståhl et al. 2009). Taking an opposing approach, *AffectAura* (McDuff et al. 2012) is "*an emotional prosthetic that allows users to reflect on their emotional states over long periods of time*". This system has been designed as a technology probe to explore the potential reflective power that might be offered by pairing affective data with knowledge of workers' information and data interaction artefacts (McDuff et al. 2012). Data is collected from a variety of devices, including a webcam, Kinect and microphone. Supervised machine learning was then used to develop an affect recognition engine, based on the sensed data. The data is displayed in a timeline format, which captured the ebb and flow of affects, represented by a series of bubbles (McDuff et al. 2012). The study was run over four days, using six participants. The results indicated that users were able to leverage cues from *AffectAura* to construct stories about their days, even after they had forgotten these particular incidents or their related emotional tones (McDuff et al. 2012). In terms of correctly classifying emotional states, an overall accuracy of 68%, across all three states of valence, arousal and engagement has been achieved (McDuff et al. 2012). This system is interesting, due to its focus on the emotional state of the user.

The Centre for Digital Video Processing (DCU 2011) has also done extensive work in relation to searching lifelogs and creating systems to organise human digital memory data (Chen & Jones 2010; O'Hare et al. 2006; Kelly & Jones 2007; O'Hare & Smeaton 2009; Byrne & Jones 2008; Kelly 2007; Lee et al. 2008; Chen & J.F. Jones 2008; Lalanne & van den Hoven 2007; Kelly & Jones 2009; Chen & Jones 2009; Qiu et al. 2011; Wang & Smeaton 2012; Byrne & Jones 2009; Wang & Smeaton 2013). In one such project, *MediAssist* (O'Hare et al. 2006), personal digital photograph collections are organised based on contextual information, such as time and location. This information is then combined with content–based analysis such as face detection and other feature detectors. Time-based queries, corresponding to the users partial recall of the temporal context of a photograph-capturing event, for example, all photos taken in the evening, at the weekend, during the summer, can be used to search the data (O'Hare et al. 2006). A timeline format is also used to display the information and semi-automatic annotation of the data is another feature that is possible. Meanwhile, the centre's *iCLIPS* project (Chen & Jones 2010) uses *SenseCam* data to index every computer activity and *SenseCam* image with time stamps and context data, including location, people, and weather. It then enables these files to be searched by textual content and the context data, such as location, people present, weather conditions and date or time (Chen & Jones 2010). The retrieved data is also presented in a timeline fashion.

Location and photographic data can be easily searched and displayed. However, searching and presenting physiological data, in a manner that can be interpreted, is a difficult challenge, due to the ambiguous nature of the data. Currently, the most popular method is to display this information within a graph. Numerous studies have been undertaken in which physiological data has been captured and displayed in this way (Harle et al. 2011; Gilleade & Fairclough 2010; Dwyer et al. 2009; Almeida et al. 2011; Cavalheri et al. 2011; Dias et al. 2009). However, merging this data with our visual memories poses a greater challenge. Interpreting this information so that its context can be associated with our memories is another difficulty. Nevertheless, Chowdhury *et al.*'s (Roy Chowdhury et al. 2010) system, *MediAlly*, is a step closer to solving this problem. *MediAlly* uses a mobile device, to collect and store the subject's contextual states, whilst *Shimmer* sensors (Burns et al. 2010) collect physiological data. The system then produces a metadata stream that describes the contextual origins surrounding the physiological data collection (Roy Chowdhury et al. 2010). Nonetheless, the devices only record location data, which is quite limited. The proposed system aims to take this concept further by incorporating a variety of data sources, as memories are comprised of so much more than just location and physiological data.

As the data that we have access to, and are collecting, is growing so is the inability to effectively manage it. Current approaches are not flexible enough and can be considered one–dimensional. Human digital memory data is highly heterogeneous. In order to create vivid human digital memories this data needs to be clustered together so that related pieces of information are brought together. Furthermore, the ability to answer abstract questions and learn about the user has not been seen before. The amount of data that can be collected over a lifetime is phenomenal. If these collections are to realize their potential, we need to focus on new tools that allow participants to filter, evaluate, maintain and share the huge digital collections that they are now accumulating (Whittaker et al. 2009).

2.4 Summary

This chapter has examined the area of human digital memories, specifically focusing on capturing, searching and organising such data. As it can be seen, many methods and systems can be used to capture and create human digital memories. One of the best known devices and projects, in this area, has been Microsoft's *SenseCam* (Hodges et al. 2006) and *MyLifeBits* (Gemmell et al. 2002). *SenseCam* is designed to be worn continuously to capture photos, whilst *MyLifeBits* focused on collecting, organizing and using the memories of one individual, Gordon Bell (Fitzgibbon & Reiter 2005). Whilst the technologies and methods discussed are a useful starting point, within the area of lifelogging, novel solutions are needed that include much richer data capturing capabilities and require a less obtrusive and expensive, approach. Although the *SenseCam* has been hailed as a "*revolutionary pervasive device*" (McCarthy et al. 2007) it has its limitations. One drawback is in relation to the access of data. Images have to be manually downloaded periodically, which can be very time-consuming and mundane for the user. The device is also very expensive. Exploiting the services of devices, already present within our environment, is a far less expensive approach.

Another limitation, of current systems, is that they record only a restricted amount of information, either visual or physiological data. Human memories are not made up of a finite set of criteria. There are many dimensions to a memory. Creating human digital memories should take the same approach. This is a very important aspect to consider and one that is directly comparable to this research, which collects data in a flexible manner depending on the current sources of information that are available at the time. The services that can be accessed are not limited, and the user is not “locked-down” to only capturing a restricted set of data items. To form a better snapshot of our lives various technologies need to work together, so that we can visually recap our experiences and the feelings and changes our bodies were going through when these events were occurring. Another drawback is that physiological data can be ambiguous and can require extensive data analysis. Automatic analysis of this data would have to be performed, in order to discern meaningful information to enhance memories. Subsequently, incorporating environmental factors, such as temperature and humidity, would reduce ambiguity further and add more information to a memory. As previously stated, for example, a higher heart rate, than normal, and an increase in sweat production could be attributed to many things. Presenting only this information, as a memory, is insufficient. However, if it was known that it was a hot day, by incorporating a temperature reading and that a photograph of the user doing physical activity was also obtained, then the context of the physiological data is known. Bringing together data, from separate sources, enables a finer level of detail to be achieved.

A further limitation is that, as the user moves through different environments, their devices have to be adjusted. For instance, recording location information is not feasible indoors, due to the limitations of Global Positioning System (GPS) sensors. GPS is a popular location estimation system for use in outdoor environments. However, it does not work indoors because it uses signals from GPS satellites, which are blocked when the user is inside a building (Sugano et al. 2006). Any system that uses this method of data collection would be rendered useless in an indoor environment. As the user moves between outside and inside settings, the devices and services that are accessible change. These devices shape the memories. Devices present in one environment will differ to those of another, thus altering the information that is available. In this sense, the memories that are being created will never be the same; therefore, as we move through different environments this will be reflected in the memories that are created. Furthermore, a key concern in human digital memory research relates to the amount of data that is generated, and stored. In particular, searching this set of big data is a key challenge. The data that is being produced and stored is growing daily. Human digital memory systems need to be able to cope with searching this vast pool of information so that relevant information can be extracted and visualised. Machine learning algorithms are seen as a way to alleviate this problem, as large datasets are capable of being processed.

The following chapter details the design of the DigMem system. DigMem addresses the research challenges that have been identified, through examining the area, and addresses the formal challenges identified in section 1.3 Research Challenges. This system brings together and encompasses various data sources, for the purpose of exploiting their device-specific services. Once data is collected, it is added to the human digital memory data space and can be searched in a variety of ways. Using machine learning algorithms, life questions can be answered about our data or this information can be clustered and visualised as a memory box. Information is found based on similarity, and not by defining specific keywords or complicated queries. A vivid interactive snapshot of our lives can be captured, reasoned upon and searched through.

Chapter 3

System Design

In the previous chapter, the background of the area of lifelogging has been discussed. The chapter explored how human digital memory data is captured, searched and organised. In surveying the area, it can be seen that several technologies can be used to capture a variety of data. These include, *SenseCam* for automatically capturing photos (Hodges et al. 2006), the *SWA* for calculating physiological signals (Dwyer et al. 2009), and *activPAL* for detecting body positions (Taraldsen et al. 2011). These technologies are used to capture data about a user's activities and to access information about the physical characteristics of people. In addition, mobile devices are also an important vehicle for capturing data and for disseminating information, with several applications being developed in this area. These include the *Mobile Social Software (MoSoSo)* application for resource sharing (Tsai et al. 2009) and the *JMobiPeer* middleware for creating a mobile P2P platform; that's interoperable with JXTA (Bisignano et al. 2005). However, whilst such systems have been successful, each platform is proprietary in nature. Nevertheless, it is appropriate to build on these advances.

In this chapter, the design of the system, DigMem is presented. DigMem is composed of three components – Mobile DigMem (MoDM), the DigMem Server and DigMem Web application (see Figure 3). This chapter details the design specification and architecture of the entire system before illustrating the design of each of the three aspects of the system.

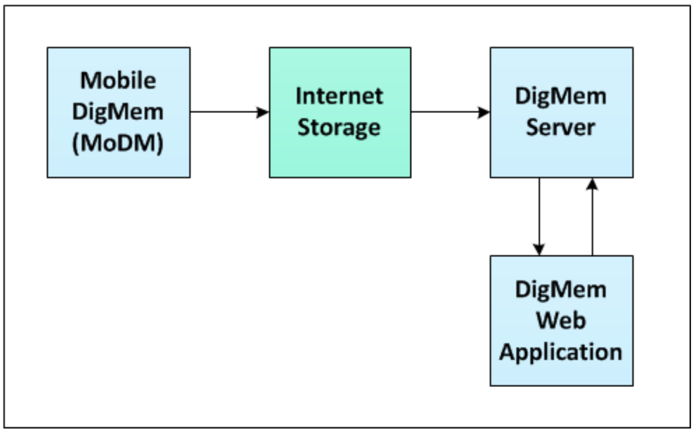


Figure 3 High Level Design of DigMem

Using this approach, the system creates dynamic and interactive memory boxes and to also enable the user to answer questions about their life. Data is obtained through the use of the MoDM middleware (Dobbins, Merabti, et al. 2012b). This middleware discovers various services, either locally or by connecting to pervasive devices, within the P2P network, that are able to offer the use of their services. The data obtained is semantically linked and searched, using the DigMem Server, and then visually depicted as a memory box, within the DigMem web application. In this way, a variety of information can be queried and brought together, to form a snapshot of a particular time.

3.1 Design Specification

Creating human digital memories, using pervasive devices from the user's environment, has the potential to revolutionise how lifelogging is performed. Specialist equipment would no longer be needed, and a more personalized human digital memory can be created. This system creates memory boxes that are composed of a variety of data, from distributed sources. These human digital memories contain vivid structures, and varied information sources emerge, through the semantic clustering of content and other memories. By combining and linking information together, from various devices, a memory composed of a "mash up" of information is created and a greater level of detail achieved. In order to do this, an open, extensible and fully functional solution is needed. The principal requirements of the DigMem system are as follows:

1. To use an open, cost-effective and extendable platform, capable of obtaining a wide range of human digital memory data, as unobtrusively as possible. This requirement focuses on widening the range of data that is available to the user so that they are not restricted to collecting data on a proprietary device (such as only collecting photos using a *SenseCam*). For instance, a mobile phone cannot take the temperature reading of a room; however, by connecting to the thermostat this information could be retrieved and sent back, to enhance the memory; the range of information that the user has access to has now increased. Furthermore, remembering to wear devices every day can become quite arduous. Instead utilizing devices in our environment requires no user involvement. Memory data can be collected without us actively requesting this information.

2. Develop a framework that is flexible and adaptable so that data can be collected in a number of environments, which will be reflective in our memories. Different environments offer a diverse range of services. For instance, in the home data from smart fridges, televisions, game consoles and furniture

could be used to build a memory. However, data from a coffee shop, such as photos, location, building information and friends could be used to document this moment. The framework needs to be able to adapt to these situations so that a plug-and-play platform for memory data sources that can be exploited by any digital life memory middleware service.

3. Provide a method that is capable of using collected data over an extensive period of time. Human digital memory systems aim to record our entire live. However, as technology progresses new devices and file formats emerge. Such systems need to be able to cope with these changes so that previously collected data can still be retrieved and viewed. As such using a metadata model ensures that this information can still be incorporated into a memory, even if the underlying device has become obsolete. The generic structure of such models makes data interoperability and evolution easier to handle.

4. Enable large amounts of data to be efficiently searched so that memory boxes can be created. Collecting data over a lifetime produces a phenomenal amount of informing. An efficient method is required that can handle searching large amounts of data without the need for user-defined queries. Machine learning algorithms are seen as a way to execute this so that information can be gathered based on a set of pre-defined criteria, such as "When have I been active?" The retrieved data is then transformed into a visual representation and displayed as a memory box.

5. As well as creating memory boxes, the user should also be able to ask questions about their data. In this instance, a separate set of pre-defined criteria is used, such as "Does this location make me happy?" Using a different set of machine learning algorithms to requirement three, the system can learn about our data to answer such questions.

The above system requirements have been formulated to address the limitations of current methods and to address the challenges identified in section 1.3 Research Challenges (pages 7-9). In this sense, a wider range of data can be included in our memories. Information can be efficiently searched so that memory boxes can be created, and life questions answered. Furthermore, collected human digital memory data can also be utilised over an extensive period of time. As we have seen, specialist equipment (for example, *SenseCam*) is needed in order to capture continuous information about the user. However, this approach is costly, proprietary and requires the user to manually upload their captured images/data each day. In addition, only photos and a small number of sensor readings are recorded. Human digital memories are composed of much more information than this. The DigMem system aims to address these issues by building on the nomadic nature of people so that human digital memories are reflective of the user's current environment. These memories are unique and are not "tied-down" to only featuring a limited amount of information. However, it is still appropriate to build on these existing technologies to address these limitations.

3.2 System Architecture

As stated above, the DigMem system is composed of three components, MoDM, the DigMem Server and DigMem web application. Figure 4 illustrates the design of the system, and below an overview of the system is presented, before describing the design of each component of DigMem.

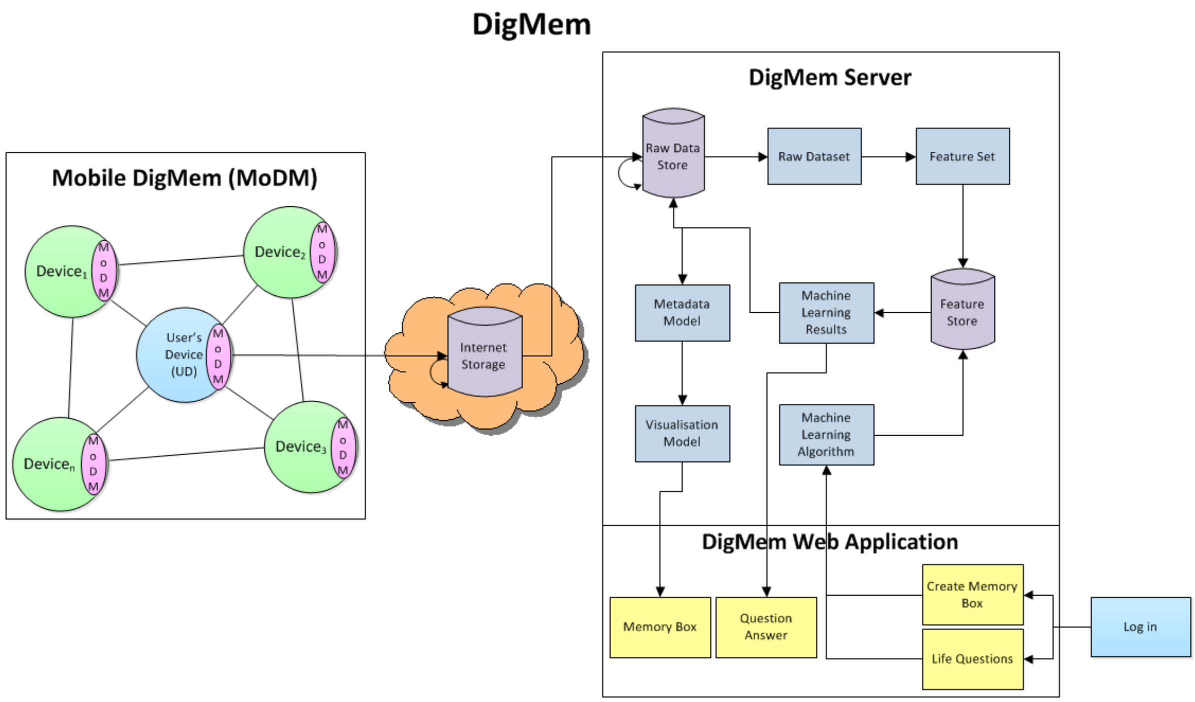


Figure 4 DigMem System Design

In order to start collecting data, all MoDM (Dobbins, Merabti, et al. 2012b) compliant devices connect and advertise, to the P2P network, the services that they can offer to other devices. These devices then wait for a connection from the user. As the user enters the environment, they launch MoDM, on their mobile device and connect to the same network. Once connected, the MoDM interface displays all the services that DigMem supports. Upon selection of a service(s), firstly, the user's device (UD) is searched, to find all local services. If a local service has been found it is started and data collection begins. However, if particular services cannot be found, on the UD, then a remote search is executed. In this instance, a request is sent out to all neighbouring waiting peers, within the network, in order to obtain the use of their services. Once a service has been found, i.e. a device in the network has that particular service, the responding device, connects to the UD and sends their data, via bidirectional pipes. After the user has selected the service(s) that they want, the application continues to collect data, without any more user intervention.

P2P was chosen, in contrast to a client-server model, for a variety of reasons. Most notably because it is a scalable solution, as more peers join the network, its capability increases and strengthens (Krishnan 2001). Furthermore, as peers exit the network a peer's ability to exchange information is not reduced. If one peer is unavailable, then another with similar capabilities will often be able to provide the same information or service. This fits in well with creating human digital memories, as memories created in various environments will require the use of different services. Being tied-down to a set number of services does not fit in with the diverse composition of memories. Human digital memories are not composed of a fixed number of items, as we move through different environments the requirements of a human digital memory will change. This idea does not suit a client-server system. As Parameswaran *et al.* (Parameswaran et al. 2001) states, "*A client-server scenario like the Web depends on a single server storing information and distributing it to clients in response to their requests. The information repository remains essentially static, centralized at the server, and subject only to updates by the provider. Users assume a passive role in that they receive, but do not contribute, information. A P2P network, on the other hand, considers all nodes equal in their capacity for sharing information with other network members*". The idea of human digital memory does not fit in well with a client-server situation. Users need to be able to gather and share information equally.

On the user's device, a local, Internet-connected, directory stores all information that is collected. This data is then transferred, periodically, from the network, to the Raw Data Store. From this point, the DigMem Server side of the system is utilised. Using the Internet bridges the gap between MoDM and the DigMem Server. The benefit of using this service is that information can be stored across multiple stores, can be accessed anywhere, at any time, and is backed-up regularly. It also eliminates the need for the user to manually upload their data onto the system. By periodically transferring the collected data into this folder also creates free space on the mobile device. This is very important, since the storage capacities of these devices are limited. This automated approach to data collection is also far less intrusive, than previous lifelogging methods, and is unique. Once the MoDM application is started, it collects data, saves this information in an Internet-connected directory, transfers this information into the Raw Data Store and frees up space on the user's device, all without user intervention.

Once information is transferred to the Raw Data Store, periodic extraction and transformation of the raw information, into a Raw Dataset occurs. Using this dataset, features are automatically extracted, analysed and a new Feature Set is formed. This Feature Set is then saved, within the Feature Store.

At this point, the user has two options. Either they can create a memory box, which illustrates their data, or they can choose to ask the system questions about their lives (memory data). In the case of creating a memory box, once logged in, the memory search criteria are defined, which consists of a set of pre-defined questions that the system is able to use to group their data on. For example, “When have I been active?” Upon selecting a question, the retrieval of information occurs by using machine learning algorithms to explore the Feature Store and cluster information based on their chosen question. Within these results, timestamps are used as an index to search the Raw Data Store for the corresponding information. Once the data has been retrieved, it is converted into a metadata model. The reason behind converting the data into this model is for flexibility. This model enables the manipulation, and transformation, of the data, into any other format, for further analysis. For display purposes, the metadata model is then transformed into a visualisation model. The reason behind conversion, into another model, is so that the data can be visually displayed, within the DigMem web application. The model is then loaded into the DigMem web application, and a memory box is created and displays all of the information, as graphical items. This is opposed to just listing the raw information, for example; location coordinates, file locations of photos or physiological readings, which do not hold as much meaning, perhaps as photographs or a map does. However, if the user wants to ask questions about themselves, another set of pre-defined questions are presented, for example, “This place looks familiar to me. Have I been here before?” Again, upon selecting a question, machine learning algorithms are used to explore the Feature Store and answer the question. Based on the results from this question, the answer is displayed to the user, i.e. Yes or No.

Based on the system design from Figure 4 and the above description, Figure 5, below, illustrates an overview of the process that DigMem uses to create memories. The diagram demonstrates how DigMem’s three components interact with each other so that the goal of creating memory boxes or answering life questions can be achieved. This process enables more dynamic and detailed human digital memories to be created, whilst also enabling the system to learn about its user. The use of pervasive devices, linked data, and machine learning are central to the system, and are what make it unique. By incorporating data from local and outside sources, as well as from body sensors, a better understanding of ourselves can be retrieved and reasoned over. DigMem is flexible enough to grow with the user and allows an entire lifetime to be digitally captured and reflected upon.

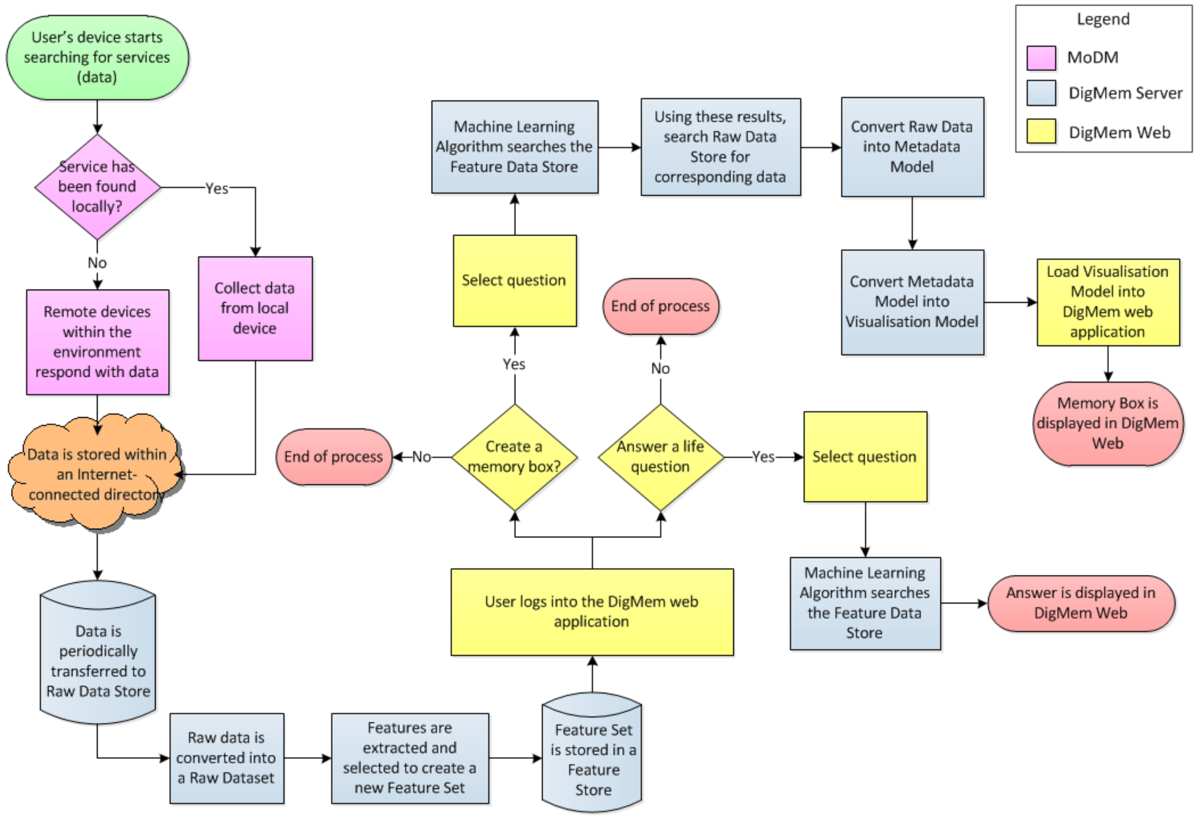


Figure 5 Overview of DigMem

The remainder of the chapter provides a more detailed design of the Mobile DigMem (MoDM), DigMem Server and DigMem Web Application components of the system.

3.2.1 Mobile DigMem (MoDM)

This section details the design of the MoDM middleware (Dobbins, Merabti, et al. 2012b) that has been developed, during the research. There are two categories of devices that are used within this scenario – sensor devices, which provide information and the user's mobile device, which request's information. Sensor devices are only concerned with producing information and sending this to the user. In this context, a sensor device is defined as any object, with an embedded sensor and the user's device is defined as a mobile device (for example, smartphone, tablet or laptop). It should be noted that the user's device would also have its own set of device-specific, local, services, which can be used to collect information.

In order to collect data, for our human digital memories, services can be discovered either locally (on the user's device) or remotely (from sensor devices). Any device that is used to gather, or supply, information needs to implement a core set of components and the MoDM middleware. Since there are two categories of devices, each device category needs to implement a slightly different version of the MoDM application. The sensor peers will need to implement the sensor version of the MoDM application, whilst the user peer runs the user version of the application. Both applications handle connecting to the network, receiving, and processing messages; however, the user's version is composed of an extra component, the user interface. This is so that the user can select what service(s) they want to use. Below is a description of each component and their associated functions.

A. Network Handler

The network handler manages the state of the network. This includes starting the P2P network, creating the peer group, connecting the peers to the peer group and finding peers within the network.

B. User Interface

The user interface, of the user version, consists of a simple menu screen that lists the services that DigMem can support (see Figure 6). Upon selection of a service, a message is sent to other peers to locate that particular service. The responding peers send a reply back, which is processed and the information is saved in the user's Internet-connect folder.

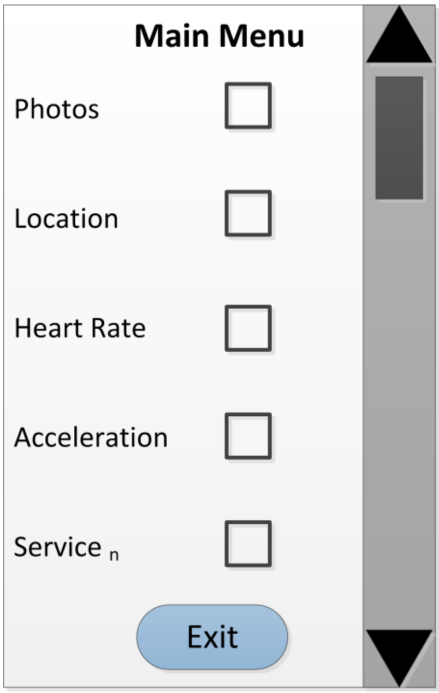


Figure 6 MoDM Interface Design

C. Connection Handler

The connection handler listens for any events that happen along the pipe and processes all incoming messages, for both applications. The sensor application processes the arriving message and extracts the command, so that appropriate action can be taken. For example, if the command "Take a photo" is received the handler extracts the command and starts the camera service. A photograph is sent back, in a response message, to the peer that requested it. The connection handler, within the user's application, does the same job of processing the incoming response. In the case of receiving the photograph, the connection handler gets the message, extracts the photograph and saves it. In summary, sensor devices need to have the following requirements:

1. Connect to the P2P network
2. Publish the availability of their services.
3. Allow the user's device to connect
4. Gather information and send this back to the user's device
5. Conform to accepted DigMem data types

Similarly, the user's device also needs to have the following requirements:

1. Connect to the same P2P network as the sensor devices
2. Search for services (locally and remotely)
3. Connect to sensor devices
4. Gather information from sensor devices.
5. Register with DigMem to receive a unique user ID

As well as these core components, each device also implements its own set of device-specific services. An example of these can include, but are not limited to, a camera service for taking photos, a GPS service for location, an accelerometer service for obtaining acceleration information or a heart rate service for obtaining heartbeat information.

The following sequence (see Figure 7) and activity (see Figure 8) diagrams illustrate the MoDM middleware. These diagrams demonstrate the necessary steps that are taken in order for sensor devices to connect to the network and advertise the use of their services. They also illustrate how the user's device connects to the same network, searches for services, both locally and remotely, and obtains information. The following section provides a full description of this process.

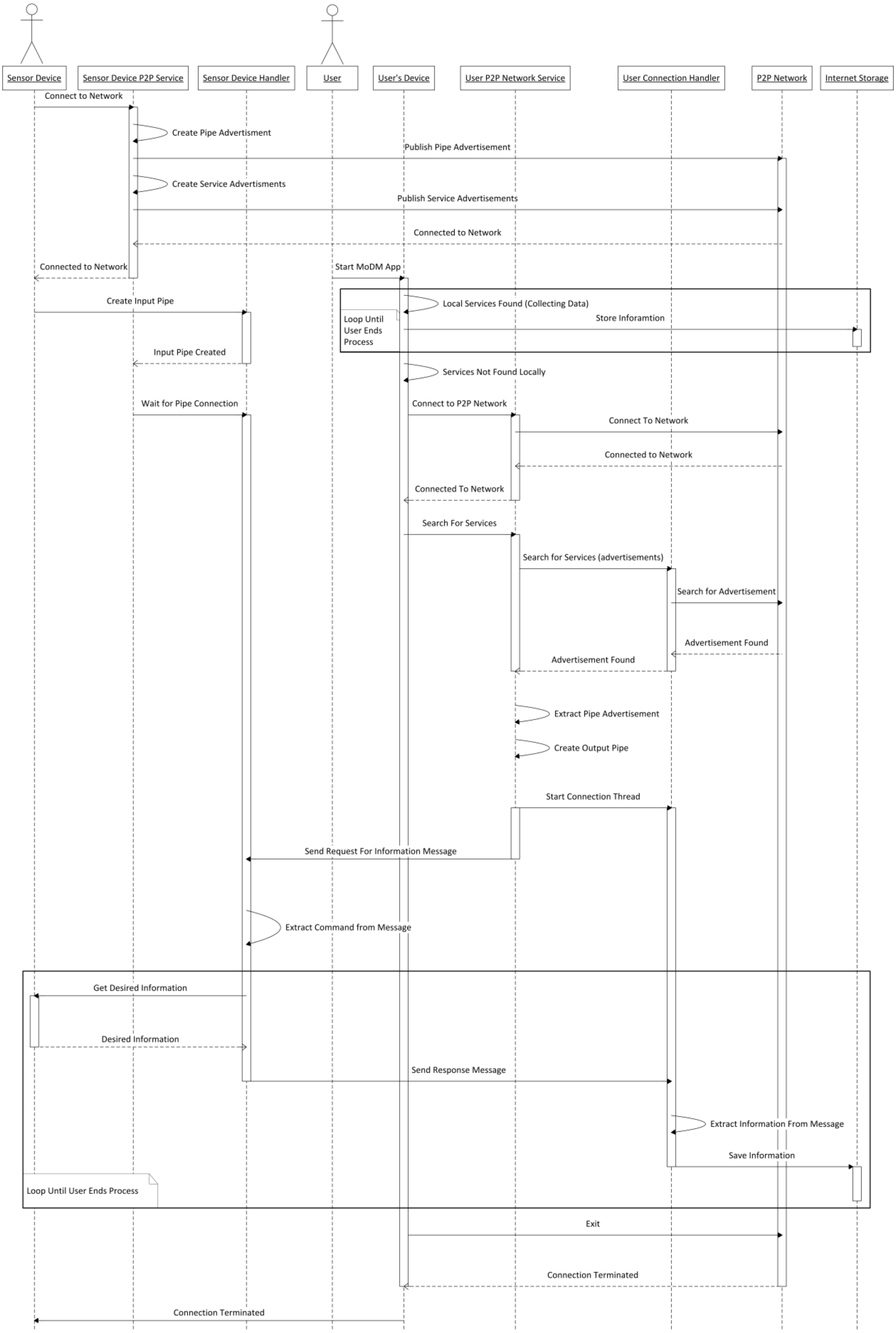


Figure 7 MoDM Sequence Diagram

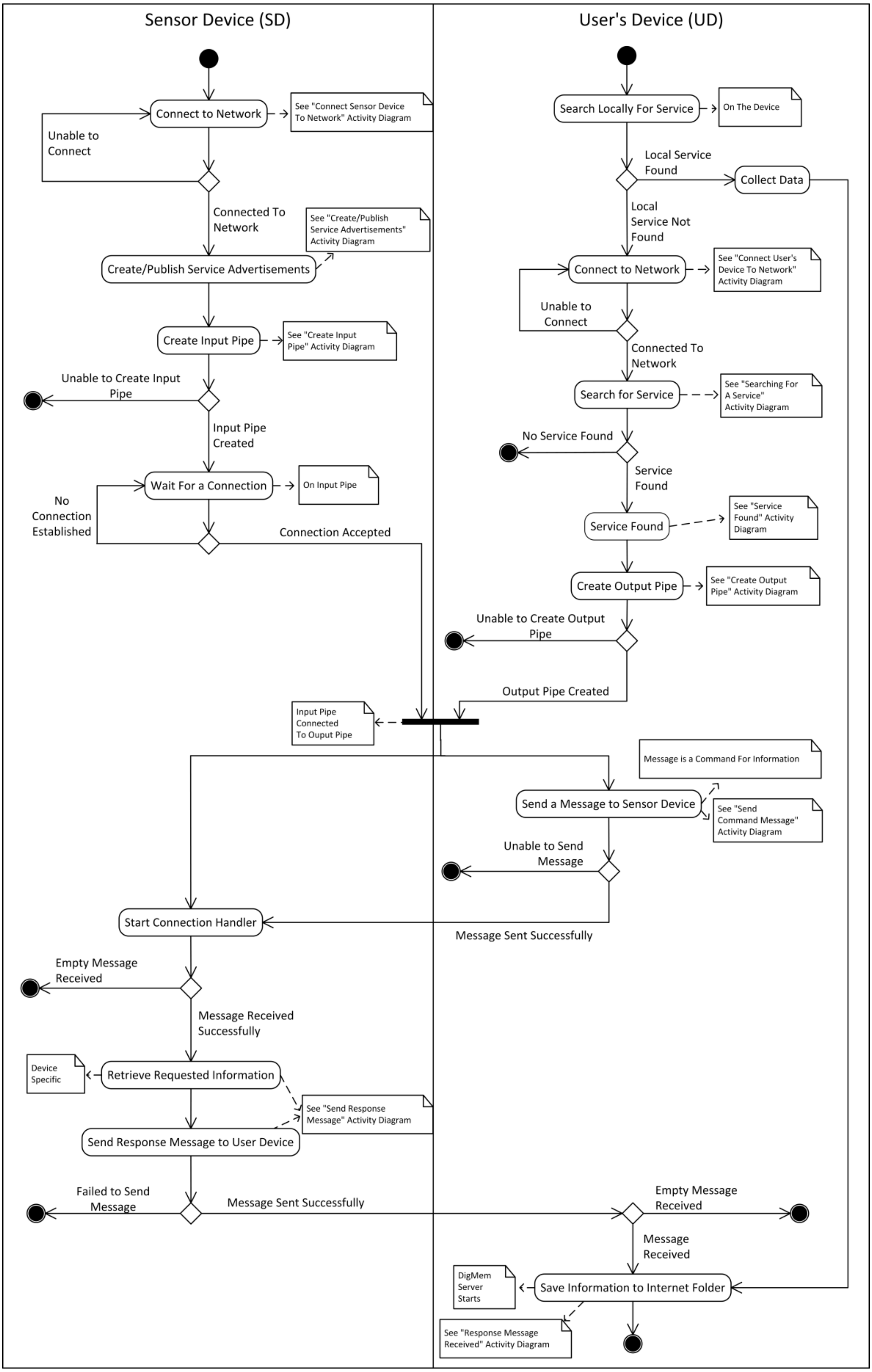


Figure 8 MoDM Activity Diagram

Initially, the sensor device (SD) needs to connect to the P2P network. Figure 9, below, illustrates this procedure.

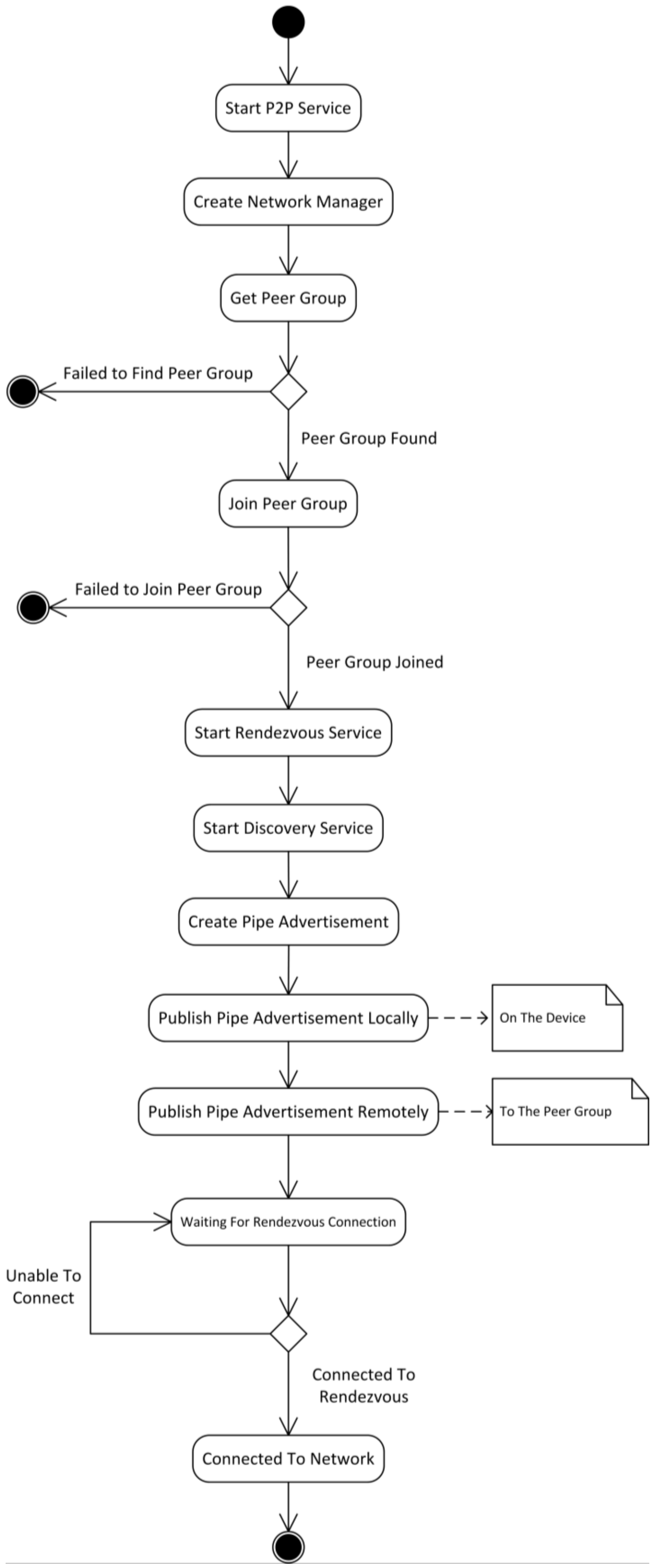


Figure 9 Connect Sensor Device to Network

Firstly, the P2P service is started, which gets the credentials of the P2P network and initiates the process of connecting. In order to join the P2P network a network manager is needed. This component is responsible for getting the configuration of the network, starting it and obtaining and joining the network peer group (Net Peer Group). By default, all peers belong to this group. Once the device is part of the peer group, a rendezvous service is obtained, and the peer attempts to connect to the service. This service is responsible for propagating messages within a peer group (Sun Microsystems Inc. 2003). Meanwhile, a discovery service is also initialised. This service is used to locate any published peer resources, and is the default discovery protocol for the net peer group (Sun Microsystems Inc. 2003). The device then needs to create its pipe advertisement. A pipe advertisement is a XML-structured document that names, describes, and publishes the existence of a pipe, or a service (Gong 2001). Once this has been created it is then published locally, on the device, and remotely, to the peer group. At this point, either the device will still be trying to connect to the rendezvous or a connection has successfully been made.

Once a connection, to the network, has been obtained, the device will then need to create and publish its three service advertisements. Figure 10, below, illustrates this procedure. The first advertisement to be created is the service class advertisement, which contains the service class ID, name and description of the service that the device can offer. This is used to broadcast the existence of a service (Sun Microsystems Inc. 2003). Once this has been created it is then published locally, on the device, and remotely, to the peer group.

Once publication of the service class advertisement has been successful, the creation of the next advertisement (service specification advertisement) commences. This advertisement contains the previously generated service class ID and pipe advertisement. The pipe advertisement is included so that the client is able to contact the service (Sun Microsystems Inc. 2003). Once this has been created it is then published locally, on the device, and remotely, to the peer group.

Once the service specification advertisement has been successfully published, the creation of the next advertisement (service implementation advertisement) commences. This advertisement contains the previously generated service class ID and enables a peer to retrieve the necessary data to execute the implementation (Sun Microsystems Inc. 2003). Once this has been created it is then published locally, on the device, and remotely, to the peer group.

The deployment of these advertisements enables any device to publish the existence of its own device-specific service. These advertisements are used to identify the existence of the service (Service Class), the requirements of the service (Service Specification), or the execution of the service (Service Implementation) (Sun Microsystems Inc. 2003).

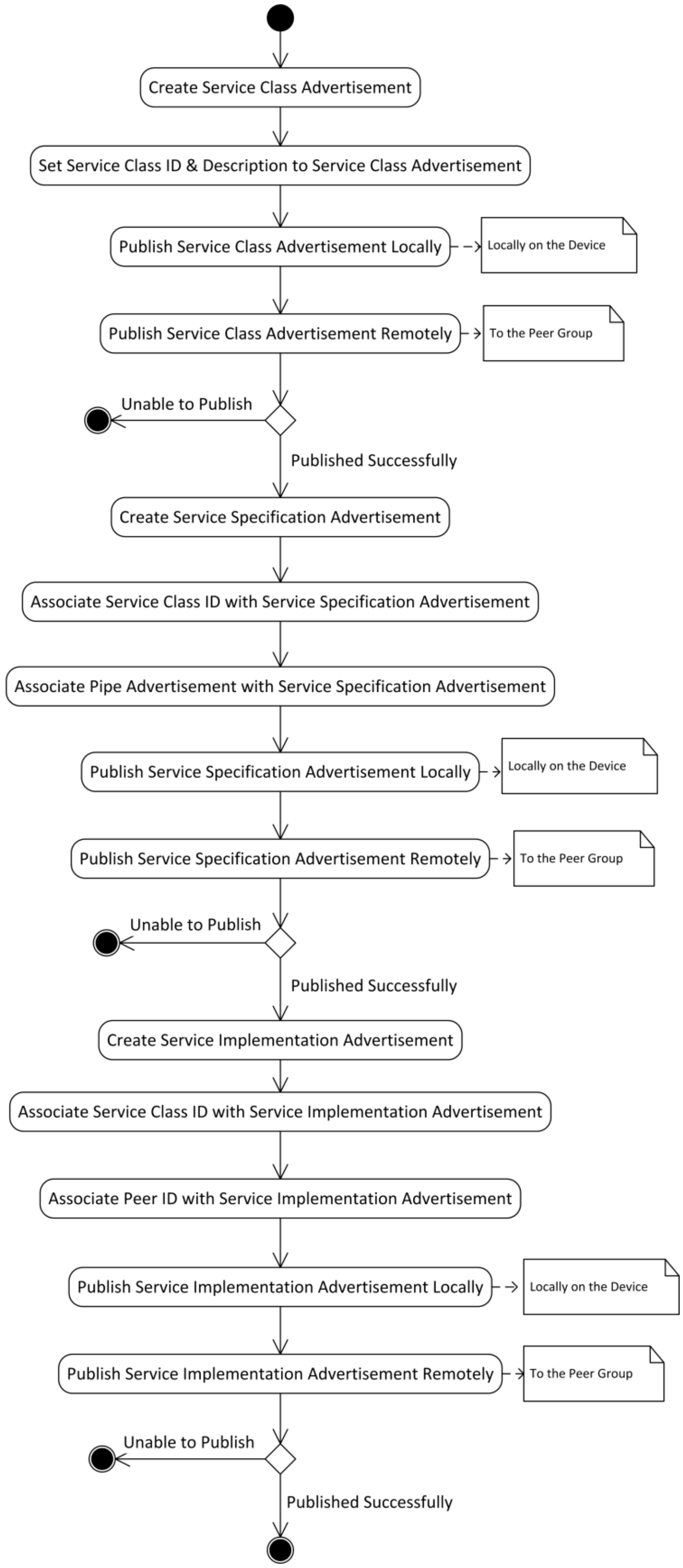


Figure 10 Create and Publish Service Advertisements

Having successfully published all the service advertisements, the creation of the Input Pipe commences. Figure 11, below, illustrates this procedure.

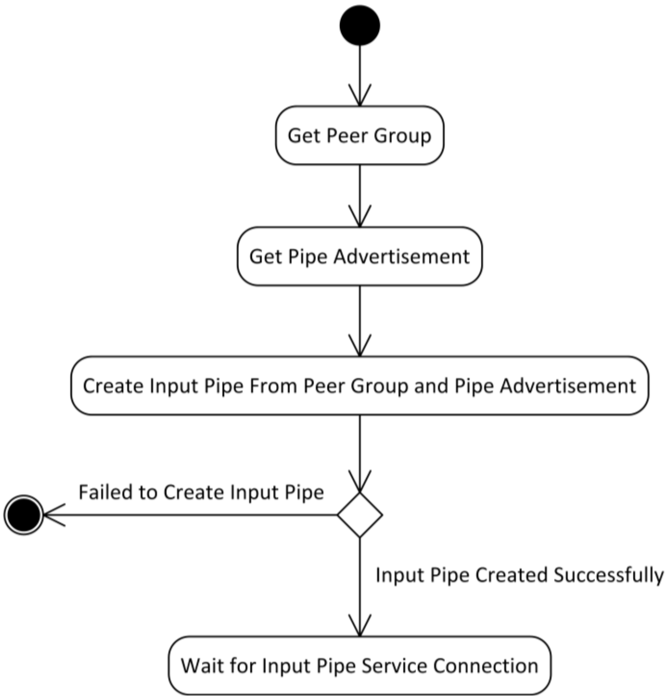


Figure 11 Create Input Pipe

The creation of the Input Pipe occurs by retrieving the Net Peer Group and pipe advertisement. Pipes are the core mechanism for exchanging messages and provide a simple, uni-directional and asynchronous, channel of communication, between two peers (Sun Microsystems Inc. 2003). A device that wants to open communication with other peers creates an input pipe and binds it to a specific pipe advertisement (Sun Microsystems Inc. 2003). The device then publishes the pipe advertisement so that other devices can obtain the advertisement and create corresponding output pipes, in order to send messages to that input pipe (Sun Microsystems Inc. 2003). The PipeId, enclosed in a pipe advertisement, uniquely identifies pipes. This unique ID is used to create the association between input and output pipes (Sun Microsystems Inc. 2003). Once this pipe has been created, it waits for a connection. The pipe endpoints are referred to as the input pipe (the receiving end) and the output pipe (the sending end). Pipe endpoints are dynamically bound to peer endpoints at runtime (Sun Microsystems Inc. 2003).

In the meantime, whilst the sensor device awaits a connection, the User's Device (UD) begins its search for information. Firstly, the UD is searched, to find all local services. If a local service has been found it is started and collects data. The data is saved in the users, local Internet-connected folder. This folder is populated with various pieces of information, from many devices, that the UD connects to. However, if particular services cannot be found, on the UD, then a remote search is executed. In this instance the UD also connects to the network. Figure 12, below, illustrates this procedure. Like the sensor device, the procedure is practically the same, the P2P service starts, followed by the creation of a network manager. This is followed by obtaining and joining the network peer group (Net Peer Group). Once the device is part of the peer group, the rendezvous and discovery services start. At this point, the device will either still be trying to connect to the rendezvous or a connection has successfully been made.

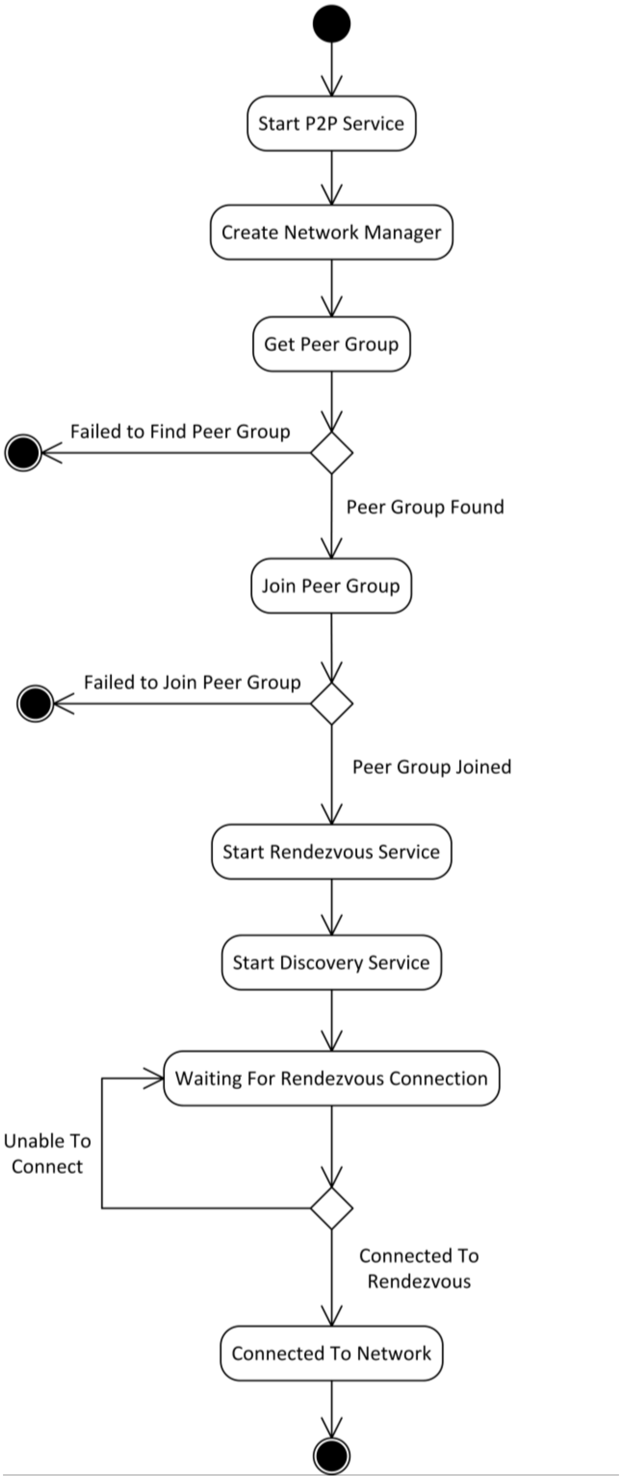


Figure 12 Connect User's Device to Network

The device then needs to search, remotely, for a service. Figure 13, below, illustrates this procedure. After establishing a connection, to the network, a Discovery Listener is required. By making the user's device a Discovery Listener, this interface enables the user's device to asynchronously be notified every time it receives a Discovery Response message (Sun Microsystems Inc. 2003). This is important for later, when the sensor device sends its data. Once we have added ourselves as the listener, a discovery message is sent to all peers, within the peer group. This message contains the name of the advertisement that the user's device is looking for. This loop continues until the service has been found.

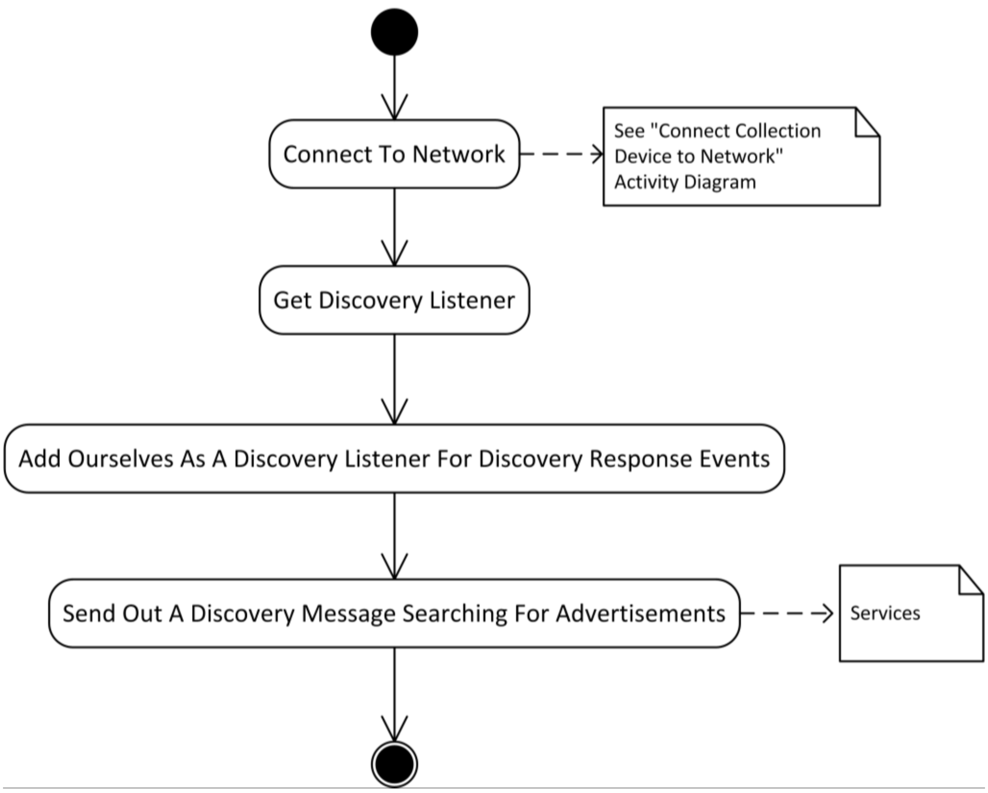


Figure 13 Searching For a Service

Once the service has been located, Figure 14, below, illustrates the next procedure. After the user's device locates the service, it gets the responding previously published advertisements, from the sensor device. The sensor device's pipe advertisement is extracted, from the Service Specification Advertisement. Once extracted successfully, the user's device then creates its output pipe, sends a command message to the sensor device and starts its service connection handler, to receive incoming messages.

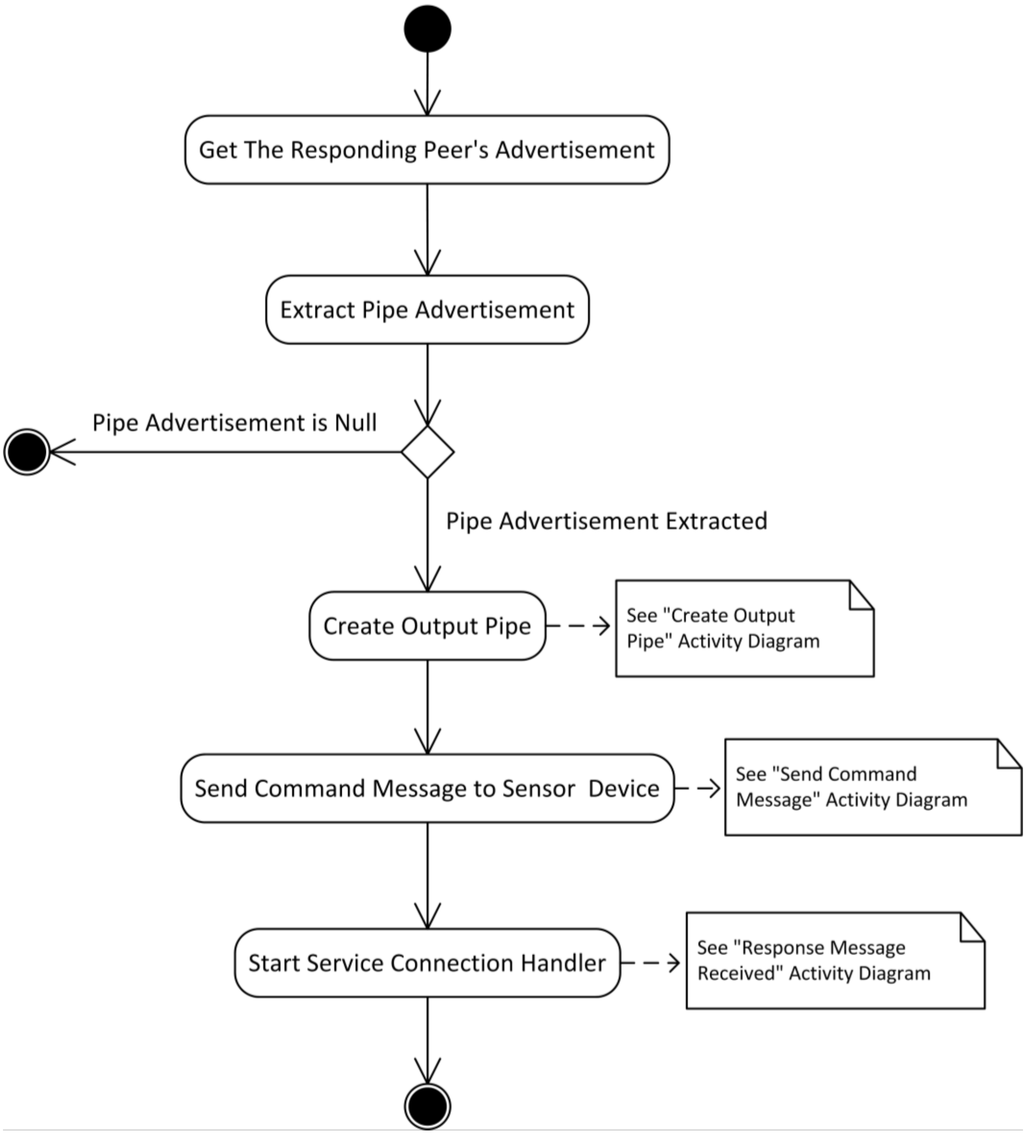


Figure 14 Service Found

The following two activity diagrams illustrate, in more detail, the process of creating the output pipe and sending a command message.

Figure 15, below, illustrates the procedure to create the output pipe. Once the user's device extracts the pipe advertisement, from the sensor device, this is then used, along with the peer group, to create the output pipe, from the user's device. As stated previously, pipes are the core mechanism for exchanging messages and enables devices to communicate with each other. The output pipe, from the user's device, then connects to the waiting input pipe, from the sensor device. Once a connection has been established, the user's device sends a command message to the sensor device, via this new channel of bi-directional pipes.

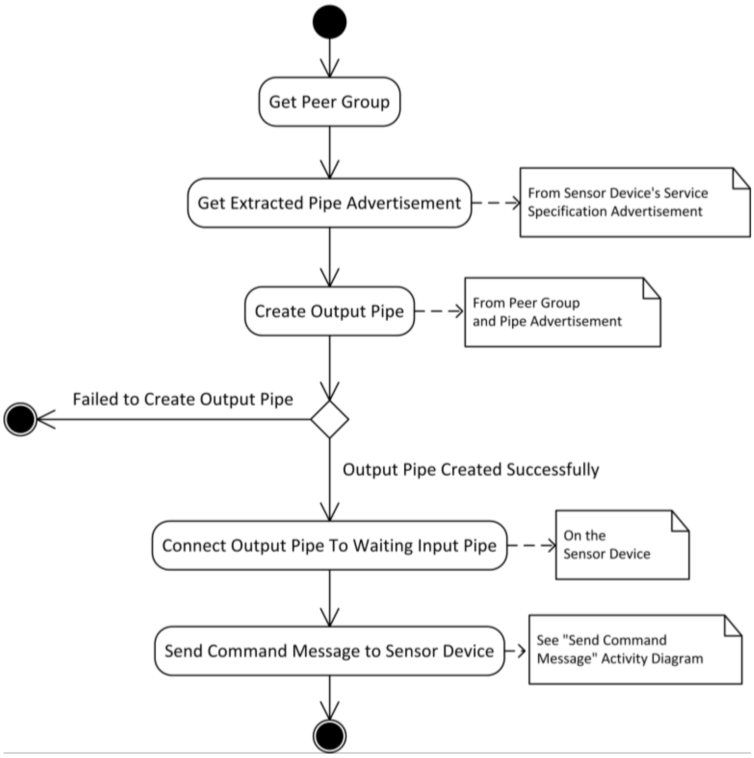


Figure 15 Create Output Pipe

Figure 16, below, illustrates the procedure for the user's device to send a command message to the sensor device. Firstly, a message needs to be created. This is then populated with the message elements. Message elements contain the command for information that is needed for the sensor device to obtain information. For example, if a camera service has been found a command could be "Take Photo". Upon receiving this command, the sensor device would take a photograph and send it to the user's device. After the addition of these elements, using the output pipe, the message is sent to the sensor device.

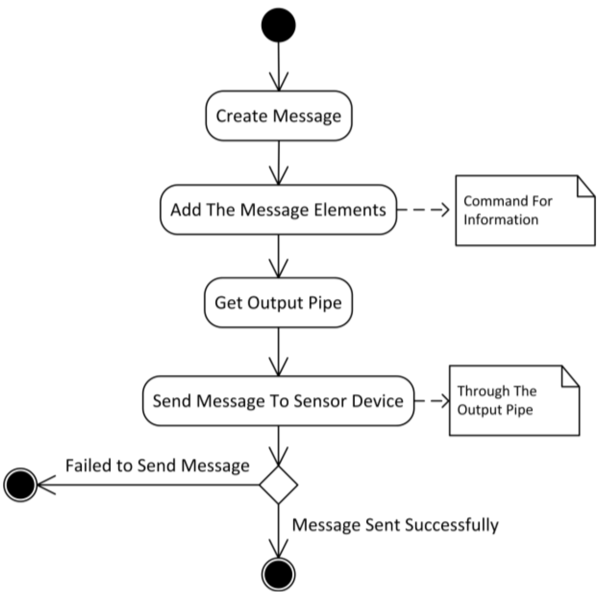


Figure 16 Send Command Message

Figure 17, below, illustrates the procedure that occurs when the sensor device receives the user device's command message. The sensor device starts by getting the message from the pipe. If the message is not empty, then extraction of the command occurs. Upon obtaining the command element, the service starts. This service is device-specific. For instance, in the case of a camera service, the camera would start to take photos. Once the service starts gathering data, a response message is created. The service data is then added to this message. After the message has been populated, the input pipe is obtained, and the message is sent down it, to the user's device. The process of gathering data and sending it to the user's device repeats until the user closes the connection, i.e. stops requesting information.

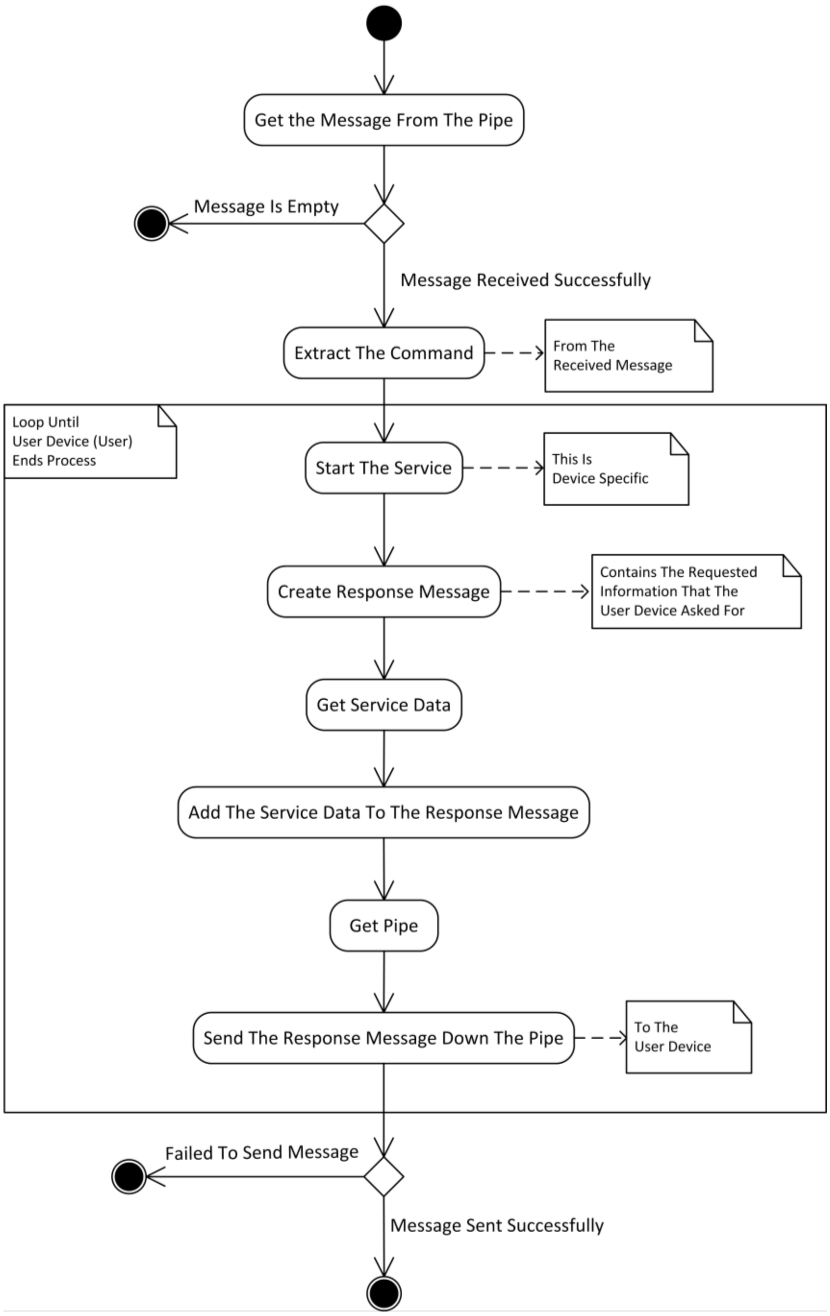


Figure 17 Send Response Message

Figure 18, below, illustrates the procedure that occurs when the user's device receives the sensor device's response message. The user's device starts by getting the message from the pipe. If the message is not empty, then extraction of the data occurs. The data is saved in the users, local Internet-connected folder.

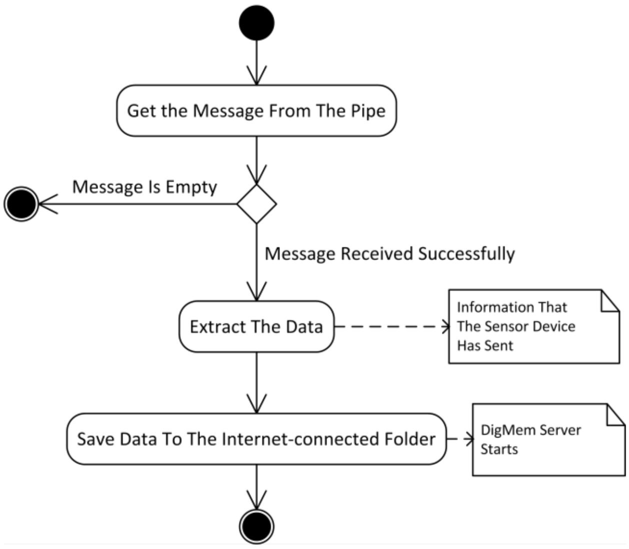


Figure 18 Response Message Received

This section has detailed how the MoDM aspect of the system enables the user to collect information, from ubiquitous devices, prevalent within their environment. The retrieval of this data enables human digital memories to become richer in context. Information can be incorporated into the memory that necessarily would not have been available previously. This platform opens up a completely new dimension into the information that can be integrated into human digital memories (HDM). Users are no longer restricted to only incorporating a minimal amount of information into their HDMs. As advances are made in this area, the amount and types of data that can be collected is also likely to increase.

Once the information has been retrieved, it then needs to be processed, so that the raw data can be transformed into a HDM. The following section details how the DigMem Server achieves this.

3.2.2 DigMem Server

This section details the design of the DigMem Server that has been developed, during the research. The following sequence (see Figure 19) and activity (see Figure 20) diagrams illustrate this aspect of the system. These diagrams demonstrate the necessary steps that are taken in order for the retrieved information (from MoDM) to be transformed into a feature set, which will be used later, by the DigMem web application, to create a memory box or to answer life questions. The following section provides a full description of this process.

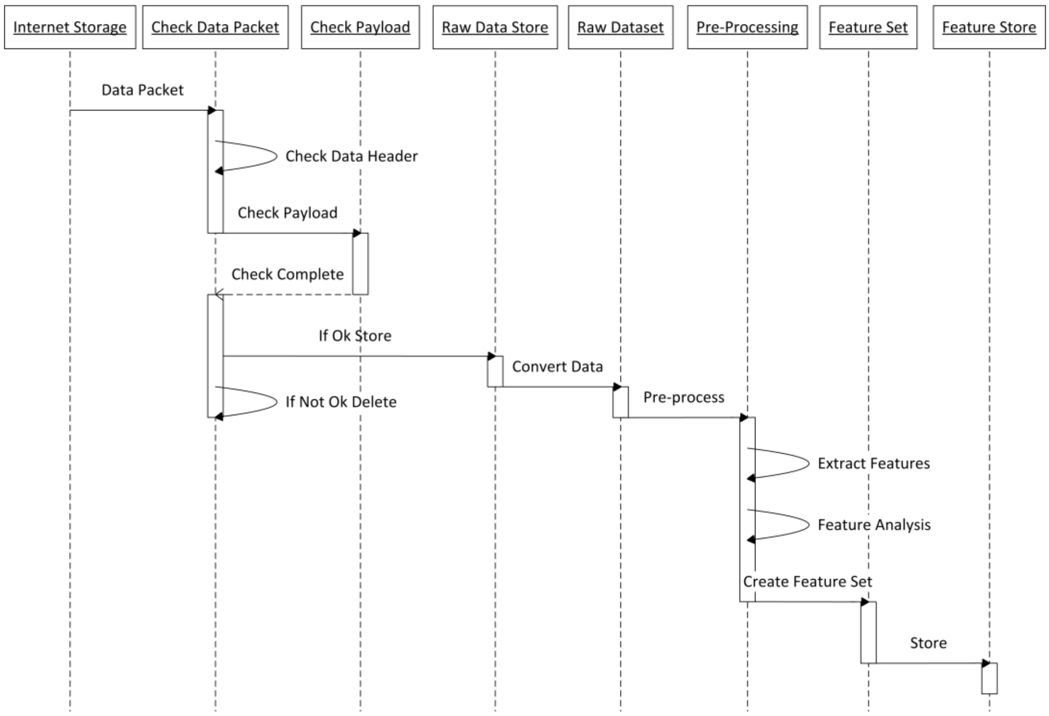


Figure 19 DigMem Server Sequence Diagram

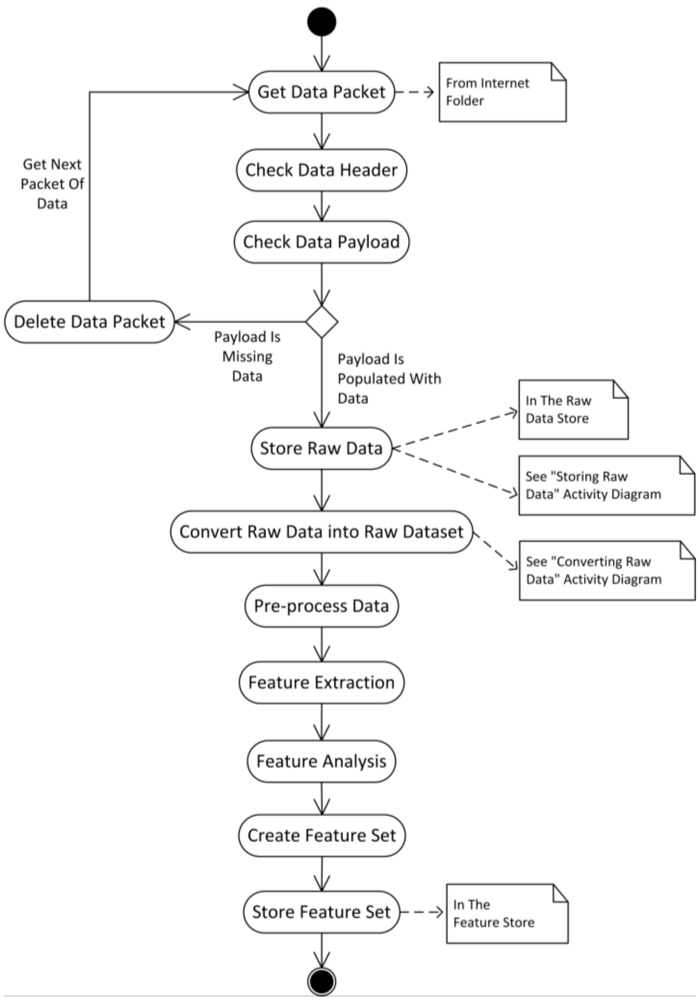


Figure 20 DigMem Server Activity Diagram

Once data is collected, it is stored, within the user's Internet-connected directory. Extraction then occurs, periodically, so that this raw information is saved, within the Raw Data Store. Within this data store, there are many tables. These tables store details of the registered users of the system, and correspond to the data that DigMem can collect and process. Initially, when the user registers with DigMem, and downloads MoDM, a start-up set of data items can be processed. As the system matures, and new versions, of DigMem, are released, this data store will grow and a vast catalogue of accepted types of data will emerge. However, for initial purposes, location, photographic, heart rate and accelerometer data are able to be collected and processed. These specific data types have been chosen because, when combined, they are capable of illustrating the user's movements, interactions, emotions (to a certain extent) and activities. Each user is also assigned a unique ID to which their raw data is indexed against. Figure 21, depicts the entity-relationship model that illustrates the design of this data store, whilst Figure 22 demonstrates the corresponding class diagram.

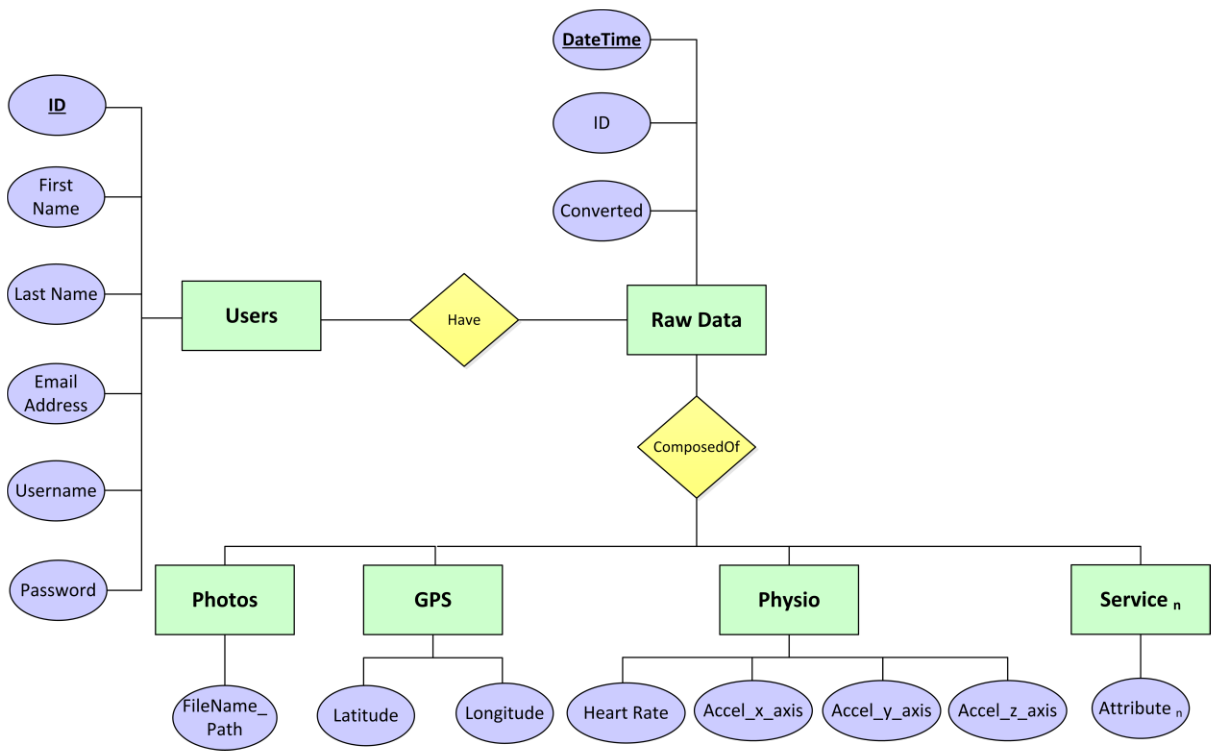


Figure 21 Raw Data Store Entity–Relationship Model

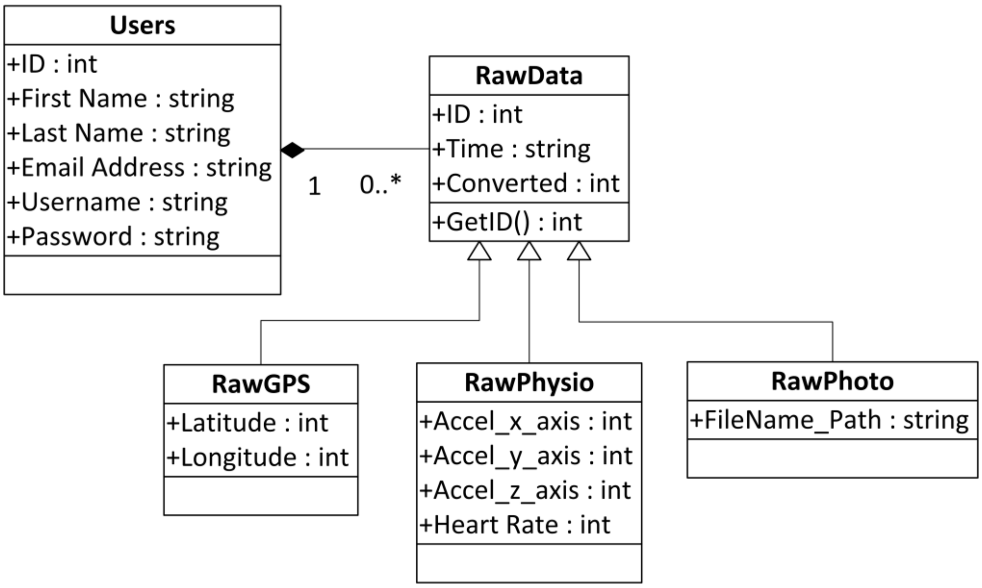


Figure 22 Raw Data Store Class Diagram

Figure 23, below, illustrates the procedure of storing raw data.

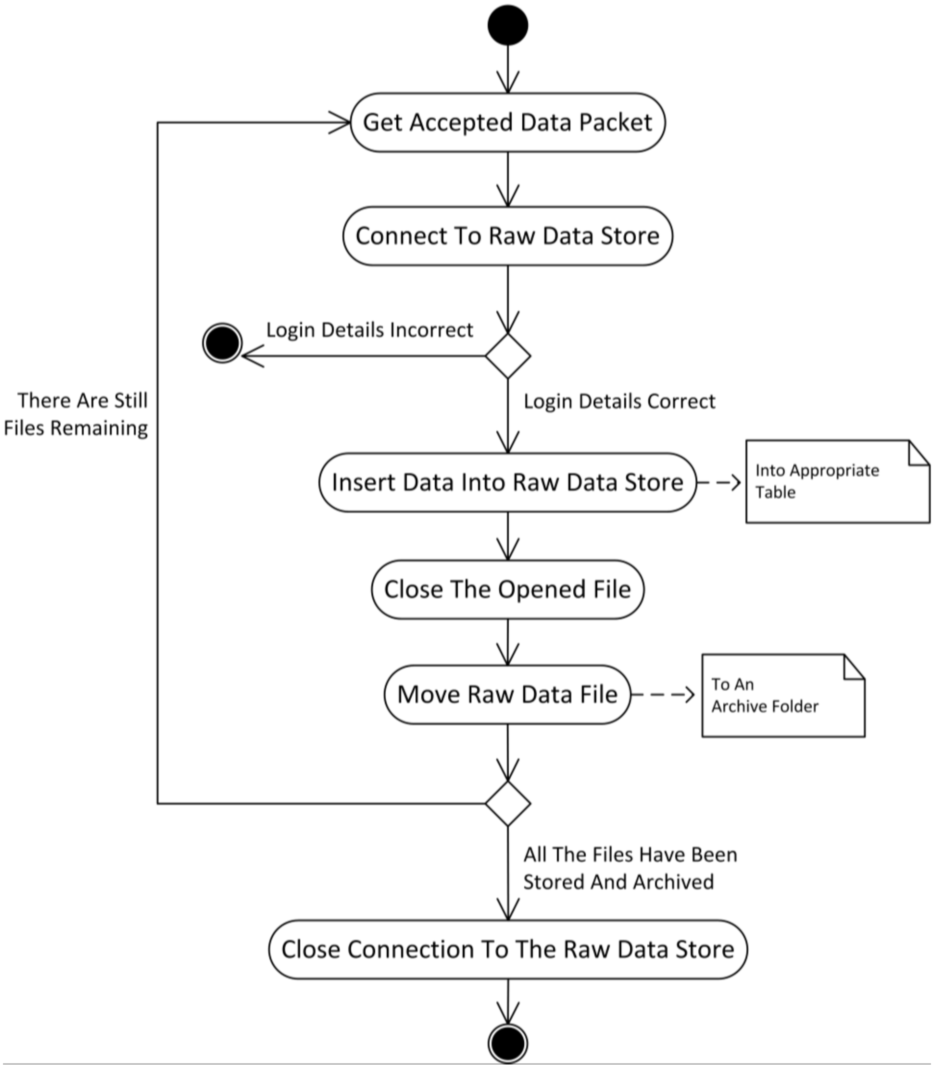


Figure 23 Storing Raw Data

As depicted in Figure 23, firstly, the header, of the data packet, needs to be checked, so that the system knows what table to store the data in and what user to index it against. Once the header has been checked the contents of the packet is checked to determine if it is populated with data. If data is missing, for example, in the case of collecting location data, if a time value exists but not latitude or longitude values or if all fields are null, this packet would be rejected, as it is incomplete. However, if the packet contains the correct information, for example, there are values for time, latitude and longitude, then this packet is valid. The data, within the packet, needs to be saved, in the Raw Data Store, within the location table. After deeming a packet acceptable, a connection to the Raw Data Store is required. Once successfully connected, the data is read in and stored, within the appropriate table. The opened file is closed, and the original file is then moved to an archive folder. This process is repeated for all files in the Internet folder. Once all the files have been processed, the connection, to the Raw Data Store, is terminated. This process is run as a batch process every *x* minutes to transfer collected information. Once the raw data has been stored, conversion, into the Raw Dataset occurs. Figure 24, below, illustrates this procedure.

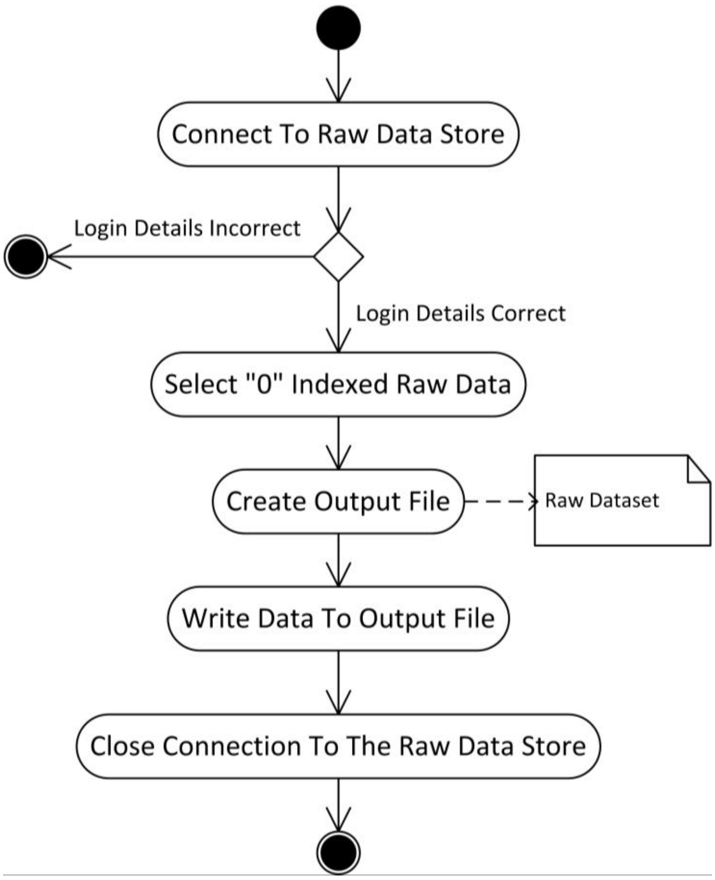


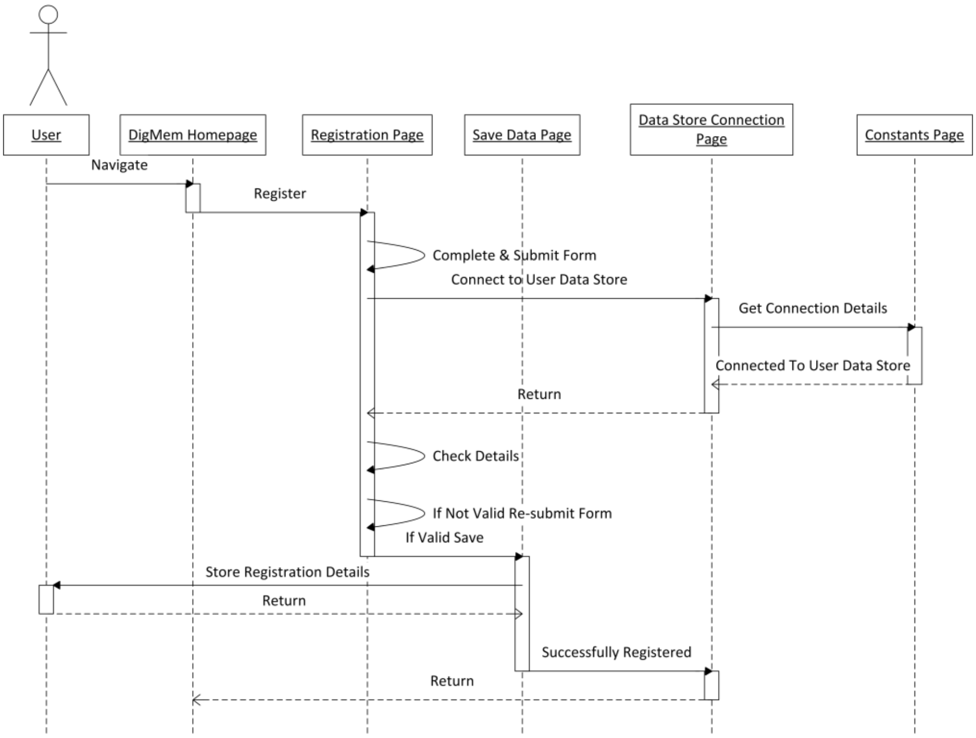
Figure 24 Converting Raw Data in the Raw Dataset

A connection to the Raw Data Store is required. After establishing this connection, all unprocessed raw data is selected, along with the user’s ID, and an output file is created. The data is then written to the file, creating a dataset of the raw information. After the data has been transformed, the connection, to the Raw Data Store, is terminated. Each table, within the Raw Data Store, is processed to create a Raw Dataset that contains all of the user's data. This dataset is then pre-processed and features are extracted. These features are then analysed and used to create a new Feature Set, which is saved in the Feature Store. Each evening, the Raw Data Store is checked for any unprocessed data that needs to be appended to the dataset and subsequently used to extract features from.

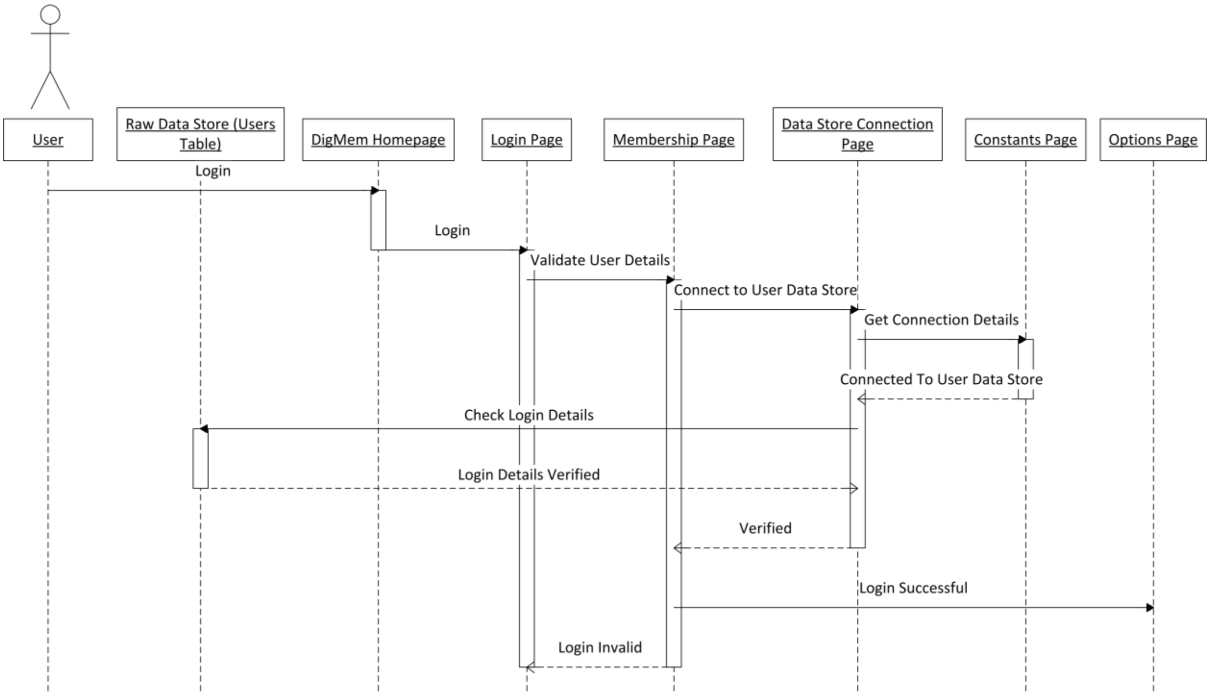
This section has detailed how the DigMem Server aspect of the system enables raw data, which the user has collected from MoDM, to be checked, stored and transformed into a set of features. As time passes, this element grows with the user. Once the Feature Store has been populated with data, it is then ready for the user to create their memory boxes or to question their data. The following section details how the DigMem Web Application achieves this.

3.2.3 DigMem Web Application

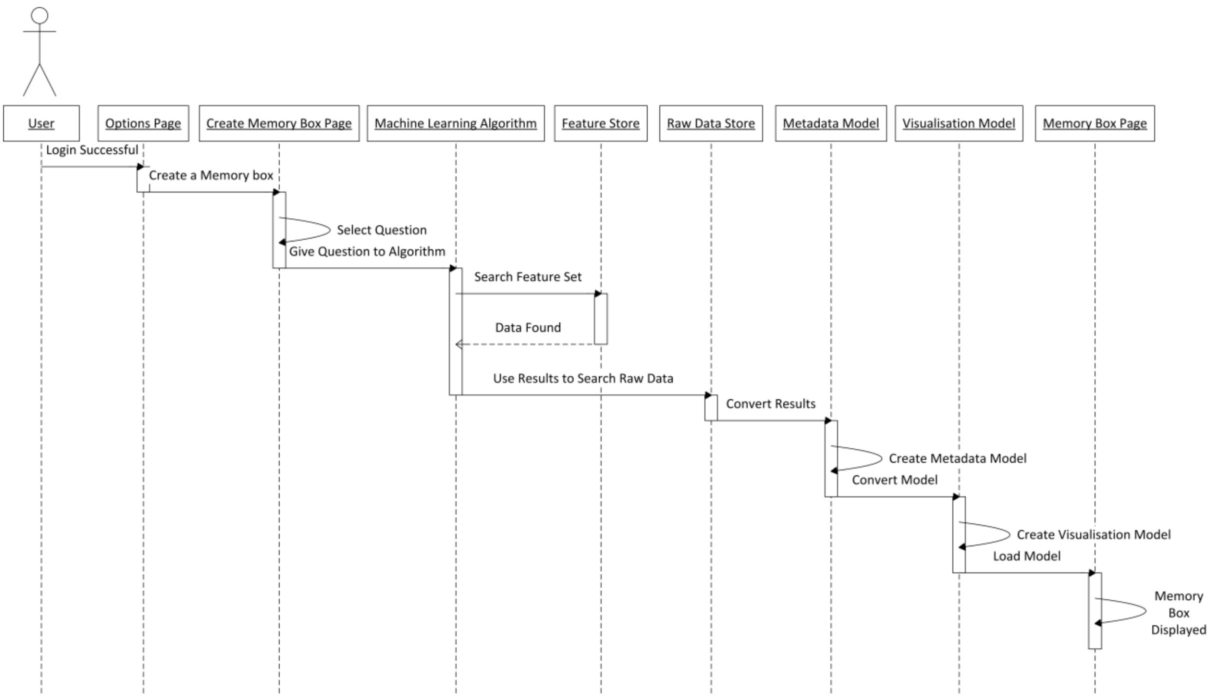
This section details the design of the DigMem Web Application that has been developed, during the research. The following sequence (see Figure 25) and activity (see Figure 26) diagrams illustrate this aspect of the system. These diagrams demonstrate the necessary steps that are taken in order for the user to register, log into DigMem, create a memory box or question their data. The following section provides a full description of this process.



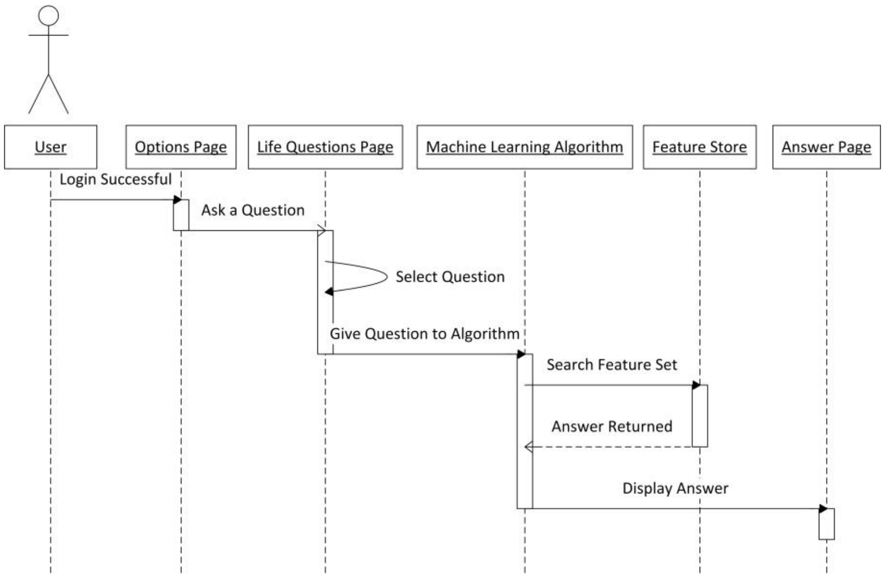
a) Registration Process



b) Login Process



c) Create Memory Box Process



d) Question Life Data Process

Figure 25 DigMem Web Application Sequence Diagrams

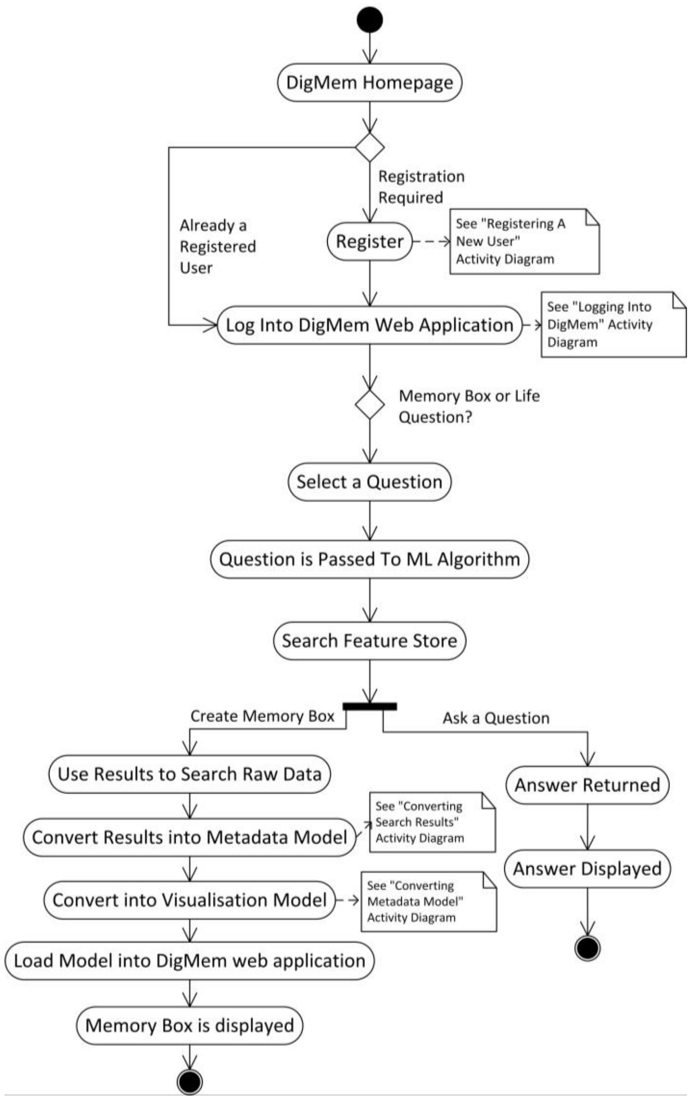


Figure 26 DigMem Web Application Activity Diagram

The layout of the DigMem web application is depicted, below, in Figure 27. The yellow pages are not visible to the user. These pages operate in the background and are used to store the Raw Data Store connection details and to verify the registration and login details.

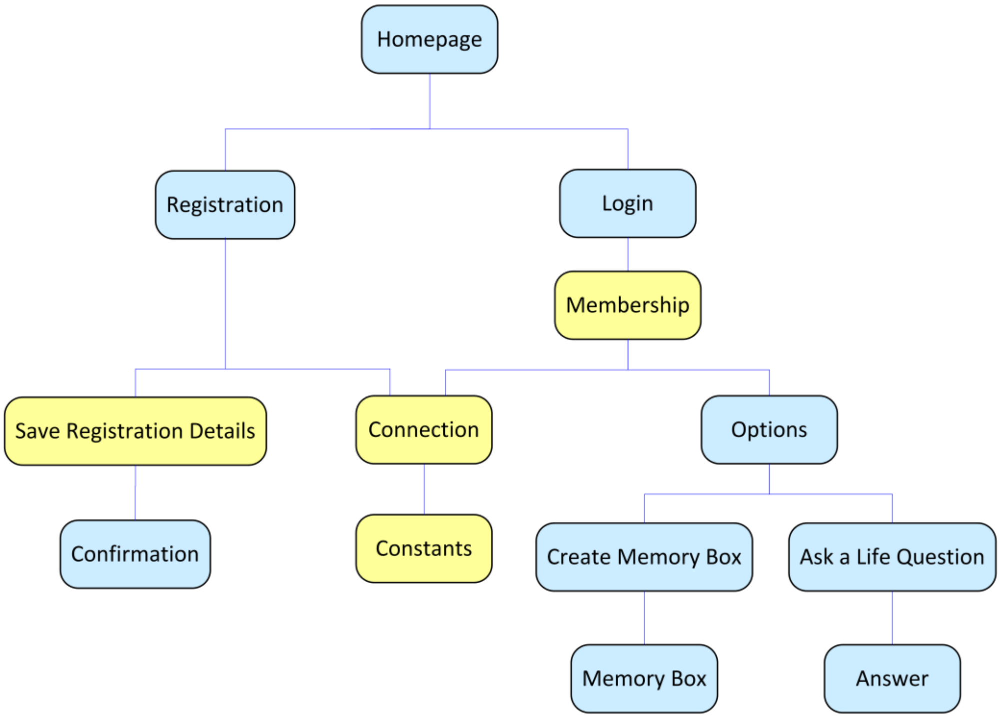


Figure 27 DigMem Sitemap

Figure 28, below, illustrates the design of the Homepage. On every page, the logo banner is present. The logo image is a hyperlink back to the homepage. After reading a short summary of the system, the user is either able to log in, or they must first register.

Figure 28 Homepage Design

In the case of registering a new user, navigation to the Registration page is required. Figure 29, below, illustrates the design of this page.

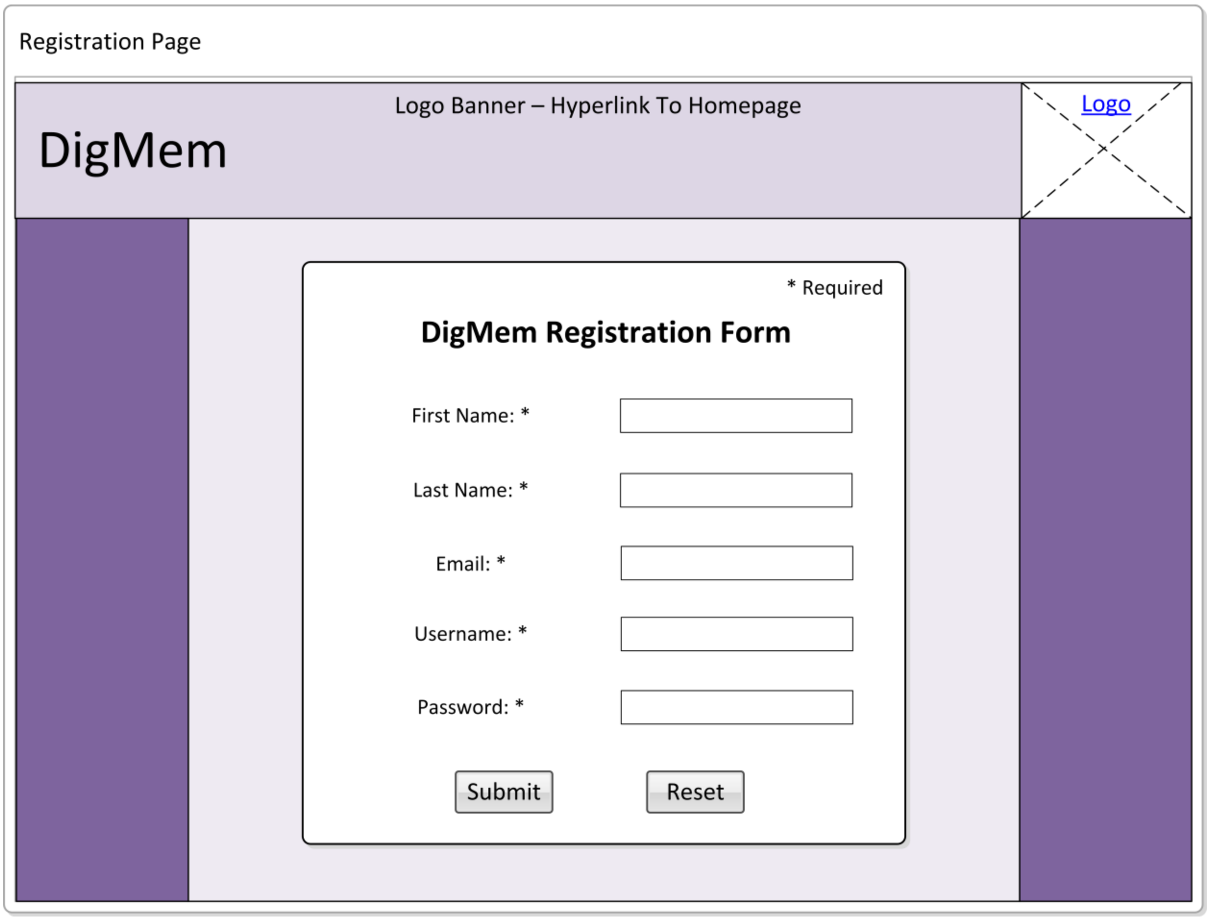


Figure 29 Registration Page Design

Figure 30, below, illustrates the activity diagram for registering. Meanwhile, Figure 31 illustrates the related class diagram. After all the fields, of the form, are completed, and the "Submit" button has been clicked, a connection to the Raw Data Store occurs. The Connection page (see Figure 27) handles this process. This page gets the connection details, from the Constants page (see Figure 27), and if these are correct, establishes a connection to the Raw Data Store. Once a connection has been established, the user's information is verified to check that all the required information has been submitted. If all the information is not there, the user is directed back to the Registration page. However, if all the details are present, then the Save Data page (see Figure 27) handles inserting the data into the "Users" table of the Raw Data Store.

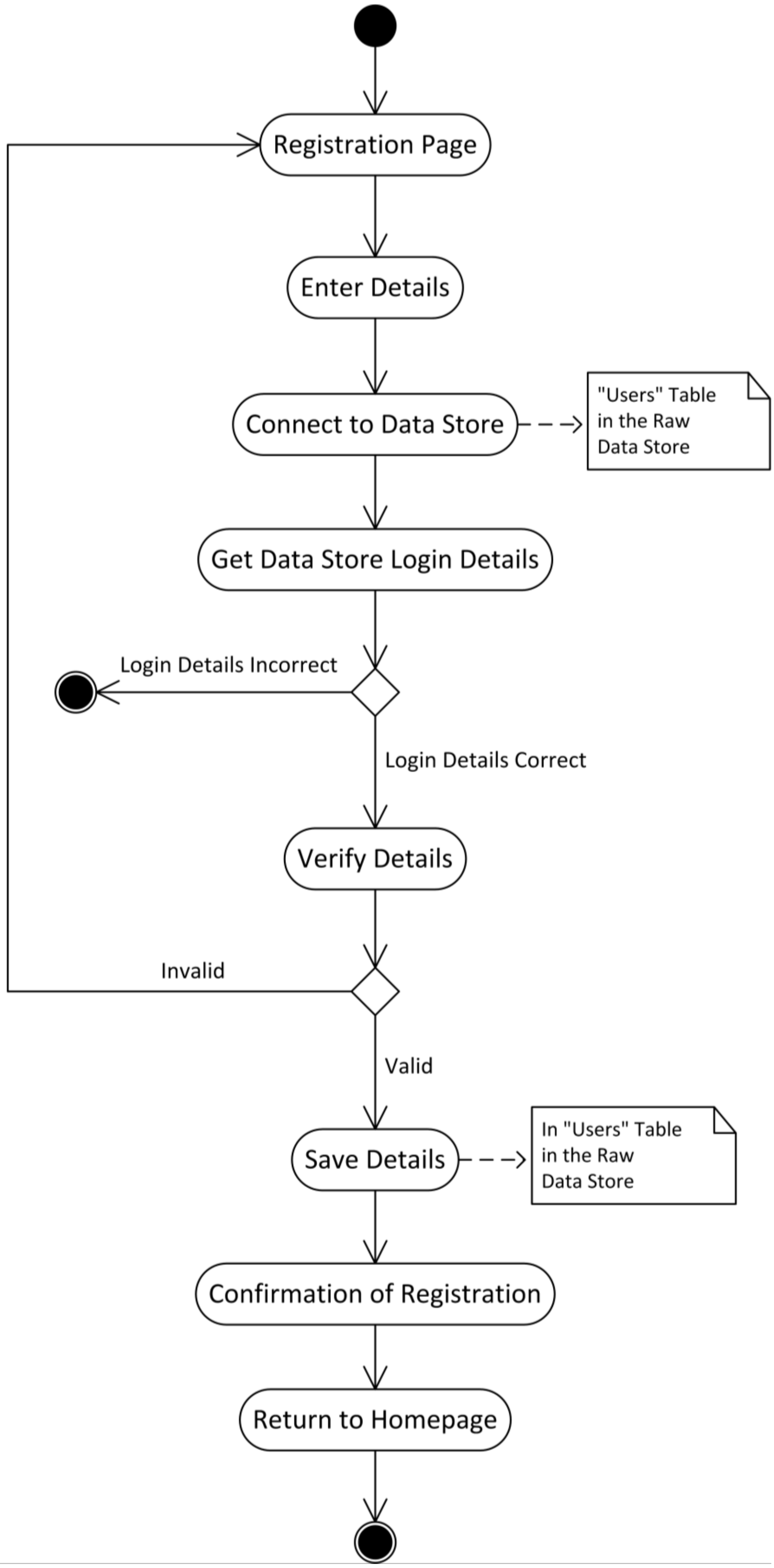


Figure 30 Registering a New User Activity Diagram

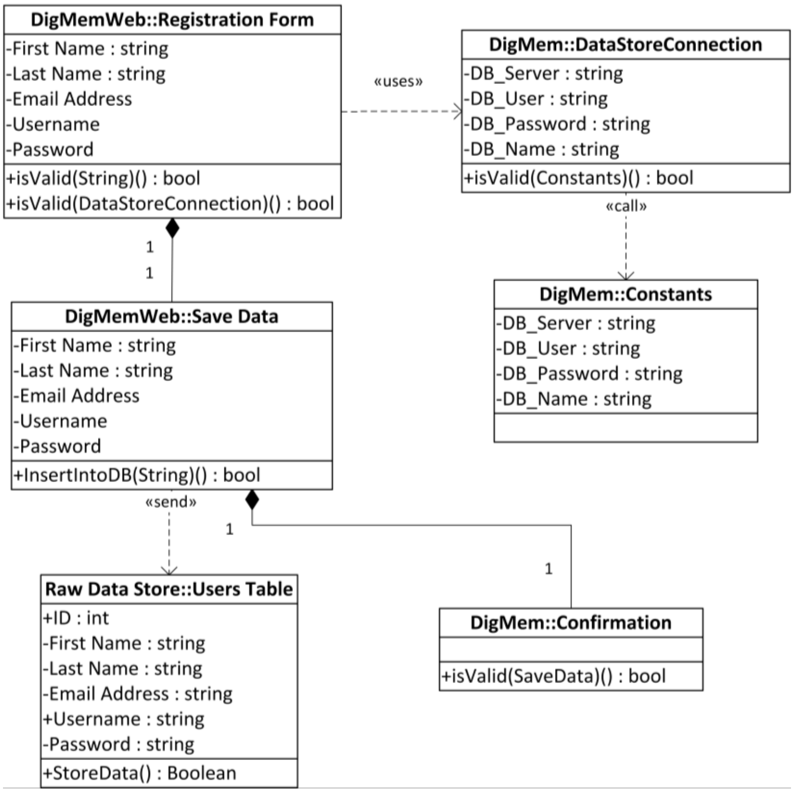


Figure 31 Registering a New User Class Diagram

Upon insertion of the data, the user is presented with the Confirmation page, verifying that they have successfully been registered with DigMem. Figure 32, below, illustrates the design of this page. The user is then asked to log into the system.

Figure 32 Confirmation Page Design

In the case of logging into the DigMem system, navigation to the Login page is required. Figure 33, below, illustrates the design of this page.

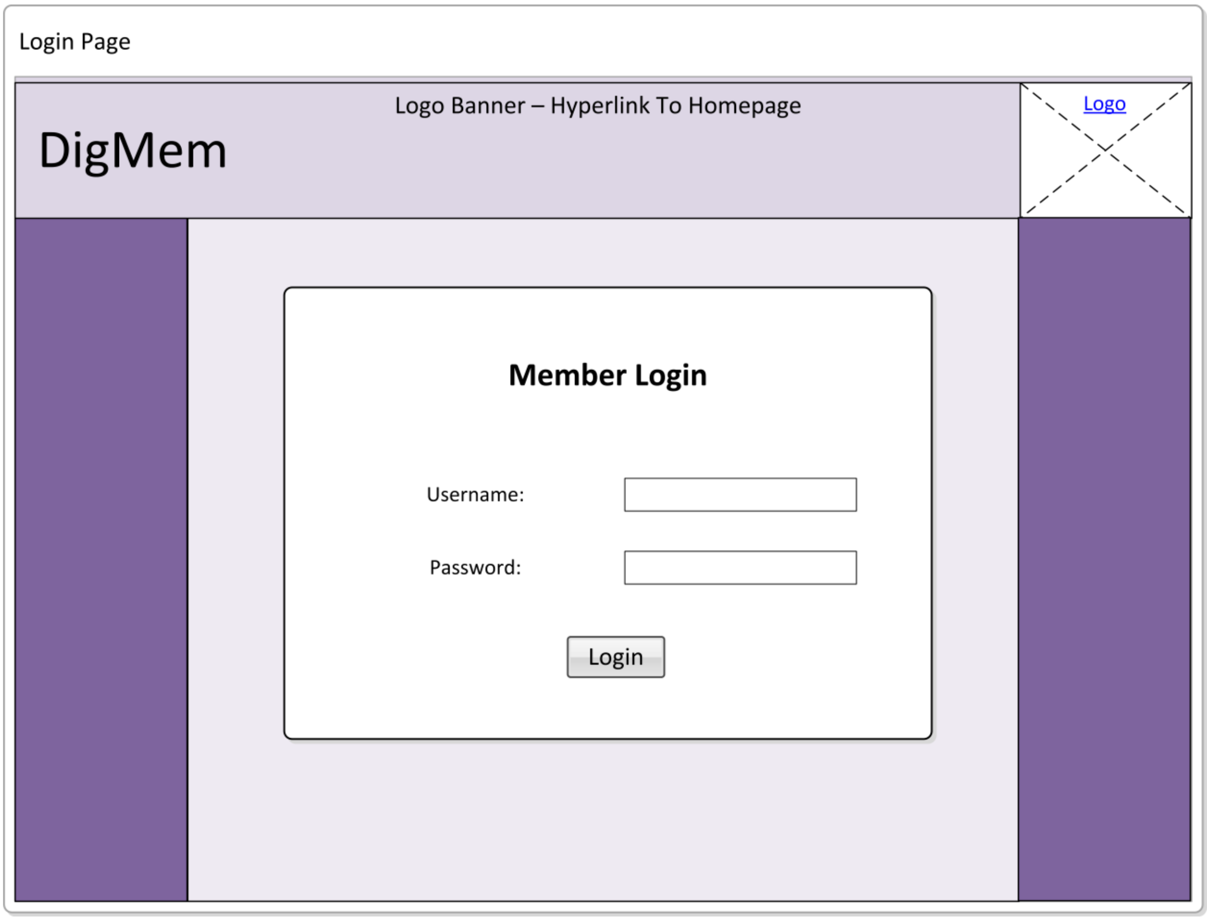


Figure 33 Login Page Design

Figure 34, below, illustrates the activity diagram for logging into DigMem. Meanwhile, Figure 35 illustrates the related class diagram, and Figure 36 illustrates the use case diagram. After all the fields, of the form, are completed, and the “Login” button has been clicked, a connection to the Raw Data Store occurs. The Membership page (see Figure 32) handles verifying the login details. However, this page needs to call on the Connection page (see Figure 27) first. This page gets the connection details, from the Constants page (see Figure 27), and if these are correct, establishes a connection to the Raw Data Store. Once a connection has been established, the user’s login information is verified. If all the information is not there, the user is directed back to the Login page. On the other hand, if all the details are present, then the user is directed to the Options page (see Figure 37), where they can either create a memory box or ask the system questions about their data.

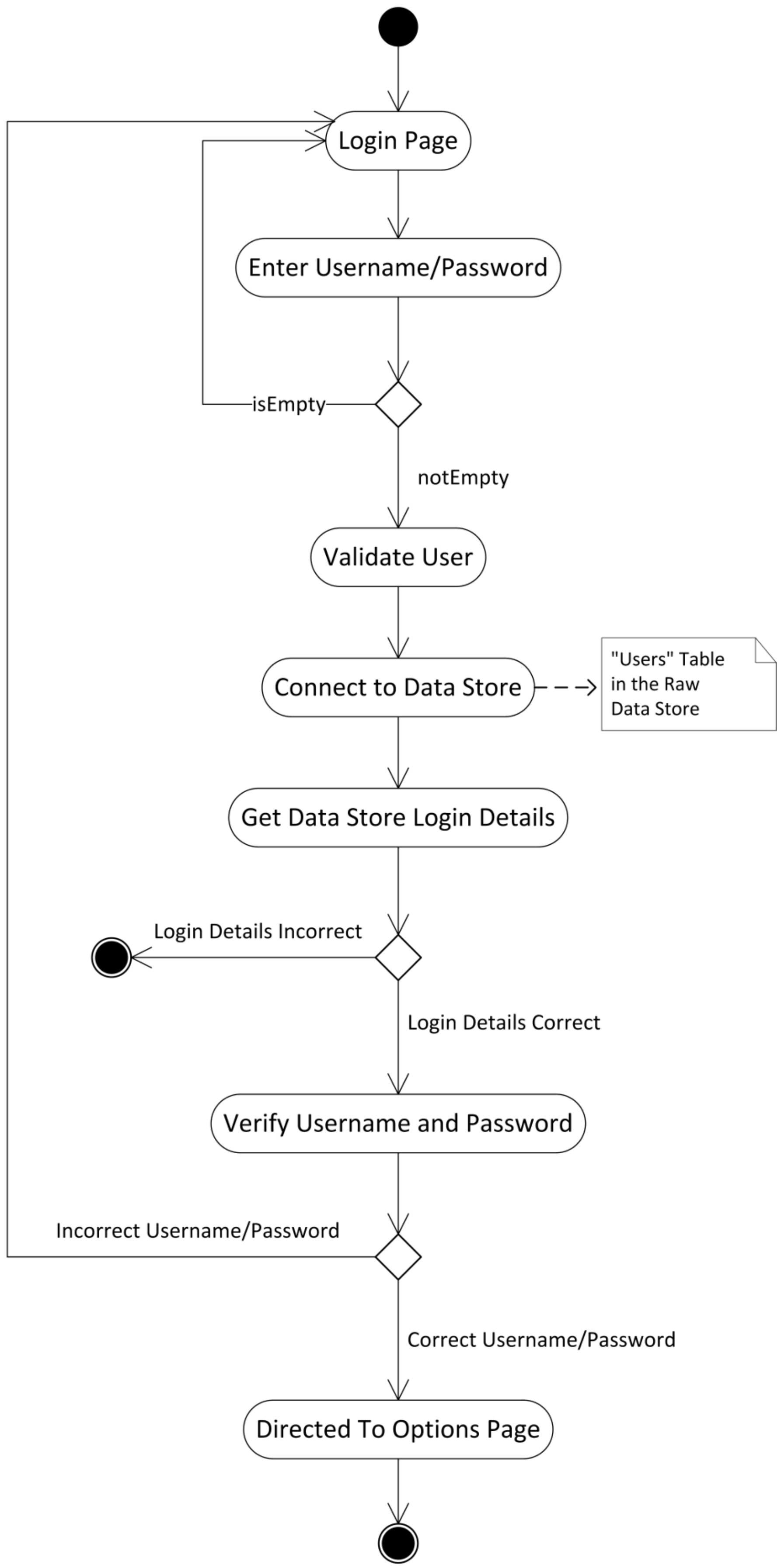


Figure 34 Logging into DigMem Activity Diagram

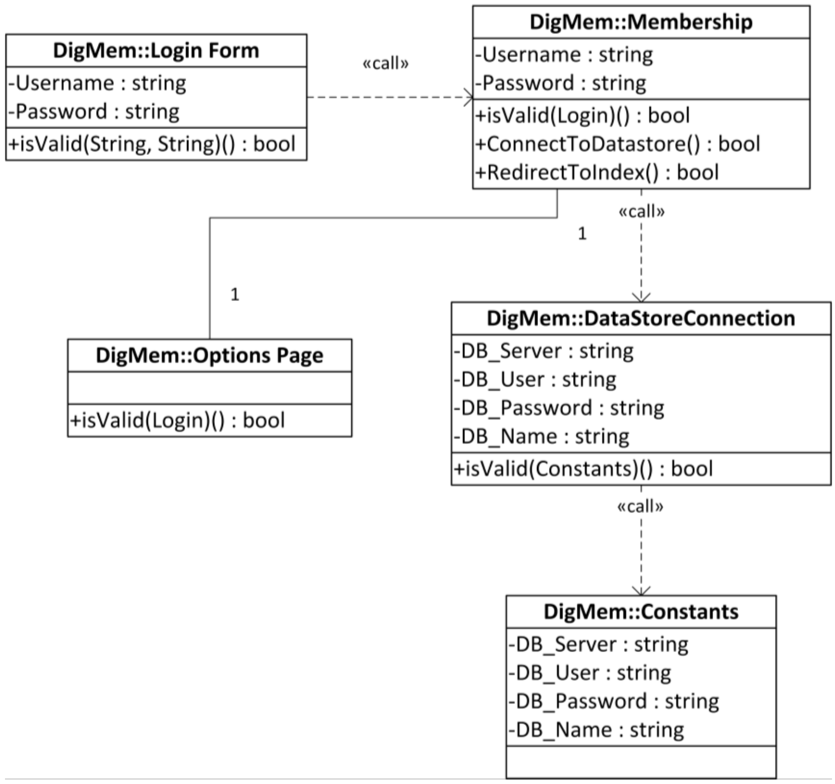


Figure 35 Logging into DigMem Class Diagram

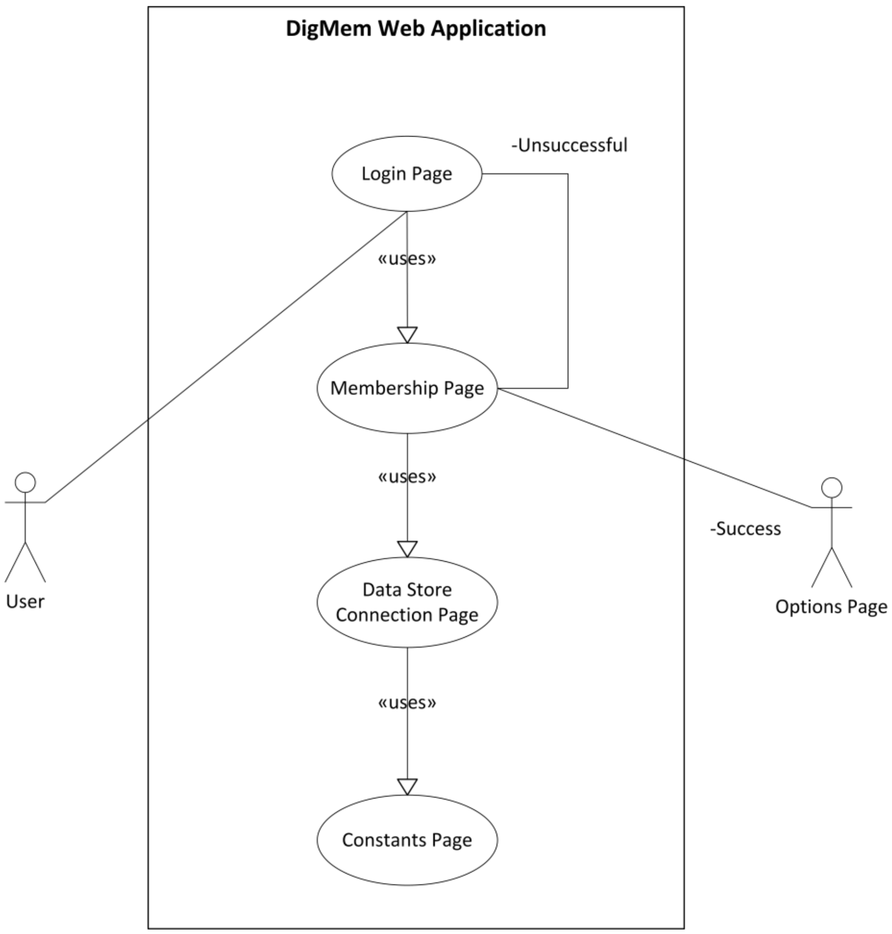


Figure 36 Logging into DigMem Use Case Diagram

After logging in, using the Options page, the user is able to select how they want to query their data. Figure 37, below, illustrates the design of this page.

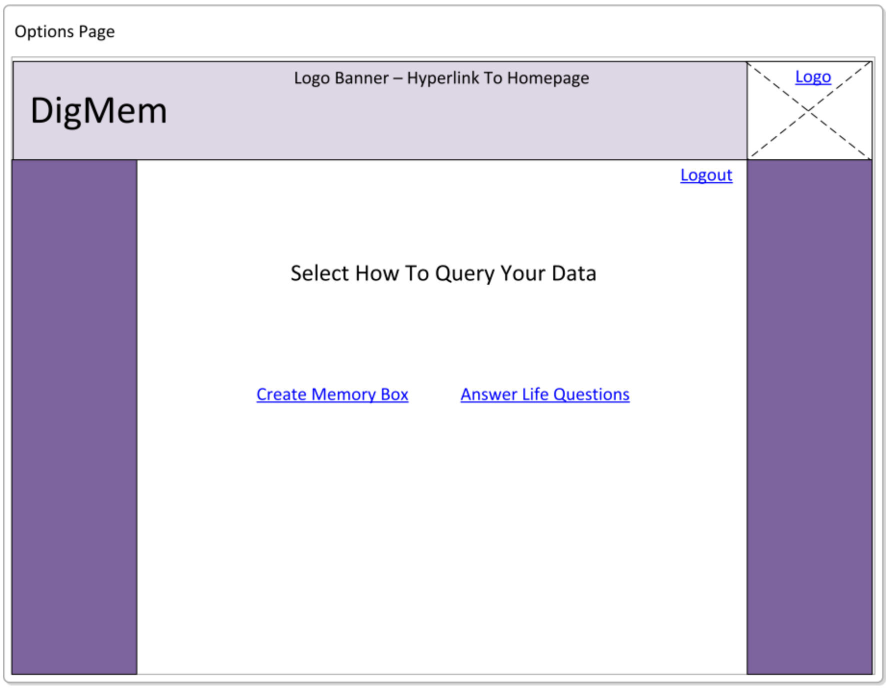


Figure 37 Options Page Design

Depending on which route the user takes, different questions are displayed. In the case of selecting “Answer Life Questions”, the user is taken to the Questions page and is able to search their memory data, using a set of questions. The answer (i.e. Yes/No) is then displayed. Figure 38, below, illustrates the design of this page.

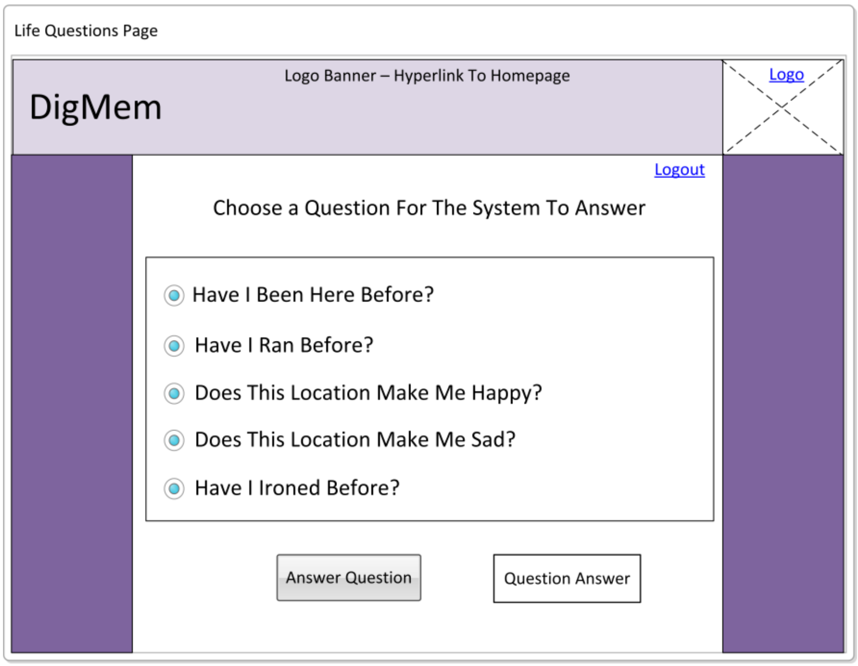


Figure 38 Ask the System a Question Page Design

In the case of selecting "Create Memory Box", the user is taken to the Create Memory Box page and is able to search their memory data, using a different number of questions. Figure 39, below, illustrates the design of this page.

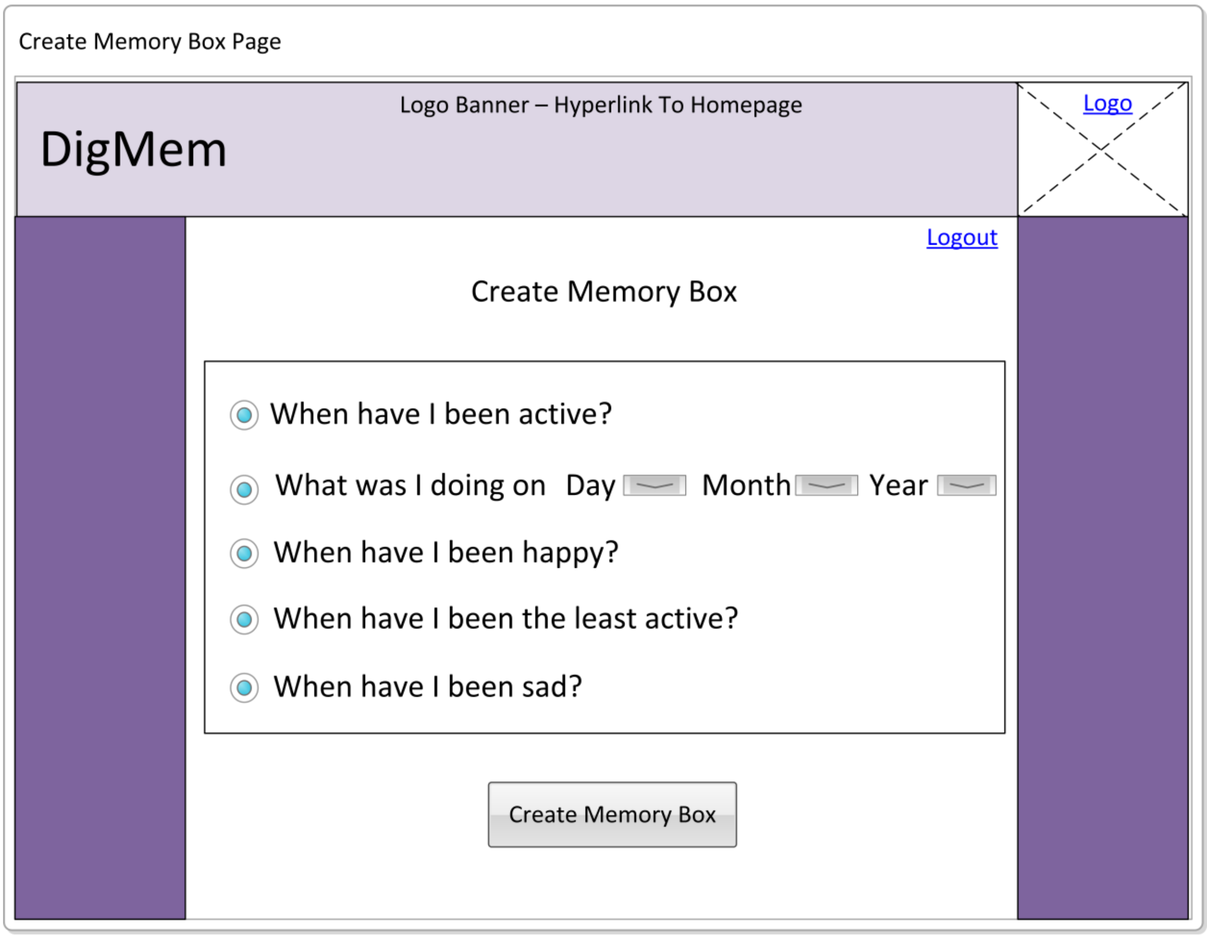


Figure 39 Create Memory Box Page Design

In the case of asking the system questions, classification algorithms will need to be trained on a variety of parameters. For example, emotions and activity recognition can be learnt, i.e. the features that represent being happy, sad, angry, etc. and also what constitutes running, walking, sitting, standing, etc. The Feature Set is unlabelled and is comprised of many features that represent the different types of data that DigMem supports. For example, features that represent accelerometer and location data are present, as well as examples of other data. Figure 40, below, illustrate an example of such a feature vector.

[[[Time][Accel][Location][Heart Rate][Emotion][…]]]

Figure 40 Example Feature Set Vector

Taking question two, in Figure 38, as an example, when this question is asked the feature vectors, from the feature set, are passed into an activity algorithm. The algorithm then matches this set of features to a training set of pre-defined activities. Upon recognizing the activity, the activity label is inserted into the vector. This vector is then used in the activity classification algorithm to answer the user's question. The answer is displayed to the user. In the case of creating a memory box, clustering algorithms are used to bring similar pieces of data together. Using the time feature in the results, the Raw Data Store is then searched to find all instances that match the time stamp. Figure 41, below, illustrates both of these processes.

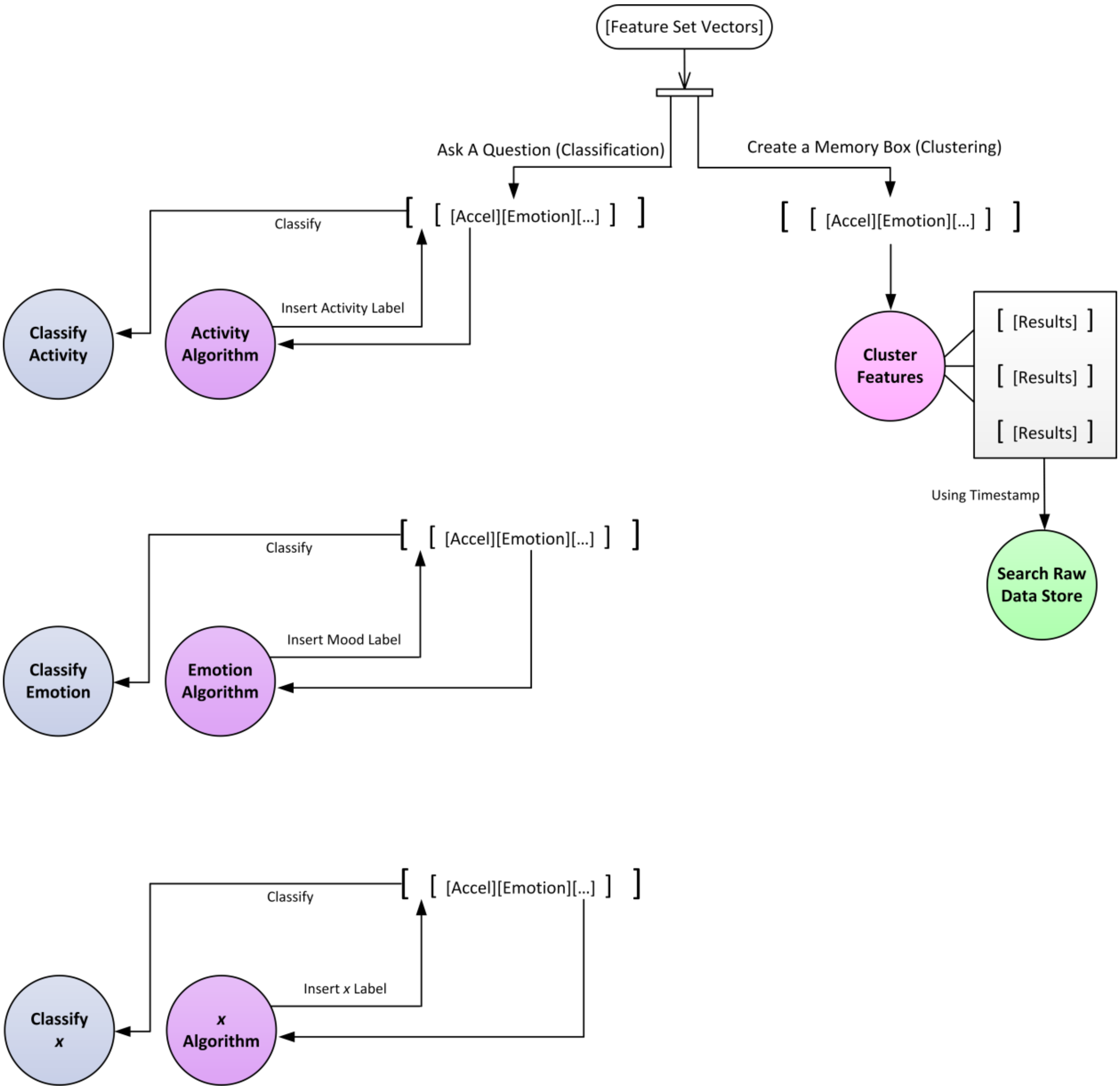


Figure 41 Method to Answer Different Types of Questions

In the case of creating a memory box, as previously stated, using the time feature in the results, the Raw Data Store is then searched to find all instances that match the time stamp. Once all of the related information has been found, it is then converted into a Metadata Model. Figure 42, below, illustrates this procedure. The obtained search results are written to a file, thus creating the new metadata model.

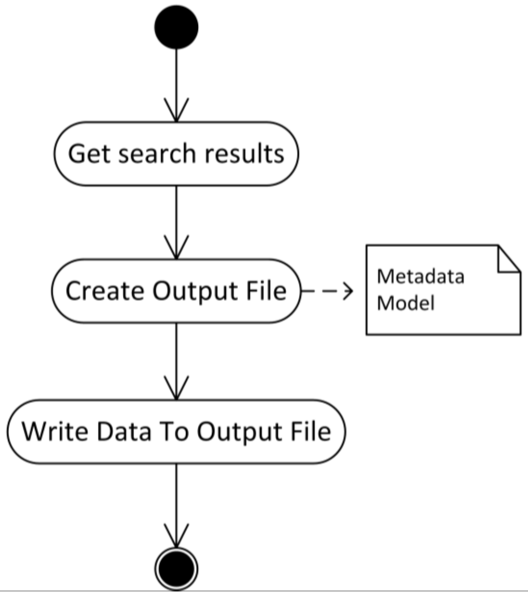


Figure 42 Converting the Search Results into a Metadata Model

After the metadata model has been created, the information is transformed again, into a visualisation model. Figure 43, below, illustrates this procedure. The earlier created metadata model (see Figure 42) is obtained, and the information is written to a new file, thus creating the visualisation model. As stated previously, the reason behind this double conversion is due to flexibility. The metadata model enables the data to be easily manipulated, and transformed, into any other format, for further analysis. The visualisation model is then used to transform the raw information into graphical items, within the DigMem web application.

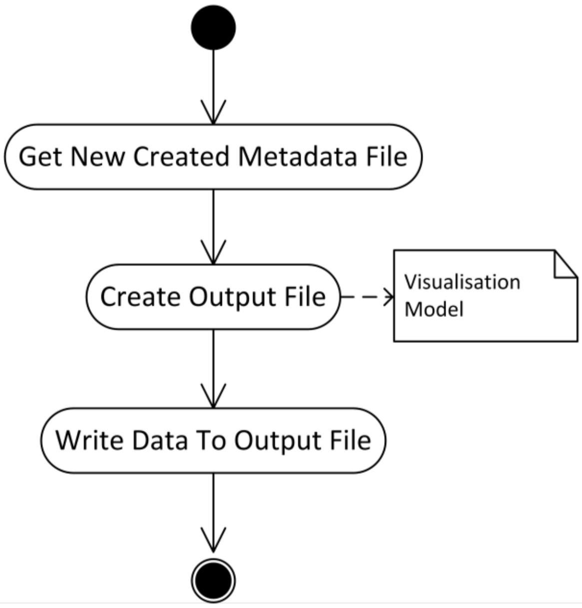


Figure 43 Converting the Metadata Model into a Visualisation Model

The model is then loaded into the DigMem web application, thus creating the memory box. Figure 44, below, illustrates the design of this page.

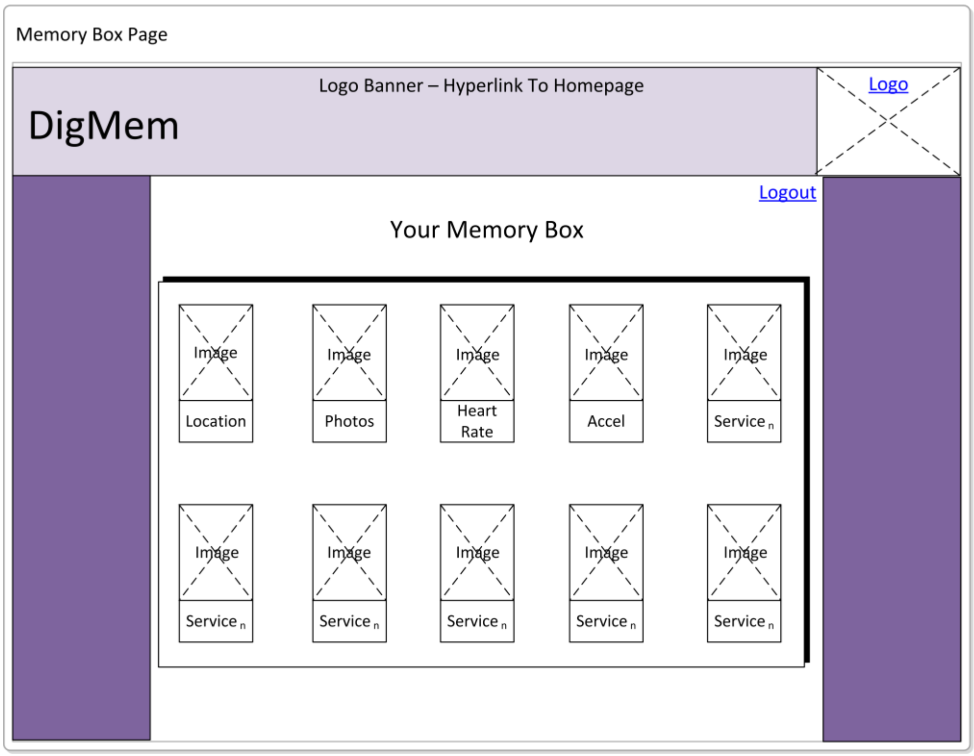


Figure 44 Memory Box Page Design

The memory box (see Figure 44) displays images of all of the inputs that DigMem is currently capable of collecting. When the user clicks on an image, of their selected input, a new window opens. This window graphically displays the previously searched information. Figure 45, below, illustrates the design of these sample service pages.

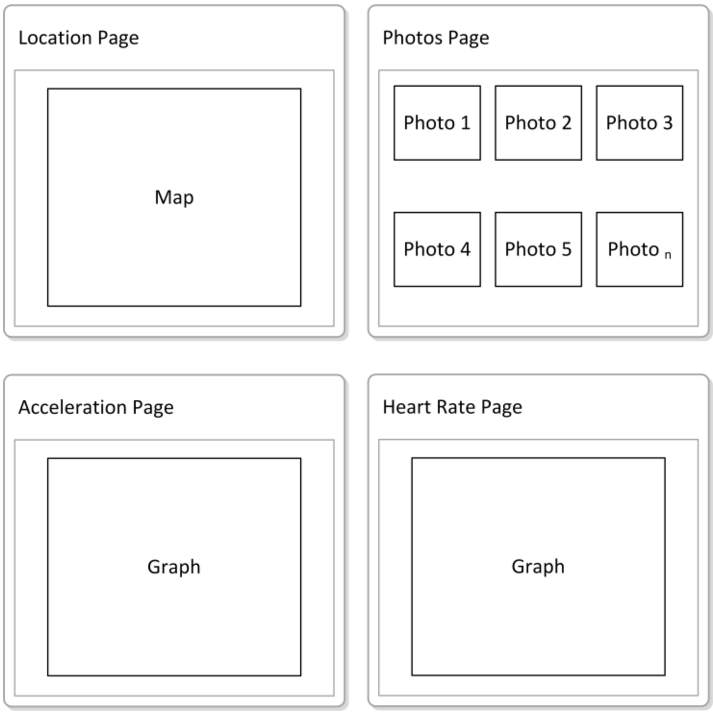


Figure 45 Services Page Design

For example, the user searches for information from a particular date and time. When they click on the "Location" image, a new "Location Page" opens, and their location coordinates, from that time, are plotted on a map. When the user clicks on the "Photos" image, the "Photos Page" opens and displays all of their photos, from that time. Similarly, clicking on the "Accel" or "Heart Rate" images opens the "Acceleration" or "Heart Rate" pages, and their physiological data, from the same time, are plotted, as a graph. All of the user's clustered data has now been brought together into one central location.

3.3 Summary

In this chapter, the DigMem system has been presented. The design specification and system architecture have been discussed, as well as the design of the Mobile DigMem (MoDM), DigMem Server and DigMem Web Application components. The chapter examines how these components work together to enable the user to create memory boxes of events and to ask questions about their data. As data is collected over months, years and decades, any time during our lives can be recorded and those details brought back. Using this configuration, more dynamic and detailed human digital memories can be created. The use of pervasive devices, linked data and machine learning are central to the system, and are what make it unique. By incorporating data from outside sources, as well as from body sensors, a better understanding of ourselves can be retrieved and reasoned over. DigMem is flexible enough to grow with the user and allows an entire lifetime to be digitally captured.

In realising the design, the following chapter discusses the implementation of the core system components of the system. This includes illustrating how devices connect to the P2P network, the device-specific services that have been implemented to test the design, how data extraction of the raw data occurs, how the raw dataset has been created and the implementation of the DigMem web application. As such, the implementation illustrates how data has been collected and processed so that memory boxes can be created or life questions answered. In concluding the chapter, a demonstration of the DigMem prototype is also presented.

Chapter 4

Implementation

In the previous chapter, the design of the system has been discussed. The chapter described the design specification and architecture of the system, as well as detailing the design of each of the three main components of DigMem. Using this design, this chapter discusses the implementation of the system. Each of the main components of the system is described, in detail, to illustrate its development. As such, the chapter illustrates how devices connect to the P2P network so that data can be collected and the device-specific services (photographic, location, heart rate and accelerometer) that have been implemented to test this approach. Furthermore, the chapter illustrates how this raw data is automatically extracted from the cloud and inserted into the DigMem Server. It also demonstrates how the raw dataset is created and the implementation of the DigMem web application. In addition, in order to test the design, a prototype is presented, which illustrates how human digital memory data is collected and processed so that life questions can be answered, and memory boxes created.

4.1 System Components

DigMem is composed of a number of core components. These elements enable the system to create an ad-hoc P2P network, capture information and process and transform this raw data. The following section describes the implementation of the services that have been developed in order to achieve this.

4.1.1 Peer-to-Peer Network

The MoDM middleware (Dobbins, Merabti, et al. 2012b) that has been developed, during the research, enables heterogeneous devices to communicate and share resources (services), for providing human digital memory data. In order to connect these devices together a P2P network is required. This has been implemented using the JXTA P2P protocol (Sun Microsystems Inc. 2003). JXTA was chosen because it is a generalised P2P protocol that allows any connected device on the network (from mobile phones to PDAs, from computers to servers) to communicate and collaborate, as peers (Sun Microsystems Inc. 2003). It also enables developers to build and deploy interoperable P2P services and applications (Sun Microsystems Inc. 2003). Because the protocols are independent of both programming language and transport protocols, heterogeneous devices with completely different software stacks can interoperate with one another (Sun Microsystems Inc. 2003). It is also implementable on every device with a digital heartbeat (Gong 2001). It is for these reasons that the prototype utilizes JXTA.

In order to create this P2P network, a rendezvous peer configures and starts the JXTA network and becomes a member of the default *NetPeerGroup*. Sensor devices, which provide services, and the user's device, which requests information, are then able to become part of the same *NetPeerGroup*. The code to achieve this is illustrated, partly, in Figure 46.

```
public void startJXTA(String principal, String password) {
        try {
            //Creating the NetworkManager
            networkManager = new NetworkManager(NetworkManager.EDGE, "My Local Network",
            instanceHome.toURI());

            //Persisting it to make sure the Peer ID is not re-created each time
            networkManager.setConfigPersistent(true);

            //Retrieving the Network Configurator
            config = networkManager.getConfigurator();
            netPeerGroup = networkManager.startNetwork();
            netPeerGroup = networkManager.getNetPeerGroup();
            rendezvous = netPeerGroup.getRendezVousService();
            discovery = netPeerGroup.getDiscoveryService();
        } catch (PeerGroupException e) {
            e.printStackTrace();
            System.exit(1);
        } catch (Exception e) {
            e.printStackTrace();
        }
        //Waiting for rendezvous connection
        while (!rendezvous.isConnectedToRendezVous()) {
                synchronized (rendezvous) {
                        try {
                                rendezvous.wait(1000);
                        } catch (InterruptedException e) {
                            e.printStackTrace();
                        }
                }
        }
        rdvId = null;
        Enumeration rdvEnum = rendezvous.getConnectedRendezVous();
        if (rdvEnum != null) {
          while (rdvEnum.hasMoreElements()) {
                rdvId = (PeerID) rdvEnum.nextElement();
                if (rdvId != null)
                        break;
                }
        Log.d(TAG, "I am connected to " + rdvId.toString());
       }
}
```

Figure 46 Connecting to the P2P Network with JXTA

A key component in communicating with peers, in the network, is pipe advertisements. These are used to describe a pipe communication channel and to create the associated input and output pipe endpoints (Sun Microsystems Inc. 2003). When an application wants to open a receiving communication with other peers, an input pipe is created and is bound to a specific pipe advertisement. The application then publishes the pipe advertisement so that other applications or services can obtain the advertisement and create corresponding output pipes to send messages to that input pipe (Sun Microsystems Inc. 2003). In the case of MoDM, the sensor device creates an input pipe and a subsequent pipe advertisement. This advertisement is then published locally (on the device) and remotely (to the peer group), whilst the input pipe waits for a connection from the user's device. In order for the sensor device to advertise the use of its services, *Module Class*, *Module Specification* and *Module Implementation* advertisements are also required, which are published locally and remotely.

Once the user selects a service, from the MoDM interface (see Figure 47), a handler sends out an advertisement discovery message.

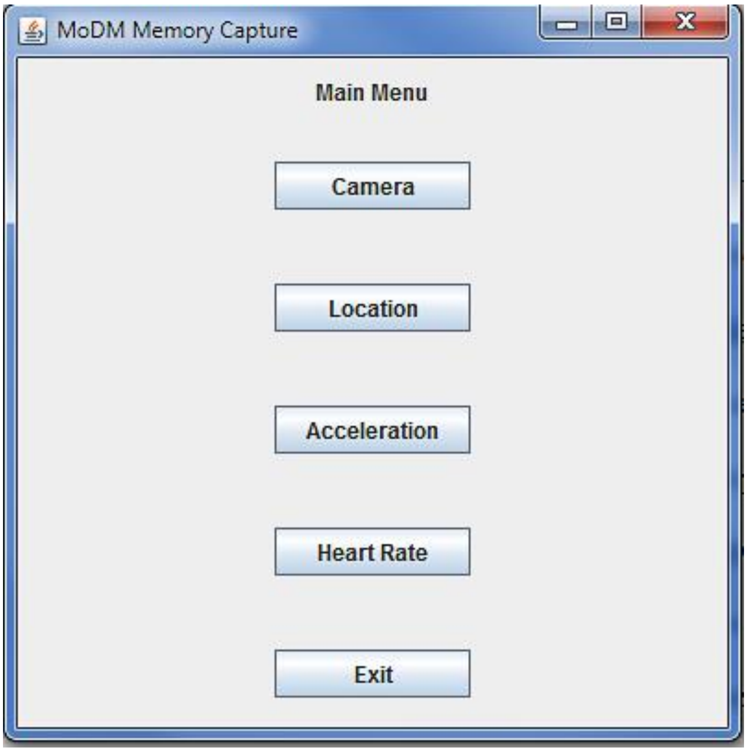


Figure 47 MoDM Interface

This message is used to find all peers with the particular *Module Specification* advertisement of the required service. For example, in the case of finding a device with a camera service, the advertisement "JXTASPEC:CAMERA" would be broadcasted to the network. Once the advertisement has been found, the user peer extracts the *Pipe Advertisement*, and creates the corresponding output pipe. The code to achieve this is illustrated, partly, in Figure 48.

```
if ( adv1 instanceof ModuleSpecAdvertisement) {
        ModuleSpecAdvertisement app = (ModuleSpecAdvertisement)adv1;
        System.out.println("FOUND MSA ADVERTISEMENT ----> " + "ADV NAME:" + app.getName() + " ----> PIPE
        ADVERTISEMENT ID: " + app.getPipeAdvertisement().getID());

        pipeAdv = app.getPipeAdvertisement();
        String modulename = app.getName();

        System.out.println("EXTRACTED THE PIPE ADVERTISEMENT: " + pipeAdv + "FROM THE MODULE SPEC: " + app);

        if (modulename.equals("JXTASPEC:CAMERA")){
          //once pipe advertisement has been found create the ouptput pipe
          createCameraOutputPipe(pipeAdv);

          Thread connectionThread = new Thread() {
                public void run() {
                      System.out.println("Starting the Camera Connection Handler");
                      CameraConnectionManager = new CameraConnectionHandler(myCameraOutputPipe);
                      System.out.println("Connection Handler started...");
                }
          };
          connectionThread.start();
          }
```

Figure 48 Retrieving the Pipe Advertisement in JXTA

The output pipe then connects to the sensor peer's waiting input pipe. This set of bidirectional pipes are now ready for communication. At this point, the P2P network has been established, and the devices are able to exchange information.

4.1.2 Device-Specific Services

In line with the data types that DigMem can support, and to demonstrate the system, a number of device-specific services have been developed. These include camera, location, accelerometer, and heart rate services. After finding the required service advertisement and establishing a connection, the user peer sends a JXTA message, containing a service command, to the sensor peer and waits for a response, i.e. the requested data. Upon receiving the message, the sensor peer extracts the command and starts the appropriate method to retrieve the required data. For example, in the case of the camera service, photos are taken every ten seconds and are sent, in a response message, back to the user device. The code to achieve this is illustrated, partly, in Figure 49. In the case of the GPS service, when a location has been retrieved, the coordinates (latitude and longitude) and time are sent, in a response message, back to the user. The code to achieve this is illustrated, partly, in Figure 50. Upon receiving a JXTA message, the user's device grabs the message from the event, automatically extracts the data and saves it to a temporary location. In this instance, temporary storage is the user's local Dropbox[8] folder. The following code, illustrated partly in Figure 51, demonstrates this procedure.

[8] https://www.dropbox.com/

```
int delay = 2000;   // delay for 2 sec.
int period = 10000;  // repeat every 10 sec.
Timer timer = new Timer();

//Takes pictures at a fixed rate
timer.scheduleAtFixedRate(new TimerTask() {
        public void run() {
                mCamera.takePicture(null, mPictureCallback, mPictureCallback);
        }
   }, delay, period);
}

Camera.PictureCallback mPictureCallback = new Camera.PictureCallback() {
        public void onPictureTaken(byte[] imageData, Camera c) {
             if (imageData != null) {
                  Intent mIntent = new Intent();
                  Log.d(TAG, "context is: " + mContext);
                  StoreByteImage(mContext, imageData, 50, "ImageName");
                  boolean image = StoreByteImage(mContext, imageData, 50, "ImageName");
                  mCamera.startPreview();
                  setResult(FOTO_MODE, mIntent);

                  if (image = true);
                  try
                    {
                         Log.d(TAG, "Sending Response....");
                         ConnectionHandler.sendImageResponse(ConnectionHandler.pipe, lastImageSaved);
                         Log.d(TAG, "Image Sent Successfully: " + lastImageSaved);
                    }
                  catch(Exception e)
                    {
                         Log.d(TAG, "ERROR: Sending Response: " + e.getLocalizedMessage());
                    }
                }
        }
};
```

Figure 49 Capturing and Sending Photograph Data

```
private void updateWithNewLocation(Location location) {

             if (location != null) {
                  double lat = location.getLatitude();
                  double lng = location.getLongitude();
                  latString = "" + lat;
                  longString = "" + lng;
                  latLongString = "Lat:" + lat + " Long:" + lng;
             }

             else {
                  latLongString = "No location found";
             }

             Try

             {
                  Log.d(TAG, "Sending Response....");
                  ConnectionHandler.sendGPSResponse(ConnectionHandler.pipe, latLongString);
                  Log.d(TAG, "Location Sent Successfully!");

             }

             catch(Exception e)

             {
                  Log.d(TAG, "ERROR: Sending Response: " + e.getLocalizedMessage());

             }
}
```

Figure 50 Capturing and Sending Location Data

```
public void pipeMsgEvent(PipeMsgEvent event) {
Message msg;
try {
	//grab the message from the event
	msg = event.getMessage();
		if (msg == null) {
			System.out.println("Received an empty message, returning");
			return;
		} else {
			System.out.println("Message received: " + msg.toString());
			//Extract the image from the message
			MessageElement msgElement = msg.getMessageElement("Photo Details", "Photo");
			long size = msgElement.getByteLength();
			if (size > 0)
			{
				byte[] ImageData = new byte[(int)size];
				System.out.println("File received: " + ImageData);
				SimpleDateFormat sdf = new SimpleDateFormat ("dd-MM-yyyy-HH-mm-ss");
				currentTime = new java.sql.Timestamp(System.currentTimeMillis());
				timestring = sdf.format(currentTime);
				directory = "C:\\Users\\researcher\\Dropbox\\MemoryData\\mymemories\\Photos";

				//Location of where to save the photos
				File ImageDirectory = newFile(directory);
				FileOutputStream fileOutputStream = null;
				fileOutputStream = new FileOutputStream(ImageDirectory.toString() + "\\" + timestring + ".jpg");
				System.out.println("FileOutputStream: " + fileOutputStream);

				//Save the photo to the directory using the timestamp as the name
				fileOutputStream.write(msg.getMessageElement("Photo").getBytes(false));
				lastImageSaved = new String(ImageDirectory.toString() + "\\" + timestring + ".jpg");
				fileOutputStream.close();

				System.out.println("Photo saved: " + lastImageSaved);

				CameraSearchAction.PhotoReceivedfield.getText();
				CameraSearchAction.PhotoReceivedfield.append("\n" + "Photo received:" + lastImageSaved);
			}
		}
	} catch (Exception e) {
		System.out.println("Error getting message from event: " + e.getMessage());
		e.printStackTrace();
		System.exit(-1);
	}
	}
}
```

Figure 51 Extracting and Saving Photograph Data

In the case of the physiological services using accelerometer (located on the ankle, chest and hand) and heart rate sensors, data is transferred, via Bluetooth, to the mobile device and saved in Dropbox. The following code, in Figure 52, demonstrates how the user's device would connect to a Bluetooth physiological sensor. In this scenario, it has been necessary to repeat activities with each sensor, individually, so that each sensor can submit their data to the mobile device.

```
try {
	BluetoothDevice device = foundDevices.get(params[0]);
	socket = device.createRfcommSocketToServiceRecord(uuid);
	socket.connect();
	Toast.makeText(getBaseContext(),"Connected to:" + foundDevices, Toast.LENGTH_LONG).show();
} catch (IOException e) {
	Log.d("BLUETOOTH_CLIENT", e.getMessage());
	Toast.makeText(getBaseContext(),"Connection Failed", Toast.LENGTH_LONG).show();
}
```

Figure 52 Capturing and Sending Physiological Data

This method of data collection does not rely on specific pieces of hardware to capture human digital memory data. Any pervasive device that can provide information, and running the MoDM middleware, can contribute to the memory. This plug-and-play platform enables the system to maintain flexibility. It also allows the creation of memories to occur within many different environments.

4.1.3 Data Extraction and Data Set Creation

The DigMem Server (Dobbins, Merabti, Fergus & Llewellyn-Jones 2013a) enables retrieved information, from device-specific services, to be transformed into a universal set of features. This feature set is used later, by the DigMem web application, to create a memory box or to answer life questions about the user's data. Once data has been collected, the development of a number of python scripts enables the information to be transferred from the cloud directory to a remote MySQL database (Raw Data Store). For example, in the case of storing retrieved location information, this code is illustrated, partly, in Figure 53.

```
#Iterates through all the files within the specified directory and writes each file's data to mysql

#Connects to the memorydata database
con = MySQLdb.connect(server, username, password, database)

#creates the cursor from the database
cur = con.cursor()

#opens the csv file
root_src_dir = sourceDirectory
root_dst_dir = destinationDirectory

for infile in glob.glob( os.path.join(root_src_dir, '**.csv') ):

    print "current file is: " + infile

    openfile = open (infile)

    csv_data = csv.reader(openfile)

    for row in csv_data:

        print row

        #cursor executes the sql query to insert the data into the rawGPS table and set Converted to ‘0’
        cur.execute('INSERT INTO rawGPS(UserID,DateTime,Latitude,Longitude,Converted)' 'VALUES (1, %s, %s, %s, 0)', row)
```

Figure 53 Python Script to Insert Data into the Raw Data Store

Once the data packet has been obtained, the data is inserted into the appropriate MySQL table, and the value "0" inserted into the "Converted" column to signify that this piece of data is new, and that it has not been processed yet. Afterwards, the file is closed, and moved to an archive folder. The same method occurs to process accelerometer and heart rate data. The cloud has a limited amount of storage space. Therefore, by moving the data, space is automatically created for incoming information. This is a very simple and effective way of using Cloud Computing technology to send data, via the Internet, to different locations (Dikaiakos et al. 2009). However, if photographic data is to be processed, then the files are first moved to the archive location.

Once archived, their filenames and paths are stored in the database. The reason behind transferring the photos in a slightly different manner is that if the photos were still in the cloud, then this location would be stored as their filename and path, which is incorrect. By first transferring the photos, to the archive directory, ensures that the correct filename and path have been inserted into the data store. This is important later, when the photos are retrieved for the memory box. Figure 54 demonstrates how photographic data is stored.

```
for infile in glob.glob(os.path.join(root_dst_dir, '*.jpg')):
    # infile stores the complete path of the file
    print "Current File Being Processed is:  " + infile
    #use split to separate the path and name of the file
    (root_dst_dir, FILENAME) = os.path.split(infile)

    print " PATH is " + root_dst_dir
    print " FILENAME is " + FILENAME
    #use splitext() to seperate name of the file and the extension
    (ShortName, Extension) = os.path.splitext(FILENAME)
    print " Short name: " + ShortName
    print " Extension: " + Extension
    name = ShortName + "," + infile
    instring = name.split(',')
    print instring
    cur.execute('INSERT IGNORE INTO rawPhoto(UserID,DateTime,FileName_Path,Converted)' 'VALUES (1, %s, %s, 0)', instring)
```

Figure 54 Python Script to Insert Photograph Details into Database

Once data is in the Raw Data Store, another application service extracts the information and creates a new Raw Dataset. Figure 55, below, illustrates an overview of the process of extracting data and creating the datasets.

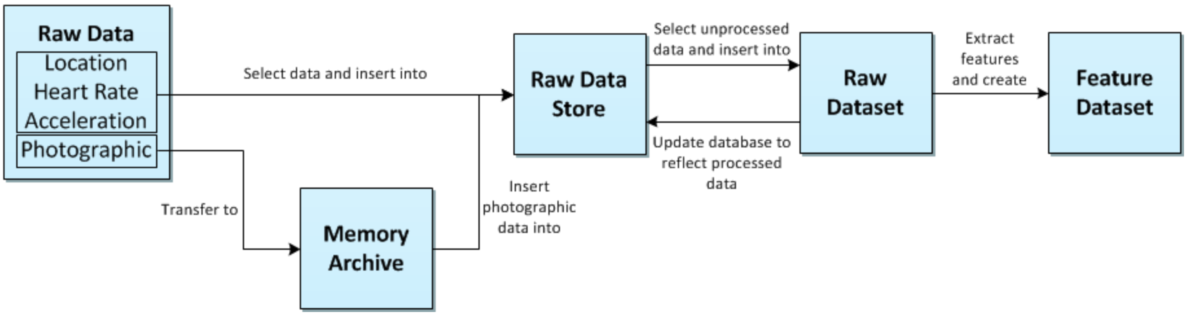


Figure 55 Data Extraction and Dataset Creation Process

Initially, all data items, which have been flagged as not yet having been processed (i.e. the "Converted" column is "0"), are selected from each of the tables in the Raw Data Store and appended to the Raw Dataset. Once the information has been stored, each row of data, where "Converted = 0", is updated to "Converted = 1". This indicates that the data item has now been stored in the raw dataset and prevents the same items being stored multiple times. When new items are processed, this dataset grows with the user and contains every piece of raw information. Features are then extracted from the raw dataset so that memory boxes and life questions can be answered. Figure 56, below, illustrates the class diagram that depicts the creation of the Raw Dataset.

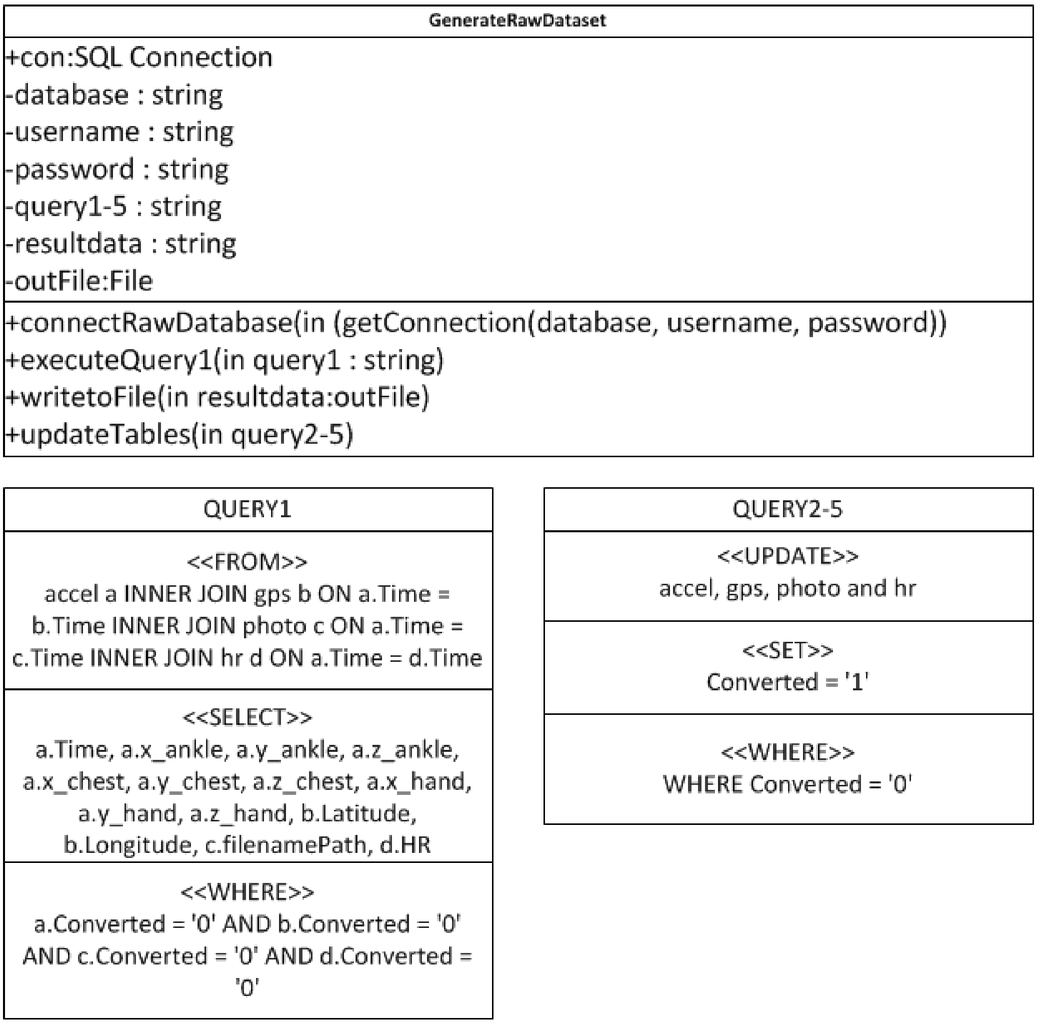


Figure 56 Class Diagram to Create Raw Dataset

This section has detailed how the DigMem Server aspect of the system has been implemented in order to store and transform the collected raw data into a dataset of features. Once the DigMem Server processes the raw information it is then ready to be searched. The results can either be transformed into a memory box or used to answer questions about the user's data. The following section details how the DigMem Web Application has been developed, in order to achieve this.

4.1.4 DigMem Web Application

This section details the implementation of the DigMem web application (Dobbins, Merabti, Fergus & Llewellyn-Jones 2013a) that has been developed, during the research. This aspect of the system enables the user to register, login, and search the universal feature set of human digital memory data, in a number of ways. Please refer to Appendix 2 for illustrations of all of the DigMem Web Pages that have been implemented.

Upon navigating to the DigMem homepage, the user is presented with a brief explanation about human digital memories and the DigMem system. In order to proceed, the user must be registered with the system. Upon entering registration details, each input field is individually checked. Figure 57, illustrates a sample of this Hypertext Preprocessor (PHP) code. If the field is empty, an error message is displayed. However, if it is not, then the value, of that field, gets stored. Further validation also checks if the user has entered a username that already exists, on the system or if they try to register with an existing email address.

```
<?php

include "classes/config.php";

$errorMessage = "";

if($_POST['submit'] == "Submit")
{
		//Get input registration details
		$firstname = $_POST['firstname'];

		$lastname = $_POST['lastname'];

		$email = $_POST['email'];

		$username = $_POST['username'];

		$password = $_POST['password'];

		//Check if a first name has been entered
		if(empty($_POST['firstname']))
		{
			$errorMessage .= "<li>Please enter your first name</li>";

		}
		//If a first name has been entered store it in the $firstname variable to insert later into the database
		else
		{
			$firstname;
		}

…
}
?>
```

Figure 57 Registration Form PHP Script Validation

Figure 58, below, illustrates the PHP code that is used to perform the MySQL validation. This piece of code is placed within the “Submit” button check that was illustrated in Figure 57.

```
//If all fields are populated then attempt to insert details into the database
if(!empty($_POST['firstname']) && !empty($_POST['lastname']) && !empty($_POST['email']) && !empty($_POST['username']) &&
!empty($_POST['password']))
{
	$dbusername = mysql_query("SELECT * FROM Users WHERE Username = '$username'");

	$dbemail = mysql_query("SELECT * FROM Users WHERE Email = '$email'");

	$sqlinsert = 'INSERT INTO Users (Firstname, Lastname, Email, Username, Password) VALUES
	("'.$firstname.'","'.$lastname.'","'.$email.'", "'.$username.'","'.$password.'")';

	  //Check if the username is already being used
	  if (mysql_num_rows($dbusername) > 0 ) {
	        echo "<p>This username is already taken</p>\n";
	  }
	  //Check if the email address is already registered
	  else if (mysql_num_rows($dbemail) > 0 ) {
	        echo "<p>This email address is already registered</p>\n";
	  }
	  //If all checks pass then insert data into database and redirect to confirmation page.
	  else {
	        mysql_query($sqlinsert);
	        //redirect to confirmation page
	        header($location) ;
	  }
}
```

Figure 58 Registration Form MySQL PHP Script Validation

Once registration is complete, the user is then able to log into DigMem. Validation on this form ensures that the username and password have been entered correctly. In order to check this information another PHP script has been developed. Figure 59, below, illustrates a sample of the code that is used to achieve this.

```
<?php

session_start();

require_once 'classes/Membership.php';

$membership = new Membership();


// The user has entered a username/password and clicked submit
if($_POST && !empty($_POST['username']) && !empty($_POST['pwd']))
{

	$response = $membership->validate_user($_POST['username'], $_POST['pwd']);
}

…

?>
```

Figure 59 Login Validation PHP Script

Firstly, the script starts a new session and then checks if the username and password fields are not empty. If they are populated with data, the *validate_user($un, $pwd)* function is used to get the values of the username and password fields to check if they match. Figure 60, below, illustrates a sample of the code that is used to achieve this.

```
<?php

require 'Mysql.php';

class Membership {

    function validate_user($un, $pwd)
    {
        $mysql = New Mysql();
        //Check the username and password on the Mysql page
        $checkdetails = $mysql->verify_Username_and_Pass($un, $pwd);

        if($checkdetails)
        {
            $_SESSION['status'] = 'authorized';
            header("location: search.php");

        } else return "Please enter a correct username and password";

    }
…
}
```

Figure 60 Validate User Function

In order to verify this information a connection to the Raw Data Store is required, and a Structured Query Language (SQL) query is then used to check the information. Figure 61), below, illustrates a sample of the code that is used to achieve this.

```
<?php

require_once 'includes/constants.php';

class Mysql {

        function verify_Username_and_Pass($un, $pwd)
        {
            $query = "SELECT *
                    FROM Users
                    WHERE Username = ? AND Password = ?
                    LIMIT 1";

            /* Create a prepared statement */
            if($stmt = $this->conn->prepare($query)) {

                /* Bind string parameters username and password*/
                $stmt->bind_param('ss', $un, $pwd);

                /*Execute the query */
                $stmt->execute();

                /* Get the results and if they match the inputs return true and pass this back to the Membership page*/
                if($stmt->fetch())
                {
                    $stmt->close();
                    return true;
                }
            }
        }
}
…
?>
```

Figure 61 Login Form MySQL PHP Script Validation

Upon successfully logging into the system, the user can choose to either create a memory box or have the system answer questions about their data. In the case of creating a memory box, the user is presented with the following page, illustrated in Figure 62, below.

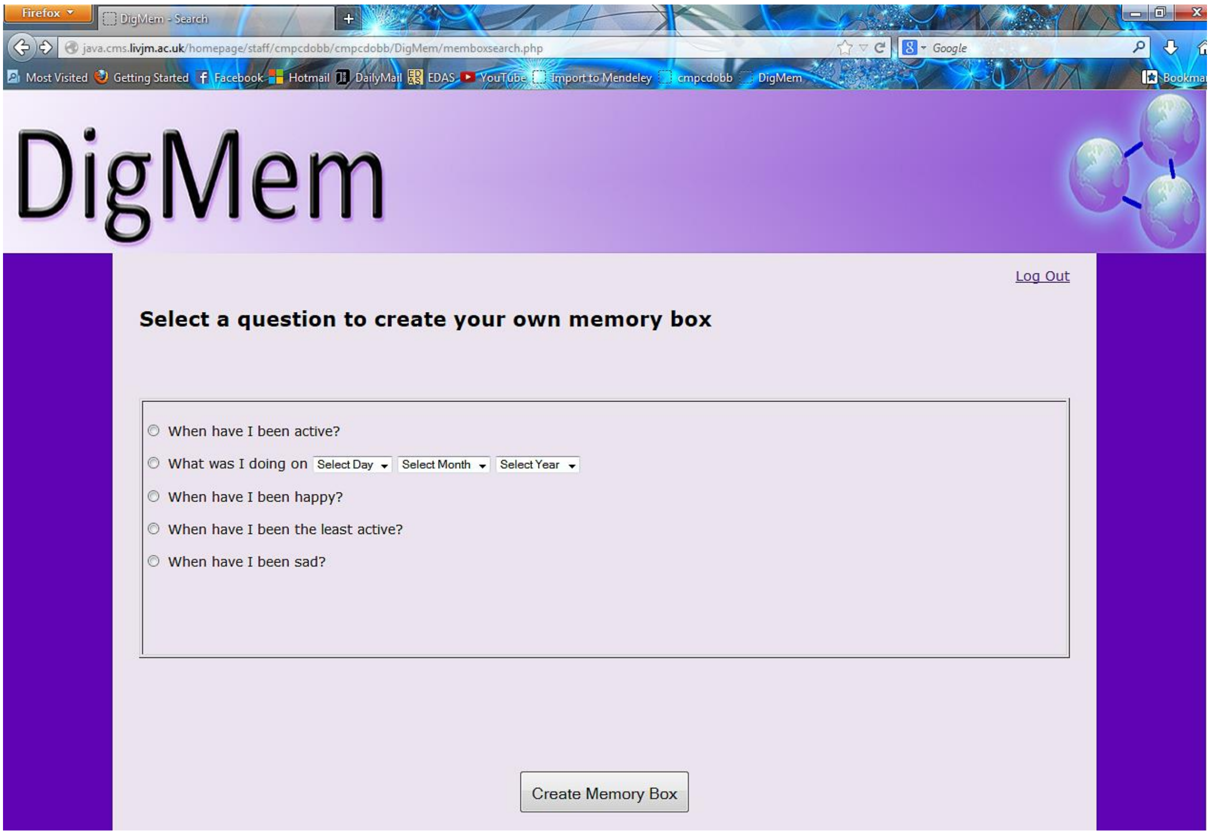


Figure 62 Create Memory Box Web Page

This page displays a list of questions. These questions are used to query the dataset of features so that memory boxes can be created. For example, in the case of selecting the first question, the system takes this query, and using the *k-means* algorithm, clusters data in the Feature Set based on periods of high energy. The code to achieve this is illustrated in Figure 63.

```
[cidx2,cmeans2] = kmeans(data,2);
ptsymb = {'bs', 'r^', 'md', 'go', 'c+'};

for i = 1:2
      clust = (cidx2 == i);
      plot3(data(clust,1),data(clust,2),data(clust,3),ptsymb{i});
      hold on
end

plot3(cmeans2(:,1), cmeans2(:,2), cmeans2(:,3),'ko');
  hold off
  xlabel('Ankle Mean'); ylabel('Hand Mean'); zlabel('Ankle RMS');
  view(-137,10);
  grid on
legend('Cluster 1', 'Cluster 2', 'Location','NE')
```

Figure 63 Clustering Code

Once these times have been identified, the timestamp is used to search the Raw Data Store for the corresponding raw information, including the previously omitted photograph data. After the related information has been found, the data is converted into individual RDF models (i.e. gps, photograph, accelerometer and heart rate models). By transforming the results into RDF enables information from different schemas to be merged and, more importantly RDF also supports the evolution of schemas over time, without requiring all the data to be changed (W3C 2004a). This feature is very important, and one of the main reasons to transform the data into RDF. Any human digital memory system needs to be flexible enough to withstand the evolution of technology and to enable people to use and create memories "across their entire lifetime" (Fitzgibbon & Reiter 2005). Through the use of RDF, as new standards and devices become available their data, as well as data collected twenty years ago, for instance, can all still be used in the creation of human digital memories. This is another unique aspect of the system. The metadata serializations provide rich semantic data structures that describe information. Using this model, the information can then be transformed into any other format. This method enables greater flexibility in the ways it can be processed. In order to display the collected information in the web application, the models are converted into the JavaScript Object Notation (JSON) format (visualisation model). JSON was chosen because it is lightweight and completely language independent, thus making it an ideal data-interchange language (Json.org n.d.). Furthermore, JavaScript has been used in the web application to display the memory box information and since JSON is a subset of JavaScript, it was an ideal choice. The following code, in Figure 64, illustrates this procedure.

```
String query = "SELECT * FROM rawgpsdata WHERE Time LIKE '07-03-2012-%' ORDER BY Time ASC";
try {
    //Connect to Raw Data Store
    Class.forName("com.mysql.jdbc.Driver");
    con = DriverManager.getConnection (database,username,password);
    pst = con.prepareStatement(query);
    rs = pst.executeQuery();

    //Create RDF Output File
    FileWriter outRDFFile = new FileWriter("C:\\Users\\cmpcdobb\\Documents\\Test\\gps_07-03.rdf");
    PrintWriter outRDF = new PrintWriter(outRDFFile);

    //Create JSON Output File
    FileWriter outJSONFile = new FileWriter("C:\\Users\\cmpcdobb\\Documents\\Test\\gps_07-03.json");
    PrintWriter outJSON = new PrintWriter(outJSONFile);

    //Write text to RDF file
    outRDF.println("<?xml version='1.0' encoding='UTF-8'?>");
    outRDF.println("<rdf:RDF");
    outRDF.println("xmlns:owl='http://www.w3.org/2002/07/owl#'");
    outRDF.println("xmlns:gps='http://myMemories/GPS/#'");
    outRDF.println("xmlns:rdf='http://www.w3.org/1999/02/22-rdf-syntax-ns#'");
    outRDF.println("xmlns:rdfs='http://www.w3.org/2000/01/rdf-schema#'>");
    outRDF.println();

    …
```

a)

```
        //Write text to JSON file
        outJSON.println("{");
        outJSON.println("  'head': {");
        outJSON.println("    'vars': [");
        outJSON.println("      'time',");
        outJSON.println("      'latitude',");
        outJSON.println("      'longitude'");
        outJSON.println("    ]");
        outJSON.println("  },");
        outJSON.println("  'results': {");
        outJSON.println("    'bindings': [");

          while (rs.next()) {
                dbtime1 = rs.getString(1); //UserID
                dbtime2 = rs.getString(2); //Time
                dbtime3 = rs.getString(3); //Lat
                dbtime4 = rs.getString(4); //Long

                outRDF.println("    <rdf:Description rdf:about='urn:http://myMemories/GPS/User#" + dbtime1 + "'>");
                outRDF.println("        <gps:time>"+ dbtime2 + "</gps:time>");
                outRDF.println("        <gps:is_atLatitude>"+ dbtime3 + "</gps:is_atLatitude>");
                outRDF.println("        <gps:is_atLongitude>" + dbtime4 +"</gps:is_atLongitude>");
                outRDF.println("    </rdf:Description>");
                outRDF.println();

                outJSON.println("      {");
                outJSON.println("        'time': {");
                outJSON.println("          'type': 'literal',");
                outJSON.println("          'value': '" + dbtime1 + "'");
                outJSON.println("        },");
                outJSON.println("    ");

                outJSON.println("        'latitude': {");
                outJSON.println("          'type': 'literal',");
                outJSON.println("          'value': '" + dbtime2 + "'");
                outJSON.println("        },");
                outJSON.println("    ");

                outJSON.println("        'longitude': {");
                outJSON.println("          'type': 'literal',");
                outJSON.println("          'value': '" + dbtime3 + "'");
                outJSON.println("        }");
                outJSON.println("      },");
                outJSON.println("    ");
          }

    outRDF.println("</rdf:RDF>");
    outRDF.close();

    outJSON.println("      ]");
    outJSON.println("  }");
    outJSON.println("}");
    outJSON.close();

    con.close();

    }
```

b)

Figure 64 Code to Create RDF and JSON Models

In this example, location data is converted into its corresponding RDF and JSON models. Figure 64 a) demonstrates connecting to the Raw Data Store and creating the models. Figure 64 b) illustrates retrieving each data item from the SQL query and writing this data into the models.

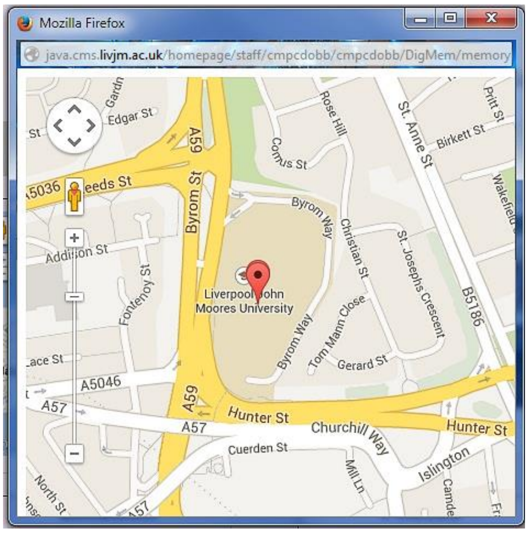
Using the newly created JSON file, the information is exhibited, as a memory box. For example, in the case of plotting GPS coordinates in the memory box, the following piece of code, in Figure 65, has been developed to illustrate the user's movements. The other JSON models (photograph, accelerometer and heart rate) are displayed using the same format of loading their JSON data into an *AJAX* function.

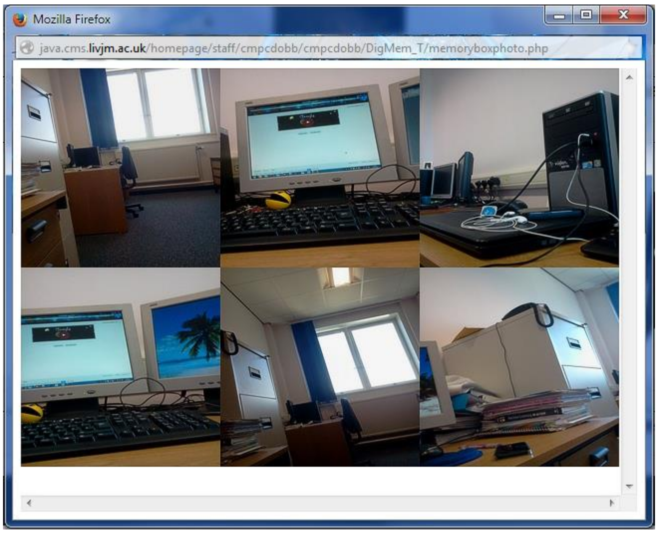
```
$.ajax(
{
     dataType: "json",
     url: queryUrl,
     success: function(data)
         {
           drawMap(data);
         }
});

function drawMap(data){

      var LatLon = new google.maps.LatLng(53.396886, -2.961094);

      var myOptions = {
           zoom: 13,
           center: LatLon,
           mapTypeId: google.maps.MapTypeId.ROADMAP
      };

      var map = new google.maps.Map(document.getElementById("map_canvas"), myOptions);

      //add a marker for each position
      var bindings = data.results.bindings;

      for (var i in bindings)
      {

           var memoryData = data.results.bindings[i];

           var latitude = memoryData["latitude"]["value"];

           var longitude = memoryData["longitude"]["value"];

           var time = memoryData["time"]["value"];

           // make a marker with the label as the time
           var marker = new google.maps.Marker(
           {
              position: new google.maps.LatLng( latitude, longitude ),
              map: map,
              title:time
           });
      }
}
```

Figure 65 JavaScript Function to Display GPS Memory Box Data

Memory boxes illustrate the user's interactions, movements and physiological signals. Based on a series of questions, the system clusters data and transforms a huge amount of data into a succinct piece of information (memory box), as demonstrated in Figure 66. An example memory box is depicted in Figure 64 a). In this figure, each of DigMem's inputs (Location, Photographs, Acceleration and Heart Beat) is displayed, as well as examples of other potential input devices that can provide information. When the user clicks on an input, a separate window opens to reveal the associated data. As demonstrated in see Figure 66 b), clicking on the "Location" box reveals the user's location at that particular time. Figure 66 c) reveals all of the associated photographs, whilst Figure 66 d) illustrates the corresponding accelerometer data, and Figure 66 e) the associated heart rate information.

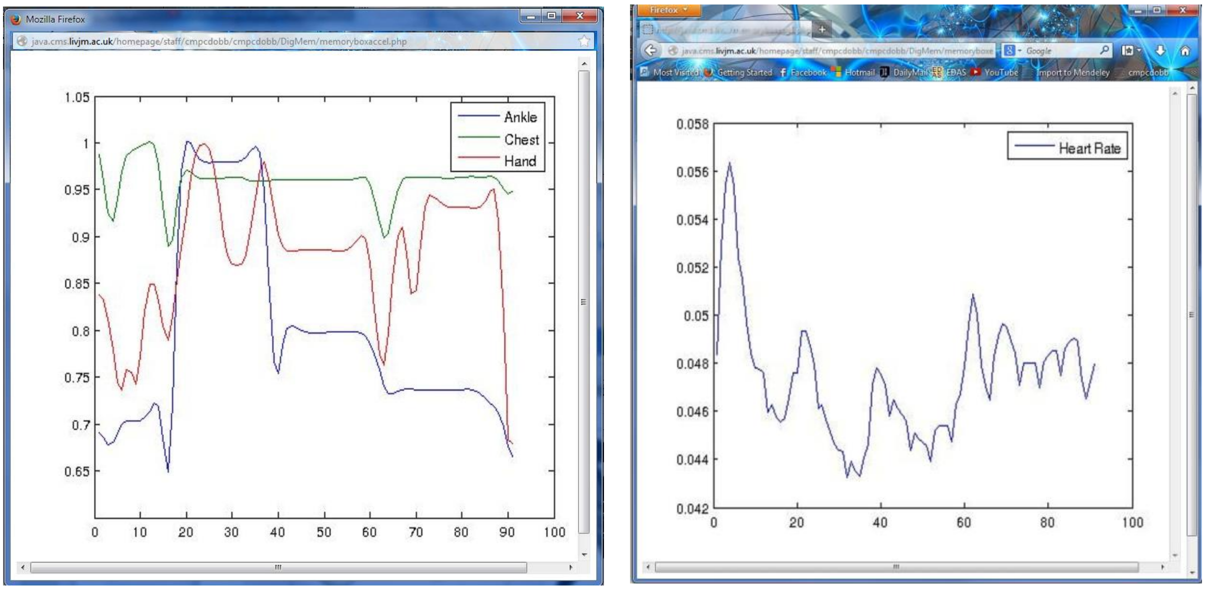
a)





b)

c)



d) e)

Figure 66 Memory Box

The system revolutionises the way in which users are able to interact and understand their data. From a massive amount of data, key points can be pinpointed and visualised. Users are able to see the times that they have been, for example, the happiest or most active. A greater understanding about ourselves will begin to emerge.

As well as creating memory boxes, DigMem is also able to answer specific questions about the user's data. In this instance, the user is presented with the following page, illustrated in Figure 67, below.

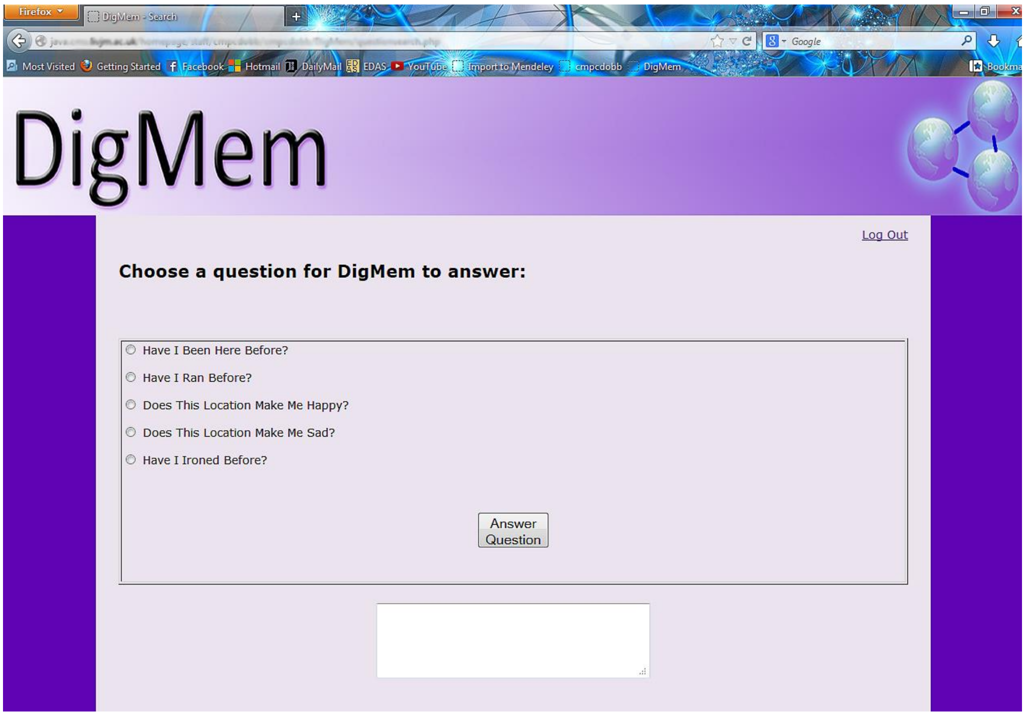


Figure 67 Life Questions Web Page

As with the Create Memory Box page, Figure 62, this page also displays a list of questions. However, in this instance, the searching of the Feature Set is treated as a classification problem. For example, in the case of selecting the second question, the system uses the decision tree (*TREEC*) classifier to classify the behaviour of the user, in order to return an answer. The system has been trained to recognise specific activities, by their features, and match the user's data with this previously trained set of activity data, thus enabling which question to be answered. The code to achieve this is illustrated in Figure 68.

```
x = 100; %number of times to loop

c = 0.80; %size of the training set - 80% training, 20% testing

for i = 1:x

labels = data(:,[5]); %Activity labels

B = data(:, 1:4); %Remove the last column in the 'data' matrix (activities)

a = dataset((B),labels);

[trainset, testset] = gendat(a,c);

truelabels = getlab(testset);

w7 = treec(trainset); %train the classifier

est_labs = testset*w7*labeld;

[matrix, NE] = confmat(truelabels,est_labs);

end
```

Figure 68 Classification Code

In this way, DigMem is able to answer specific questions about the user's data and is able to uncover things about ourselves that we might not realise. For instance, visiting particular locations may incite certain subconscious feelings. By asking the system, "Does this location make me happy?" physiological signals can be mapped to the user's location to determine if their previous visit provoked happiness. This is a very powerful aspect of the system, which makes it unique.

4.2 Prototype Configuration

In order to evaluate the Design, in chapter three, a prototype has been developed. This has been utilised to test a number of components of the system. Firstly, to test the MoDM middleware (Dobbins, Merabti, et al. 2012b), two tablets (a Samsung Galaxy[9] and Nexus 7[10]) have been used, which act as the sensor peers, whilst a

[9] http://www.samsung.com/uk/consumer/mobile-devices/tablets/tablets
[10] http://www.google.co.uk/nexus/7/

Toshiba[11] laptop has been used as the mobile peer and as a rendezvous peer. The following tests have been undertaken to assess the appropriateness of the design:

1. Ensure that the P2P network can be created
2. Ensure that all devices can connect to the P2P network
3. Ensure that the tablets can advertise their services
4. Ensure that the laptop can find these services
5. Once a service has been found, the command should automatically trigger data collection
6. Ensure that data has been automatically sent back to the laptop and saved in the Internet-connected folder
7. Ensure that physiological data can be recorded

These tests have been formulated to ensure that it is feasible to connect to external devices, for the purpose of data collection. The tests have been undertaken using the configuration setup depicted in Figure 69.

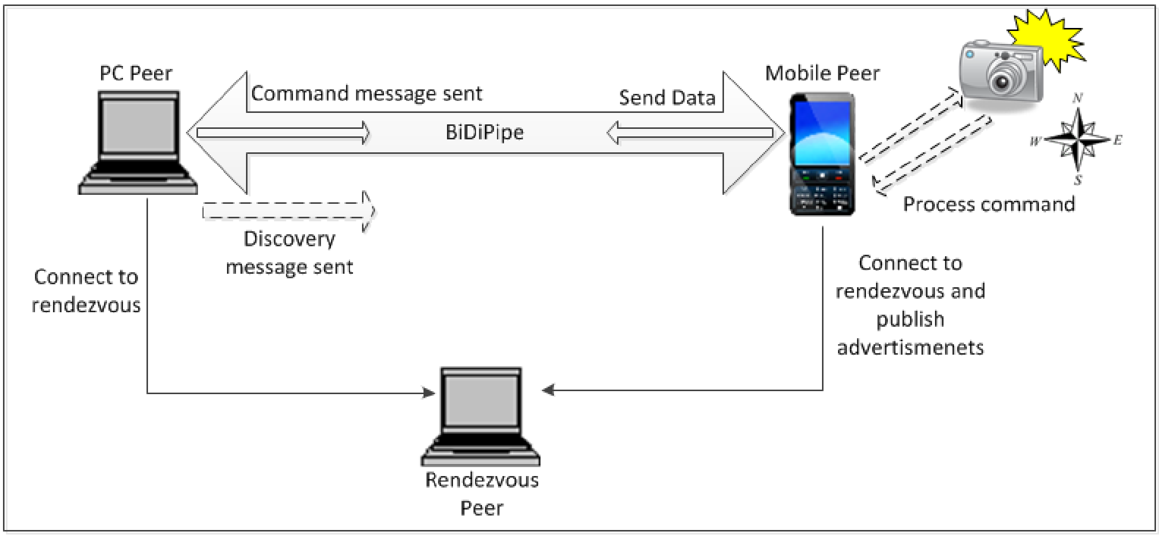


Figure 69 MoDM Service Tests

The first item to be tested has been the establishment of the P2P network. This has been successful, and the P2P network has been created. Upon creating the network, the next test ensures that the peers could connect to it. Upon successfully connecting to the network, the sensor peers were successful in advertising their services to the network. Once this was established, the menu screen, on the mobile peer, needed to be assessed. This test ensured that, upon selecting a service, the appropriate discovery messages have been propagated to all peers on the network. In this test, the Galaxy tablet had the photograph service, whilst the Nexus 7 had the GPS service. After verifying this, the next item to be tested was that the peer with the specified service could be found and that the service could be automatically triggered, from the command. The final step was to ensure that data has

[11] http://www.toshiba.co.uk/

been sent back to the mobile peer, and saved in the user's Dropbox folder. In these tests, the mobile peer was successful in finding both services, triggering them and retrieving data. To test the physiological services, a NeXus-10[12] heart rate sensor and Actigraph[13] accelerometer have been used. Using these devices, tests have been undertaken to ensure that data has been successfully recorded whilst various activities were undertaken.

After testing MoDM, the DigMem Server (Dobbins, Merabti, Fergus, Llewellyn-Jones, et al. 2013) aspect has also been tested. The following tests have been undertaken to assess the appropriateness of the design:

1. Ensure that the location and physiological application services can transfer raw location, accelerometer and heart rate data from cloud storage to the Raw Data Store
2. Ensure that the photo application service can transfer raw photographic data from cloud storage to the memory archive and then into the Raw Data Store
3. Ensure that the unprocessed raw data can be converted into a dataset
4. Ensure that after the data, from test three, has been inserted into the dataset that this previously unprocessed data is now flagged as being processed, in the Raw Data Store
5. Ensure that features can be extracted from the raw dataset so that the Feature Set can be created

These tests have been formulated to ensure that data can be successfully transferred from cloud storage, processed and that the resulting feature set is created. The first two test items ensure that the developed application services can transfer raw data into the Raw Data Store. These tests were successful, and the data has been transferred. Tests three and four ensure that unprocessed data, in the database, can be selected and that a dataset (Raw Dataset) can be created from this information. The tests also ensure that this data is now flagged, in the database, as being processed. This ensures that, when new data is appended to the dataset, that previously processed information is not duplicated. The final test, which was also successful, ensures that features can be extracted from the raw data. This is essential so that machine learning algorithms can be applied to the data.

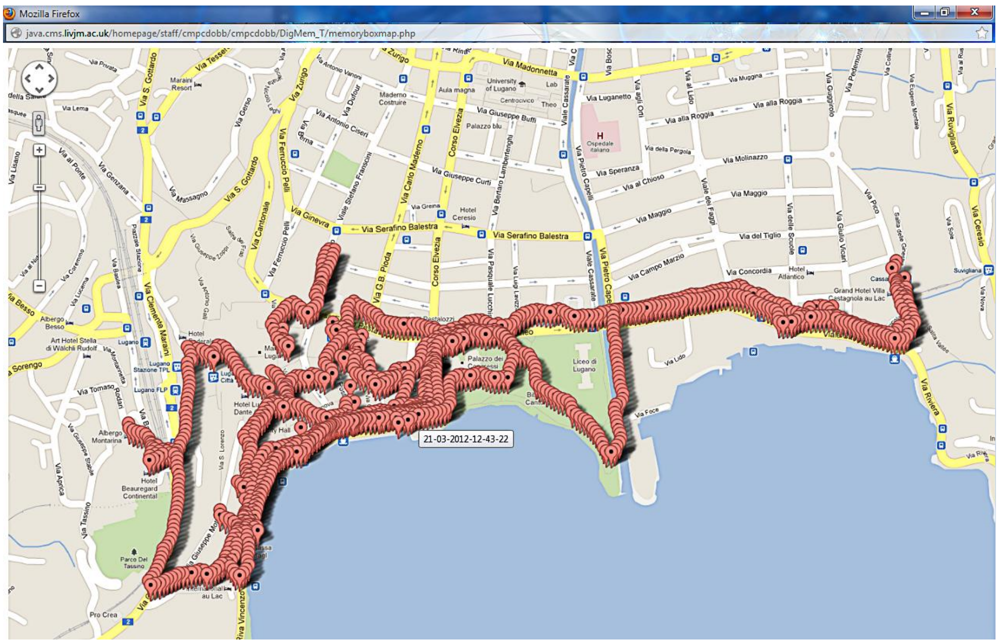
The DigMem Web application (Dobbins, Merabti, Fergus, Llewellyn-Jones, et al. 2013) was the final portion of the system to be tested. The following tests have been undertaken to assess the appropriateness of the design:

1. Ensure that the user can register with the DigMem website
2. Ensure that the Registration page validation prevents an incomplete form from being submitted
3. Ensure that the Registration page validation prevents the user from registering with an existing username

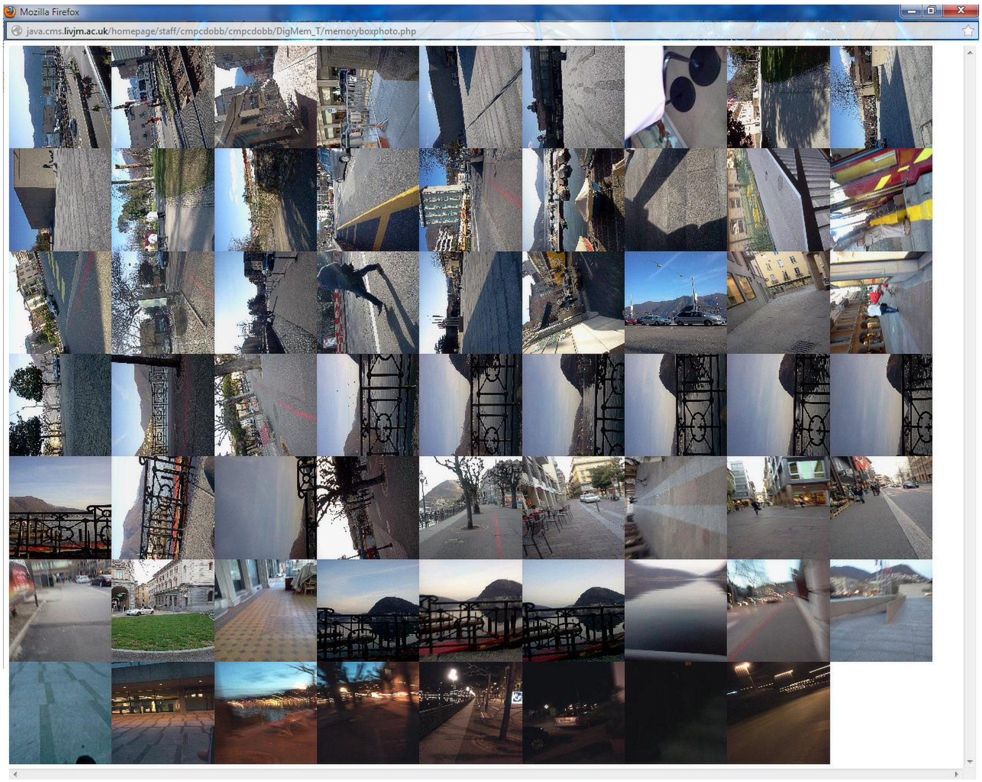
[12] http://stens-biofeedback.com/collections/computerized-biofeedback-systems
[13] http://www.actigraphcorp.com/

4. Ensure that the Registration page validation prevents the user from registering with an existing email address
5. Ensure that the user can log into the DigMem website
6. Ensure that the Login page validation ensures that the correct username and password have been entered
7. Ensure that the Login page validation ensures that the username and password fields are not blank
8. Ensure that the *k*-means clustering algorithm correctly clusters the data
9. Ensure that a memory box can be created
10. Ensure that the *TREEC* classification algorithms can successfully classify activity so that life questions can be answered

These tests have been formulated to ensure that the user can register and log into the DigMem web application and that their data be cluster and classified correctly. The first four test items have been successful in ensuring that the user can register with DigMem and that they have not submitted incomplete or incorrect registration details. Tests five to seven have also been successful and ensure that the user can log into the system. These tests also prevent incomplete or incomplete details from being submitted. In test eight the question, “When have I been active?” has been used to test that the clustering algorithm can successfully cluster periods of high energy. The next item tested has been the creation of a memory box. Over 12-days, data has been collected to document the user’s activities. A memory box of a particular day (see Figure 70) has then been created to test this part of the system. Figure 70 a) illustrates the location data, whilst Figure 70 b) displays the photos that were collected.



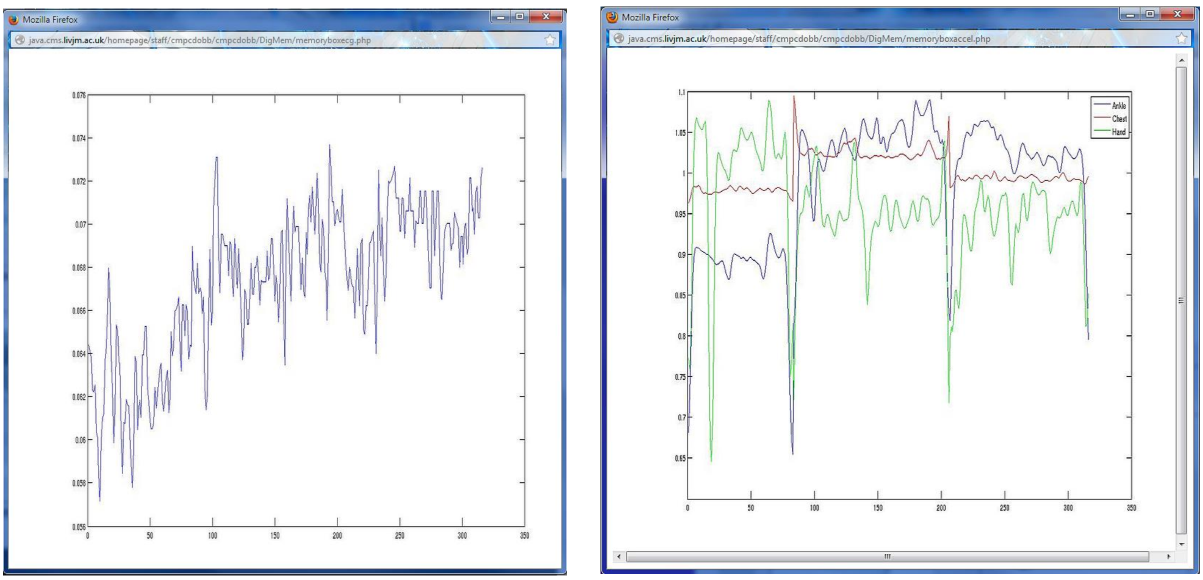
a) Location Information



b) Photograph Data

Figure 70 DigMem Memory Box Data

Drilling deeper into this memory box utilizing accelerometer and heart rate information (see Figure 71) activities begin to emerge. Using all of this information it is possible to understand and see exactly what our bodies were going through at the time. In this instance, the user was walking around, which has been validated by the memory box (see Figure 70 and Figure 71).



a) Heart Beat

b) Acceleration

Figure 71 DigMem Memory Box Physiological Data

Obtaining and linking data from a variety of sources provides a greater level of detail in the creation of human digital memories. This test has successfully demonstrated how disjointed pieces of information can be brought together to form a memory box of a particular time. These boxes can pinpoint activities and help us to relive any moment of our life. As well as this aspect, memory boxes also help us to understand and see how our behaviours have changed over time and, in one instance, can be used to identify sedentary behaviour. By providing, an outlet that highlights the amount of time spent in periods of inactivity is the evidence that is needed to change behaviour.

Test ten ensures that behaviour can be classified so that life questions can be answered. In this case, the *TREEC* classifier has been tested to determine that the question, "Have I been running?" can be answered. Table 1, below, illustrates the resulting confusion matrix from this test, which shows the how the algorithm performed in terms of predicting the correct activity. In this instance, the algorithm correctly identified lying down eleven times, whilst standing was correctly identified ten times and was misclassified once as walking up stairs.

True Labels	**Estimated Labels**								
	Lying Down	Sitting	Standing	Walking	Running	Walking Up Stairs	Walking Down Stairs	Vacuuming	Ironing
Lying Down	11								
Sitting		11							
Standing			10			1			
Walking				11					
Running					11				
Walking Up Stairs						10			1
Walking Down Stairs							11		
Vacuuming				1				10	
Ironing							1		10

Table 1 *TREEC* Confusion Matrix Results

As it can be seen, the classifier has been 100% successful in recognizing the majority of activities, with only four errors occurring. In this instance, the system has been able to recognise that the user has run before and thus has successfully answered the question. As the system learns more about the user, more probing questions will begin to emerge. For instance, emotional features can be used to determine if certain locations or people make us happy. This has profound implications and provokes a deeper understanding and reflection of ourselves. Coupled with memory boxes, not only are we able to relive any moment in our lives but a greater insight into our behaviours, thoughts and feelings will begin to emerge.

4.3 Summary

This chapter briefly describes the implementation of the various aspects of the DigMem system. More specifically, it demonstrates how data is obtained and transformed into a temporal memory box of human experiences and illustrates how life questions can be answered. This chapter has demonstrated the idea that human digital memories can be created using pervasive devices, linked data and machine learning. The main aim of this chapter has been to address the research challenges, outlined in chapter one, and to ensure that the design specification, outlined in chapter three, has also been met.

As such, the implementation is not a seamless process. Nevertheless, in order to demonstrate the idea, the implementation has been successful. Compared to other systems, this method offers a much broader range of information that can be retrieved and enables the system to analyse and answer questions about the user. Other systems, such as Microsoft's *MyLifeBits* (Gemmell et al. 2002), are considered to be one-dimensional with searching being based on keywords. The DigMem system enables users to explore their data without defining specific keywords or needing a pre-existing knowledge of the data, to create queries. By clustering the data, all of the related information can be extracted from the dataset. The user does not have to know what they are looking for (i.e. by explicitly defining queries). Various pieces of information can be brought together so that a greater level of detail can be achieved. The system also does not rely on particular pieces of hardware to create a memory. Any pervasive device that can provide information, and running the MoDM middleware, can contribute to the memory. This plug-and-play platform enables the system to maintain flexibility and allows memories to be created across many different environments.

The following chapter presents a case study, which has been devised to demonstrate the system in a real-life scenario. As such, it illustrates the collection of information in a smart home environment, the construction of memory boxes and how life questions can be answered, for self-reflection. A discussion is also presented, which describes the other application areas that DigMem can be useful in, including preventing sedentary behaviour, activity recognition and assisting with cognitive illnesses. This chapter provides context for the previously discussed design (chapter three) and implementation (chapter four).

Chapter 5

Case Study

In the previous chapter, the implementation of the system has been presented and the configuration of the prototype discussed. In this chapter, a case study is presented, which has been devised to provide context for the design (chapter three) and implementation (chapter four). A scenario is considered that discusses how memory data is collected in a pervasive environment, the construction of memory boxes and how life questions can be answered, for self-reflection. Furthermore, other application areas that the system can be used for are described, which illustrate DigMem's diversity in many areas outside of self-reflection. These include discussing activity and emotional recognition and how the system can be used to prevent sedentary behaviour and assist with cognitive illnesses.

5.1 Collecting and Transforming Data into a Human Digital Memory

This case study demonstrates how the DigMem system automatically collects data in a smart home and transforms this information into a human digital memory. In this instance, a smart home is defined as an environment where appliances are integrated with sensors to collect a range of information. This data is then processed and turned into a memory box or used to answer questions about the user. Figure 72 illustrates an example of such an environment. In this example, sensors in the bed monitor sleeping patterns and the duration of time spent in bed, whilst televisions store viewing information and their duration of use. Electronically tagged food items, in the fridge, determine food consumption and sensors in sofas and chairs determine sitting duration. In addition, every room has temperature and location sensors to record environment information and the user's position within the house. As well as sensors in the environment, body sensors also record a range of physiological data, including acceleration and heart rate. To enable each device-specific service to become available for data collection, each device would have to implement the MoDM middleware (Dobbins, Merabti, et al. 2012b). When the user enters the home, they enable their MoDM-compliant mobile device to send out a broadcast to look for information. This data can either come from local services (i.e. on the user's own mobile device) or from remote services (i.e. from the sensor devices located throughout the home).

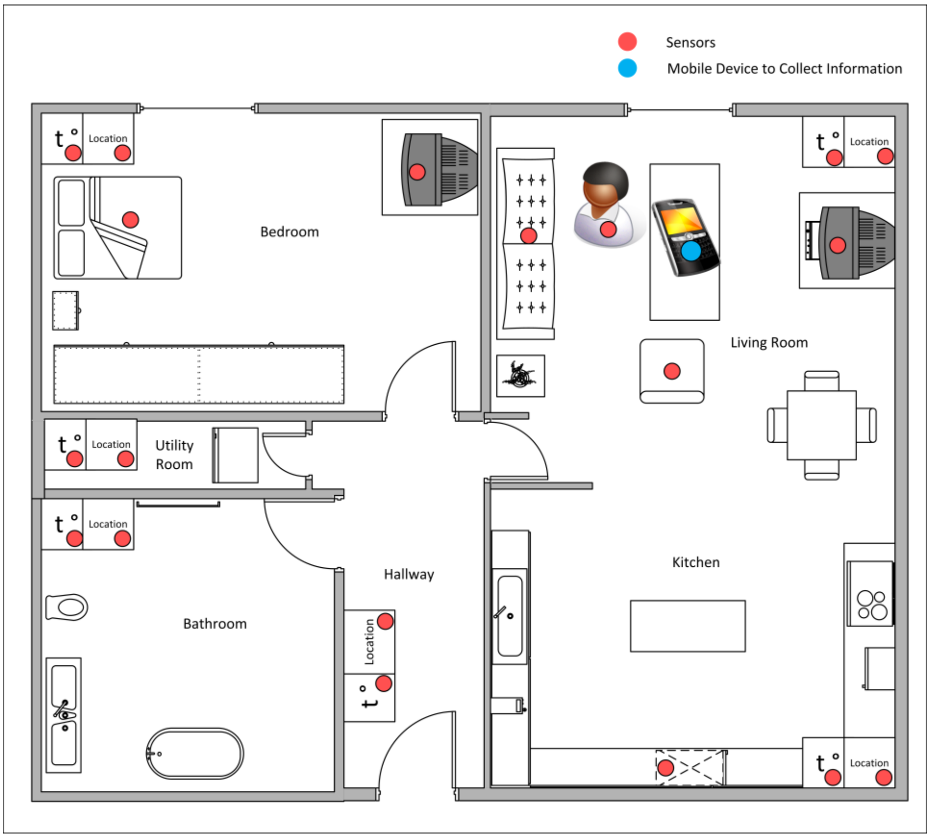


Figure 72 Example of How DigMem Collects Information in a Smart Home Environment

If local services cannot be found, an ad-hoc P2P network is created. This network connects the sensor devices to the user's device so that data can be collected from these external sources (see Figure 73). Upon connecting to the network, the user's device requests information from each device. These devices respond, connect to the user's device and send data. Once data has been received, it is automatically stored and transferred to the DigMem Server, where it is processed.

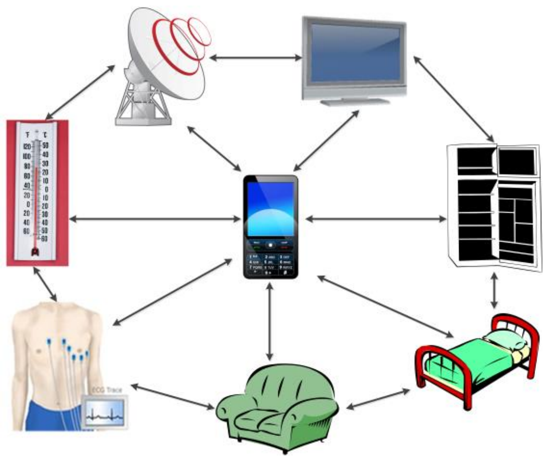
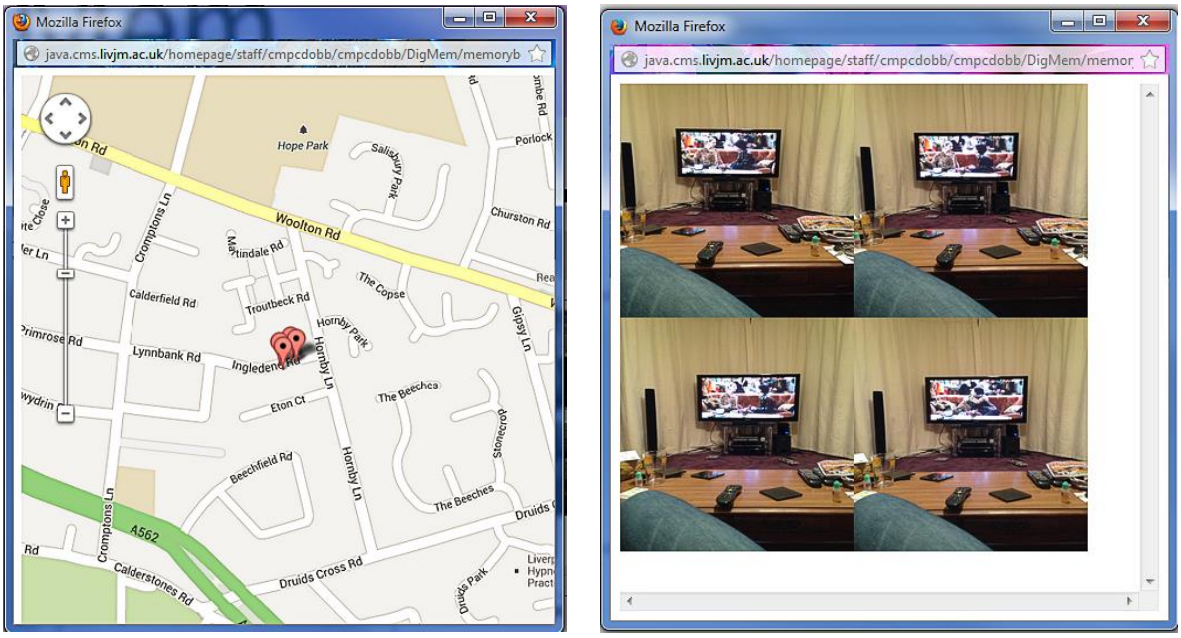


Figure 73 MoDM P2P Network

Using the setup from Figure 72 and Figure 73, Max has been recording data for a few years. Data from his home records information about his eating, sleeping and entertainment patterns. Using a wearable camera and body sensors, he also constantly records physiological and photographic data. As a new year approaches, Max is feeling reflective and decides to ask DigMem about his activity from the past year. Upon logging into the DigMem web application, Max can either create a memory box or ask life questions about his data. Each method provides him with a set of pre-defined questions. Firstly, he decides to ask the system a life question and selects the question, "Did I watch television often this year?" Using this query, DigMem classifies the activity of "watching television" and retrieves all of the instances of this activity. The results determine that, over the current year, Max has spent a considerable amount of time doing this activity. He then asks the same question again, however, he chooses to query the year before. He then compares the results and can see that he has spent more time this year watching television. However, he does not know the context of those times so he decides to create a memory box. Using a different set of pre-defined questions he selects, "When have I been inactive this year?" Using this question, the system clusters Max's information together so that he can see periods of low energy expenditure. Memory boxes are formed and each data type that DigMem supports is displayed. When an input is clicked on, a separate window opens, and a more in-depth illustration of the data is seen. Max can now see exactly what he was doing, where he was, and his body-related data. This information is presented to him in the form of photographs, his movements plotted on a map and his physiological information in a graph. Again, he decides to compare this year from the previous year to determine why he has become more inactive. From this comparison, Max has determined that during the previous year he spent less time at home, his diet was healthier and he was exercising more. However, this year he suffered an injury, which resulted in him spending more time being inactive. This led to a poor diet and an increase amount of time on the couch, watching television. The life questions enabled Max to reflect and determine that more time was spent this year doing this activity. Max did not realise that his injury had affected his lifestyle so much. However, the memory boxes enabled him to see that, due to his injury, his diet suffered and he was not getting out of the house as often as he used to. Figure 74, below, demonstrates an example of such a memory box. Each input (Location, Photographs, Acceleration and Heart Beat), and examples of potential other input devices that can provide information are displayed in the memory box.

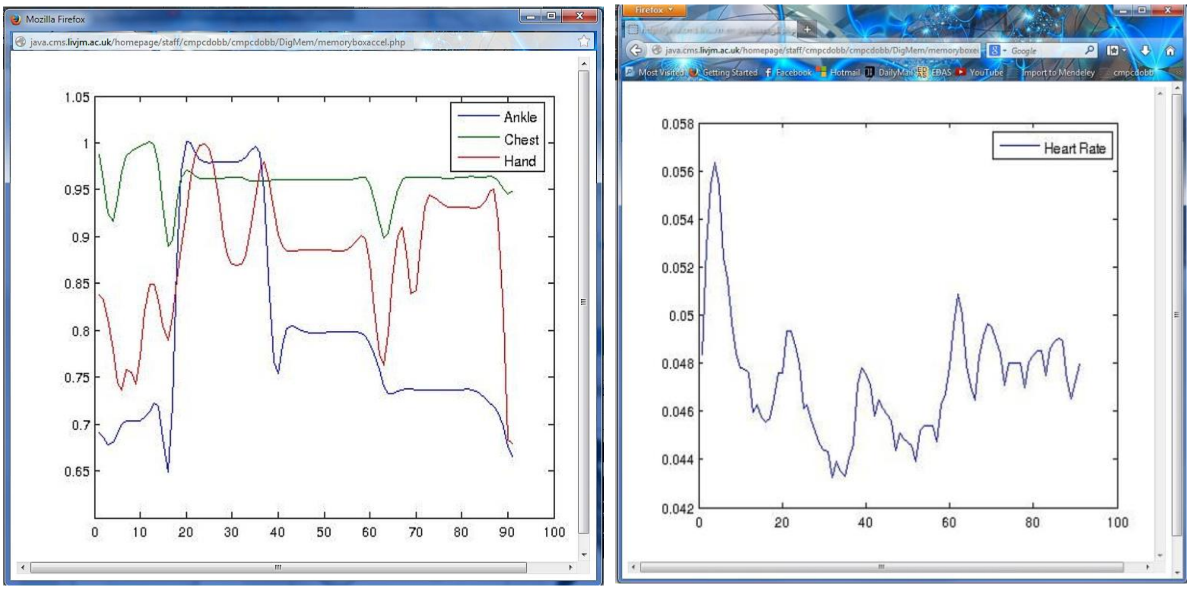
Figure 74 DigMem Memory Box

When Max selects an input, a separate window opens, and a more in-depth illustration of the data is seen. Figure 75 a) illustrates the user's location, whilst Figure 75 b) displays any collected photographs. Figure 75 c) and d) illustrates the user's acceleration and heart rate data. It should also be noted that this demonstration was only undertaken for approximately three minutes, hence the limited number of photograph and location data.



a) GPS Data

b) Photograph Data



c) Acceleration Data

d) Heart Rate Data

Figure 75 DigMem Memory Box Data

As can be seen, the results from the various input devices are now visually displayed. The memory box illustrates that the user was sitting down, watching television. The low frequency of their accelerometer and heart rate data corroborates this data. The system is able to reduce an exceptionally big dataset into a graphical display that is easier for the user to understand. This is extremely exciting and, using these boxes, any time of our lives can be reconstructed and relived.

Taking the scenario further, when Max leaves his home and goes into the city centre, a different number of services will be utilized for this memory. His mobile device can be used to capture photographic and location data, whilst also seeking out devices that are prevalent within this setting, such as other cameras, his friends' devices and building information. The two memory boxes that have been created, in the home and city, are different and reflect Max's surroundings at the time. In this sense, we are able to create a more detailed memory of an event, in a range of environments, not just the home, providing DigMem compliant devices are available. The system is adaptive enough to actively seek out devices and their services and incorporate this data into our memories. As such, a memory box is only as rich as the number of data sources that are present and this number may increase and decrease over time. The system is able to create individual and specific memories and to answer questions about the user, which are features that have not been seen before, in this area. Our human digital memories evolve with the individual. As smart environments slowly become a reality, and with all of this data available, harnessing it into a human digital memory presents us with a unique opportunity.

Human digital memories created in this way offer a new insight into the composition of a digital memory and how data can be reasoned over. Any MoDM compliant device, embedded with a sensor, is capable of being included in the memory. This is very important, and is what makes the system flexible enough to be used over an extensive period of time. The system does not rely on specific devices, which enables DigMem to be adaptive and to evolve with technology. Memories are created from data that has been accumulated over a lifetime and as our human memories develop and grow so will their digital counterpart. Furthermore, the system also allows the user to question their data. Questions such as, “Does this location make me happy?” or “Do I run a lot?” can begin to be answered. In addition, this very interesting feature can be used to gain a greater insight into ourselves. For instance, a particular location might provoke certain physiological responses, which we are unaware of having. By recording and questioning this information, a greater understanding about ourselves is gained. Interaction with our memories is fundamental and is what makes the work unique. By enabling users to be able to “go into” their memories and to see and understand various pieces of information, such as temperature, location and emotion, could lead to the augmentation of group memories and has the advantage of benefiting numerous aspects of people’s lives. Whether it enhances social groups and interactions, aids in the health and recovery of memory–related illness or is used to reduce sedentary behaviour, the possibilities are endless.

5.2 Other Application Areas

DigMem is a flexible system that is capable of creating human digital memories in a variety of environments. It can also be used within a wide variety of areas, for many purposes. So far, the system has been used to document daily life, whilst earlier deployments focused on tracking journeys that the user had made. The purpose of these case studies was to create human digital memories for the purpose of reflecting upon past events. However, human digital memories can be used for so much more than this. This section details how the system can be used within other application areas.

5.2.1 Preventing Sedentary Behaviour

There is growing global concern over the growing levels of obesity and the fact that people in general are not as active as they once were. Many believe that this is directly related to an unhealthy diet and an increasing reliance on technology, such as television, social networking, computer games, and voice activated home control systems. These kinds of activities increase sedentary behaviour, across all age groups, and are considered to be one of the main contributors to obesity and poor health.

Sedentary behaviour is identified as a class of behaviours characterized primarily by sitting, with associated low levels of metabolic energy expenditure (Owen et al. 2011). In other words, it relates to the amount of time we spend doing non-physical activities, such as sitting, lying down and watching television. This type of behaviour can be associated with excess weight gain and an increased risk of other diseases in later life, such as metabolic disorders and memory loss (Clark et al. 2009). Whilst this behaviour does not seem particularly dangerous in our younger years the effect of little or no exercise, poor diet and alcohol misuse, becomes apparent, as we get older. The premature worsening of an individual's cognitive and physical capabilities are something that can occur over decades, as opposed to being evident over a number of days, weeks or months. This significantly impacts on the ability to live better during older age (Strayer et al. 2011). As the life expectancy of adults in Great Britain is increasing so is the occurrence of illness and disability (Office for National Statistics 2010). As a result, research into sedentary behaviour is growing rapidly, with early results indicating a potentially important negative health outcome for various markers of this type of behaviour (Biddle et al. 2010). For this reason, decreasing this type of behaviour is considered a crucial theme within many research programs in health. Ironically, there is general agreement that the use of technology is likely to help researchers understand this type of behaviour. One interesting approach is based upon the use of human digital memories to provide visual lifelogs of a user's activity and to identify the behaviour patterns of individuals. In this way, lifelogs provide a way for users to evaluate their lifestyle choices.

As we reach our later years, a lifetime of bad habits is hard to change. As a result, the elderly are particularly at risk for developing such afflictions that are associated with this behaviour. In particular, one side effect of being sedentary for too long is the development of pressure ulcers. Pressure ulcers are formed when sustained pressure is placed on a particular part of the body and interrupts the blood supply, eventually leading to infection (NHS 2012). As well as being extremely painful, pressure ulcers can cause severe social and financial consequences for individuals, health services and the community (Wang et al. 2011). Extended hospital stays or extra nursing care is often needed, resulting in severely increasing the healing time and cost of care. Dealing with pressure ulcers is estimated to cost 4% of the total NHS expenditure (between £1.4 and £2.1 billion annually) (Bennett et al. 2004). In America, this figure is even higher, with the annual cost reaching $11 billion (Agency for Healthcare Research and Quality (AHRQ) 2012). However, in most cases they are preventable with the correct amount of patient assessment and monitoring. Nevertheless, given that the workload placed on nurses, in the United Kingdom, has increased (International Council of Nurses (ICN) & Pfizer Inc. 2009), dedicating a set amount of one–on–one time is not feasible.

However, the use of human digital memories in preventing this behaviour is seen as an ideal platform to monitor a patient's health and wellbeing, over a significant amount of time. Monitoring and measuring sedentary behaviour, with the aid of human digital memories, allows a visual illustration of a user's habits and state of their health to be observed, at any time. It also allows a continuous flow of information about the patient to be collected and used to help determine the impact that their lifestyle choices are having on their health. Therefore, by providing this instant gratification of how positive changes can affect their health, patients would be more inclined to change their behaviour. The use of the DigMem system in preventing sedentary behaviour has been explored, during the research (Dobbins, Fergus, et al. 2012a; Dobbins et al. 2013; Dobbins, Fergus, et al. 2012b; Fergus, Attwood, et al. 2012). More specifically, reducing the onset of pressure ulcers has been investigated (Dobbins, Fergus, et al. 2012b; Dobbins et al. 2013).

The elderly are particularly susceptible to this condition, due to their increase in sedentary behaviour. Patients are often bedbound and unable to move their body around as freely and frequently as required. The following scenario focuses on how prolonged periods of sedentary behaviour, which can lead to the development of these ulcers, can be identified. As it can be seen in earlier research, monitoring and preventing sedentary behaviour is considered essential to improve a patient's quality of life, particularly as they grow older (Dobbins, Fergus, et al. 2012a). Previous research has indicated that using lifelogs, or digital memories, presents an accurate representation of sedentary behaviour. These logs provide evidence of periods of inactivity that would otherwise get overlooked or over–reported, by an individual (Kelly et al. 2011).

In the following scenario, an elderly patient has just suffered a stroke and is recovering in hospital. They are unable to move around freely; the ward is particularly busy and understaffed. Sensors attached to their body are monitoring any movements and physiological data, whilst static sensors attached to the bed monitor the pressure being exerted, on particular, pressure point hotspots. A camera is hanging around their neck, which is capturing photos every few minutes, and there are sensors in the room that are monitoring the state of the room (i.e. temperature, humidity, etc.). Next to their bed is a laptop that is broadcasting messages for the retrieval of services, from these devices prevalent within their environment (all devices would also be MoDM compatible (Dobbins, Merabti, et al. 2012b)). The data collected, from the services and the sensors, is serialised as RDF triples and saved. From this location, the information is extracted every few minutes and stored, within the DigMem application. This data is now available worldwide to anyone that is affiliated with the patient's account; machine learning algorithms can be run on this data and memory boxes created.

Reiterating back to the current scenario, it is the weekend and the patient's doctor is out of the country on a short break. However, with the use of this system, their doctor can log onto DigMem and perform a query on the patient's memory data. They can see how the patient was feeling and what happened while they are away.

For example, the doctor can ask, “How has Joe been feeling during the weekend; how much has he moved around and how often was he checked on by staff?” The use of RDF, linked data and machine learning allows this query to be successfully executed. Data can be clustered around the time that the doctor was away and extracted into a memory box. It can be established that during the weekend, Joe was not feeling well, his heart rate was high and a lot of pressure was exerted on a particular area of his body, for a long time. It can also be determined, through photos taken on the same day that he was only checked on twice. The doctor can now see exactly what Joe was feeling, what he was doing and what is happening around him, even from another country. The doctor can then phone the ward and alert the staff to the fact that Joe has been in a sedentary position for too long, and that he should be moved around to stimulate blood flow, in order to prevent pressure sores developing.

The DigMem system offers a new way to, as unobtrusively as possible, monitor a user for potentially their entire lives. By providing visual evidence that a user has been spending too much time in sedentary positions it is hoped that this is the incentive that they will need to change their behaviour. The system is also able to provide information to medical practitioners and individuals about themselves and their behaviour patterns. This information can then be used to implement compensatory changes and view the impact this has on specific medical outcomes. Obtaining and linking data from a variety of sources provides a greater level of detail in the creation of human digital memories. Adding as much detail as possible enables the execution of smart queries, which have the ability to search data in a multi–dimensional fashion. Human memories are infinite, and their digital counterparts should not be any different. The system can track our entire lives and monitor a patient for any amount of time; something that current systems are unable to do.

5.2.2 Activity and Emotional Recognition

Whilst the previous scenario focused on preventing sedentary behaviour, the system can also be used for monitoring activity. For example, data can be clustered based on times of life that the user was most physically active. Factors that might have contributed to a decrease in physical activity can then be established. In the following scenario, the user has been recording location, photographs, physiological signals, weight and food intake data since they were 20. Now into their 30s, they have taken a greater interest in their health. When querying their human digital memory data, a trend of activity has begun to emerge. By clustering their data into age groups has illustrated that from 20 to 25, they participated in sports activities and biked to work. However, from 25 to 30, they have reduced their participation in sports and between the ages of 30-35, they now drive to work, have stopped their sporting activities altogether, are spending more time at work and have gained a significant amount of weight. By analysing their human digital memory data, collected over this period of time, the user is able to see how their level of activity has changed, over the years, and the negative impact that this is having on their health.

Emotional states can also be determined, such as stress or happiness. Queries such as, “When was I the most stressed?” can soon begin to be answered. For instance, by linking data together, such as heart rate, photographs and location, it can be determined that during examination revision or moving home, the user was very stressed. Factors contributing to this state will begin to emerge, and emotions can begin to be inferred. This is particularly important because, as observed in (Hodges et al. 2011), using physiological measures, in conjunction with images, can enhance the ability of brain-injured patients to develop awareness of, and therefore improve, emotional regulation. As stated in (Hodges et al. 2011), “*An improvement in emotional regulation is especially important to participation in everyday life after a brain injury; it may facilitate the return to social and professional activities thereby reducing the likelihood of contact with mental health services and the criminal justice system*”. Recognizing emotion is an important progression in lifelogging research. A richer recall of life experiences will enable self-reflection and positive lifestyle changes to occur.

5.2.3 Assistance with Cognitive Illnesses

As the life expectancy of adults is increasing, so is the occurrence of illness, disability and the demand on hospitals (Office for National Statistics 2010). Since the probability of becoming cognitively impaired increases with age (roughly 10% of over 65 years old), one side effect of increasing life expectancy is the emerging number of dementia patients (Huang et al. 2012). Dementia is a progressive and irreversible chronic disease, with varying degrees of stages (early, middle and late), that causes a mental deterioration, thus affecting the brain (Davies et al. 2009). People with dementia face a decline of their cognitive functions, including memory impairment, difficulties performing familiar tasks and diminished judgement (Vogt et al. 2012). It is a syndrome with different causes, of which Alzheimer’s disease (AD) is the most common type (Vogt et al. 2012). Worldwide, there is a new case of dementia every seven seconds (Alzheimer’s Research UK 2012). Currently, there are over 820,000 people living with dementia in the UK today and more than 35 million people worldwide (Alzheimer’s Research UK 2012). The financial impact on the economy is also quite severe. Dementia costs the UK £23 billion per year, which is twice that of cancer, three times the impact of heart disease and four times that of strokes (Alzheimer’s Research UK 2012). As a result, there is a very strong need to support ambient assistive living technologies, which promotes independent living, in the home (K. M. Lee et al. 2011). The strain placed on these health resources is apparent, and clearly, a user-centric approach is needed. As such, the area of dementia has been identified as one which would benefit from the introduction of innovative technological solutions (Davies et al. 2009).

Persons with dementia often rely on external memory aids, such as calendars, diaries, alarms, whiteboards, notebooks, and timers, to help them compensate for their memory deficits and to maintain an account of their daily life (Kikhia et al. 2010). As previously discussed, the area of human digital memories focuses on documenting the things we do, the places we visit and the thoughts we think (Dobbins et al. 2011). This outlet allows us to capture rich information about ourselves and our surrounding environment (Dobbins, Fergus, et al. 2012a). Content can be continually captured, which can be reviewed at a later time. A number of studies have been undertaken that explore how reviewing these lifelogs can aid in the recollection of events in patients with cognitive impairments, such as dementia and Alzheimer's (Davies et al. 2009; Karaman et al. 2011; Kikhia et al. 2010; Kikhia et al. 2009; Piasek et al. 2011; De Leo et al. 2011; Huang et al. 2012; Berry et al. 2007; Hallberg & Kikhia 2009; Loveday & Conway 2011).

The most popular method of capturing content, for use in such studies, has been with the use of the *SenseCam* (Hodges et al. 2011; Browne et al. 2011; Piasek et al. 2011; Doherty et al. 2012; Doherty, Moulin, et al. 2011; Crete-Nishihata et al. 2012; Hodges et al. 2006; Berry et al. 2007; Loveday & Conway 2011). In particular, the results from (Berry et al. 2007; Hodges et al. 2006; Hodges et al. 2011) illustrate that "*Through the use of SenseCam, a markedly amnesic patient was consistently able to remember aspects of several events. Recall was maintained almost a year after some of the events took place, and without any review of those events for up to three months*" (Hodges et al. 2006). Furthermore, the patient reported that "*Seeing the beginning of a clip brought memories "flooding back" without necessarily having to view further images*" (Berry et al. 2007). This suggests that she was remembering the event itself rather than the images, which was confirmed by her husband, who said that his wife was able to recall details of events not contained in the images (Berry et al. 2007). It also suggests that the visual images themselves provided a potent cue to recall (Berry et al. 2007).

Whilst the *SenseCam* has produced exciting results for the use of lifelogs in aiding autobiographical memory, there are limitations. As reported in (Berry et al. 2007), and as previously discussed, the time taken to upload, access and combine particular image segments is cumbersome. Furthermore, the absence of sound information and that the viewing software runs on a standard desktop or laptop PC, which may be unfamiliar or anxiety-provoking for some people, have also been identified as further limitations (Berry et al. 2007). However, the DigMem system is seen as a way to alleviate some of these limitations. Using the user's mobile device, content is automatically captured, stored and transferred to the DigMem Server. Very little user intervention is required and related data is brought together without the use of complex queries. Additionally, the results in (Berry et al. 2007; Hodges et al. 2006; Hodges et al. 2011) were produced by only reviewing images. In contrast, DigMem is able to structure a much more detailed human digital memory than previously seen. As well as photographic data, a variety of other content, such as physiological data, is able to be captured. The user is able to reflect on their activities and also to see how their bodies are reacting. This is beneficial as the more detailed a human digital memory is the more useful it is. In the case of elderly users, carers are able to work with them in order to recall any point in their life and to help them reflect and relive any moment in time. For example, User A (who has mild dementia) and her husband are visiting the Eiffel Tower. They have visited this location before, however, on this occasion User A cannot recall these previous times. In this instance she, or her husband, can log into the system and ask it, "Have I been here before, does this location make me happy?" Using a learning approach, the system would have learnt the features of being happy and mapped this to their current location to return an answer of "Yes". Furthermore, all of the times that they have visited this location can be clustered together and displayed in a memory box. She can now see all of the information that has been previously captured (for example, photographs, physiological signals, temperature, etc.). By enabling users to be able to "go into" their memories and to see and understand various pieces of information, such as temperature, location and emotions, enables a greater level of detail to be retrieved (Dobbins, Fergus, et al. 2012a; Dobbins, Merabti, et al. 2012a; Dobbins et al. 2011).

5.3 Summary

This chapter has posited a scenario that illustrates how DigMem collects data in a smart home and demonstrates how this data can be used to create a memory box or answer life questions, for self-reflection. This study validates the design, and idea, that human digital memories can be created using every day, pervasive devices and linked data. Other than recalling specific experiences, this chapter also examines how the system can be used in other areas. It illustrates how DigMem can be used to monitor and reduce sedentary behaviour as well as aiding autobiographical memory in patients with cognitive impairments, and how activity and emotional states can be recognised and inferred.

Memories created in this way offer a new insight into the composition of a human digital memory and how data can be reasoned over. The use of linked data enables any item, embedded with a sensor, to be capable of being included in the memory. Memories can be created from data that has been accumulated over a lifetime and as our human memories develop and grow so will their digital counterpart. Whether it enhances social groups and interactions, aids in the health and recovery of memory–related illness or used to reduce sedentary behaviour, the possibilities are endless.

The following chapter discusses the evaluation methodology that has been used to pre-process collected human digital memory information. As such, the chapter details how features have been extracted and selected, from this data. These features are essential so that the machine learning searching methods can be evaluated in chapter seven.

Chapter 6

Evaluation Methodology

In the previous chapter, a case study has been presented that demonstrated how DigMem collects data in a smart home. It then detailed how memory boxes or life questions could be answered, for self-reflection. Additionally, the chapter also examined how the system can be used in other areas, such as for the prevention of sedentary behaviour, assisting with cognitive illnesses and recognising activity and emotions. This chapter presents the evaluation methodology that has been devised so that features can be extracted from such collected data. Features are values that are used to represent raw data (for example, standard deviation illustrates the spread of data) and are fundamental for machine learning algorithms. Furthermore, processing raw data is very time-consuming and computationally expensive. Extracting features enables streams of raw information to be condensed into smaller files so that more data can be processed. Features are used in DigMem as part of the searching method. They are used to cluster data for memory boxes and classify data so that life questions can be answered. Following on from the feature extraction, the chapter then describes how the extracted features are analysed so that a subset can be used when clustering and classifying data. The chapter also describes the validation and data over-sampling techniques that have been used, as well as the classification and clustering algorithms that will be used to evaluate DigMem's searching methods.

The evaluation methodology requires a dataset of raw information so that features can be extracted and selected. In this instance, a sample of raw data containing location, heart rate and data from three tri-axil accelerometers, situated on the ankle, chest and hand, has been gathered. During the collection of this information, various activities have been undertaken, including lying down, sitting, standing, walking, running, ascending or descending stairs, vacuum cleaning and ironing. Each activity has been performed for approximately three minutes. These activities have been used to gather a diverse range of data because they include a mixture of high and low energy actions. This is essential so that questions such as, "Have I been running?" or "When have I been active" can be answered.

6.1 Pre-processing and Feature Extraction

Collecting raw sensor data produces a phenomenal amount of information, which is susceptible to noise. In particular accelerometer data, is sensitive to this type of interference (Liu et al. 2008). Therefore, pre-processing is essential in order to correctly characterize the physical activity of the user within a certain time frame (Figo et al. 2010). Figure 76, below, illustrates the steps that must be undertaken to achieve this. Steps two – five have been undertaken using Matlab[14].

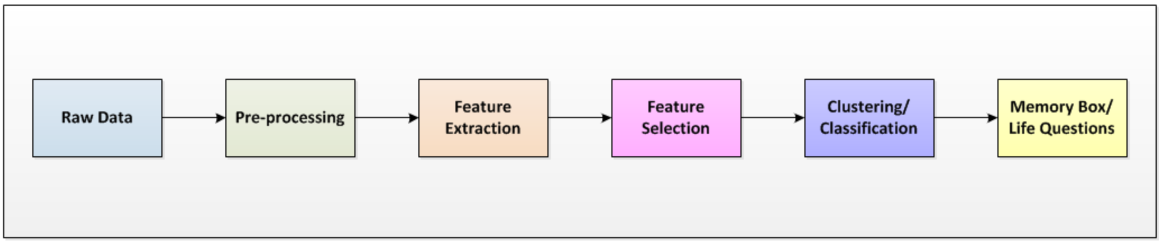


Figure 76 Data Pre-Processing Steps

When filtering accelerometer data it is necessary to specify a cut-off frequency so that any noise above that frequency is removed. As demonstrated in previous work (Mayagoitia et al. 2002; Lyons et al. 2005; Zhang et al. 2011), an appropriate value to use is 3 Hz. At this frequency noise is removed, without losing information. Following this approach, the accelerometer data has been filtered, using a second-order, forward-backward, digital low-pass Butterworth filter, with a cut-off frequency of 3 Hz, and normalised. In addition to filtering the signal, the size of the dataset needs to be reduced, without losing any information. In order to reduce the size of the data streams, a sliding average window has been applied to the data. The literature illustrates that to achieve optimal results, whilst ensuring that different activities can still be recognised, the appropriate number of samples to select is 512, with a 50% overlap (256 samples) (Reiss & Stricker 2012; Bao & Intille 2004; Song et al. 2012; Ravi et al. 2005; Mannini & Sabatini 2010; Krishnan & Panchanathan 2008). Using this approach, the average of the first 512 samples, of the signal, are calculated and stored. Then, using the last 256 samples (50%) of the first batch and the next 256 samples (50%) the next average is calculated and stored. This process is repeated until the entire signal is processed and is a method of processing the signal without losing information. Each tri-axil accelerometer provides an acceleration vector of A_x, A_y and A_z. In order to process this data as a single signal, Pythagorean Theorem (see equation 1) has been used to create a single acceleration vector, a. Acceleration data is recorded as negative numbers. By squaring the axis values ensures that a valid value is returned.

$$a = \sqrt{Ax^2 + Ay^2 + Az^2} \quad (1)$$

[14] http://www.mathworks.co.uk/

As previously stated, features are required so that the machine learning algorithms can classify and cluster data. In signal processing, various features can be extracted and the signal can be processed in a variety of domains, such as the *time* domain and the *frequency* domain. In these instances, features are extracted in order to obtain information from the signals. In the *time* domain, simple mathematical and statistical metrics can be used to extract basic signal information from raw sensor data (Figo et al. 2010). Frequency-domain techniques have also been extensively used to capture the repetitive nature of a sensor signal. This repetition often correlates to the periodic nature of a specific activity such as walking or running (Figo et al. 2010). The advantage of frequency-related parameters is that they are less susceptible to signal quality variations (Maner et al. 2003). However, for analysis and feature extraction purposes, translation, into other domains, is also often required.

After analysing the literature (Bonomi et al. 2009; Srivastava & Wong 2012; Mokaya et al. 2013; Long et al. 2009; Abdullah et al. 2012; Song et al. 2012; Bao & Intille 2004; Krishnan & Panchanathan 2008; Mannini & Sabatini 2010; Ravi et al. 2005; Reiss & Stricker 2012), from the *time* domain, the *mean*, *median*, *standard deviation*, *root mean square (RMS)*, *variance* and *correlation* have been calculated. From the *frequency* domain, *energy*, *entropy*, *peak frequency* and *median frequency* have also been determined. *Geographic* features have also been calculated, including *geographic mean*, *standard deviation* and *standard distance* as well as *mean* for the heart rate data. These features have been chosen because they represent a range of information about the signal.

In order to calculate the *frequency* domain features, a transform from the *time* domain is necessary. In several of the studies reviewed, in order to obtain frequency parameters, Fast Fourier Transform (*FFT*) and Power Spectral Density (*PSD*) are used. Therefore, these steps have been followed. Prior to calculating the *FFT*, the Direct Current (*DC*) component has been removed, as is the case in several studies (Long et al. 2009; Abdullah et al. 2012; Bao & Intille 2004). The *DC* component is the mean acceleration of the signal (Abdullah et al. 2012) and is removed so that the signal is not distorted, as this value is often much larger than the remaining coefficients (Figo et al. 2010). This is illustrated in Figure 77.

```
ankle = data(:,[4]);
ankle_no_dc = ankle - mean(ankle); %dc component has now been removed from the data
sampling_freq_ankle = 100;

t1_ankle = 0:1/sampling_freq_ankle:10;  % Time vector
n1_ankle = ceil(log2(length(ankle_no_dc)));
fft1_ankle = fft(ankle_no_dc, 2^n1_ankle); %FFT
psd_ankle = fft1_ankle.*conj(fft1_ankle); %Power Spectral Density
f1_ankle = sampling_freq_ankle*(0:(2^(n1_ankle-1)-1))/2^n1_ankle; %Frequency
```

Figure 77 Code to Calculate the *FFT* and *PSD*

Following the approach in (Song et al. 2012), *Energy* has been calculated using equation 2, whilst equation 3 calculates *Entropy*. Energy is calculated by calculating the sum of the squared discrete *PSD* components, *x*, of the signal. The sum was then divided by the window length for normalization.

$$Energy = \frac{\sum[x]^2}{\mathrm{length(x)}} \tag{2}$$

$$Entropy = \frac{-\sum[x]\log[x]}{\mathrm{length(x)}} \tag{3}$$

The above features, per accelerometer, have been calculated using this method. Following the approach in (Kim & Kotz 2011), *Correlation* has been calculated using Pearson's correlation (see equation 4). Given two variables *x* and *y*, Pearson's correlation then calculates the correlation between each pair of accelerometer vectors, in this instance between the ankle–chest, chest–hand and hand–ankle.

$$Correlation = \frac{\sum xy - \frac{\sum x \sum y}{n}}{\sqrt{\left(\sum x^2 - \frac{(\sum x)^2}{n}\right)\left(\sum y^2 - \frac{(\sum y)^2}{n}\right)}} \tag{4}$$

The raw photographic data has been omitted and is used at a later time, within the memory box. These features comprise the complete feature set. Table 2, below, illustrates the number of generated features in the *time* domain, whilst Table 3 illustrates the *frequency* and *geographic* features. For readability it has been necessary to split the features into two separate tables.

		Time Domain						
Activity	**Total Number of Days**	**Mean**	**Median**	**Standard Deviation**	**Root Mean Square**	**Variance**	**Correlation**	**Heart Rate Mean**
		Ankle, Chest & Hand	Ankle, Chest & Hand	Ankle, Chest & Hand	Ankle, Chest & Hand	Ankle, Chest & Hand	Ankle-Chest, Chest-Hand & Hand-Ankle	
Lying	7	21 Records	21 Records	21 Records	21 Records	21 Records	21 Records	7 Records
Sitting	7	21 Records	21 Records	21 Records	21 Records	21 Records	21 Records	7 Records
Standing	7	21 Records	21 Records	21 Records	21 Records	21 Records	21 Records	7 Records
Walking	7	21 Records	21 Records	21 Records	21 Records	21 Records	21 Records	7 Records
Running	7	21 Records	21 Records	21 Records	21 Records	21 Records	21 Records	7 Records
Ascending Stairs	7	21 Records	21 Records	21 Records	21 Records	21 Records	21 Records	7 Records
Descending Stairs	7	21 Records	21 Records	21 Records	21 Records	21 Records	21 Records	7 Records
Vacuum Cleaning	7	21 Records	21 Records	21 Records	21 Records	21 Records	21 Records	7 Records
Ironing	7	21 Records	21 Records	21 Records	21 Records	21 Records	21 Records	7 Records
Total	**63 Memory Blocks**	**189 Records**	**189 Records**	**189 Records**	**189 Records**	**189 Records**	**189 Records**	**63 Records**

Table 2 Time Domain Feature Set

		Frequency Domain				Geographic		
Activity	**Total Number of Days**	**Energy**	**Entropy**	**Peak Frequency**	**Medium Frequency**	**Geo Mean**	**Geo Standard Deviation**	**Geo Standard Distance**
		Ankle, Chest & Hand	Ankle, Chest & Hand	Ankle, Chest & Hand	Ankle, Chest & Hand	Lat & Long	Lat & Long	Degrees
Lying	7	21 Records	21 Records	21 Records	21 Records	14 Records	14 Records	7 Records
Sitting	7	21 Records	21 Records	21 Records	21 Records	14 Records	14 Records	7 Records
Standing	7	21 Records	21 Records	21 Records	21 Records	14 Records	14 Records	7 Records
Walking	7	21 Records	21 Records	21 Records	21 Records	14 Records	14 Records	7 Records
Running	7	21 Records	21 Records	21 Records	21 Records	14 Records	14 Records	7 Records
Ascending Stairs	7	21 Records	21 Records	21 Records	21 Records	14 Records	14 Records	7 Records
Descending Stairs	7	21 Records	21 Records	21 Records	21 Records	14 Records	14 Records	7 Records
Vacuum Cleaning	7	21 Records	21 Records	21 Records	21 Records	14 Records	14 Records	7 Records
Ironing	7	21 Records	21 Records	21 Records	21 Records	14 Records	14 Records	7 Records
Total	**63 Memory Blocks**	**189 Records**	**189 Records**	**189 Records**	**189 Records**	**126 Records**	**126 Records**	**63 Records**

Table 3 Frequency and Geographic Domain Feature Set

In summary, this data set contains 63 memory blocks (i.e. the number of rows, per activity) and has been calculated by multiply the number of activities by the number of days it was undertaken for (i.e. 9 x 7). Using equation 5, the total number of features (t), per domain, is calculated. This is achieved by summing the *Total* row of coefficients (excluding the *Memory Blocks* value) in Table 2 and in Table 3. *TR* represents the total number of records, per feature.

$$t = \mathrm{TR1} + \mathrm{TR2} + \mathrm{TR}n \quad (5)$$

Using this equation, in the *Time* Domain there are 1,197 features, the *Frequency* Domain contains 756 features and the *Geographic* features include 315, thus totalling 2,268 features.

6.2 Feature Selection

It is now essential to analyse the generated features, since not all of them might be useful. In this instance, Principal Component Analysis (*PCA*) has been used to identify and extract the features that have the best discriminative capabilities and thus contain the most information. During *PCA* three components are calculated – *Eigenvalues*, *Eigenvectors* and *Scores*. *Eigenvalues* measure the amount of variation explained by each Principle Component, with the first coefficient being the largest, *Eigenvectors* are a linear combination of the original variables and have a corresponding *Eigenvalue*, and *Scores* are used in the bi-plot to represent the data by illustrating how close the features are to the first and second Principle Components.

In previous tests, undertaking *PCA* on the entire dataset does not produce optimal results, as the resulting bi-plots are difficult to read. This is due to the number of features contained within the dataset. For this reason, *PCA* has been undertaken in stages. For every three features in that dataset, *PCA* has been performed (see Figure 78). Analysing this reduced number of features enables a clearer view of the data to be seen. As there are two Principle Components, each feature that is close to each corresponding Principle Component has been selected, thus resulting in two features being chosen. As illustrated in Figure 78, the feature closest to the horizontal axis shows that *ankle_energy* is the Principal Component, which has the most discriminant capabilities of the considered features. The feature closest to the vertical axis illustrates *hand_energy* as the second Principle Component with very good discriminant capabilities. Therefore, when choosing the best two features, this approach has been followed, as the features selected are deemed to contain the most information. This process was repeated until all of the features have been analysed. For all of the *PCA* graphs please refer to Appendix 3.

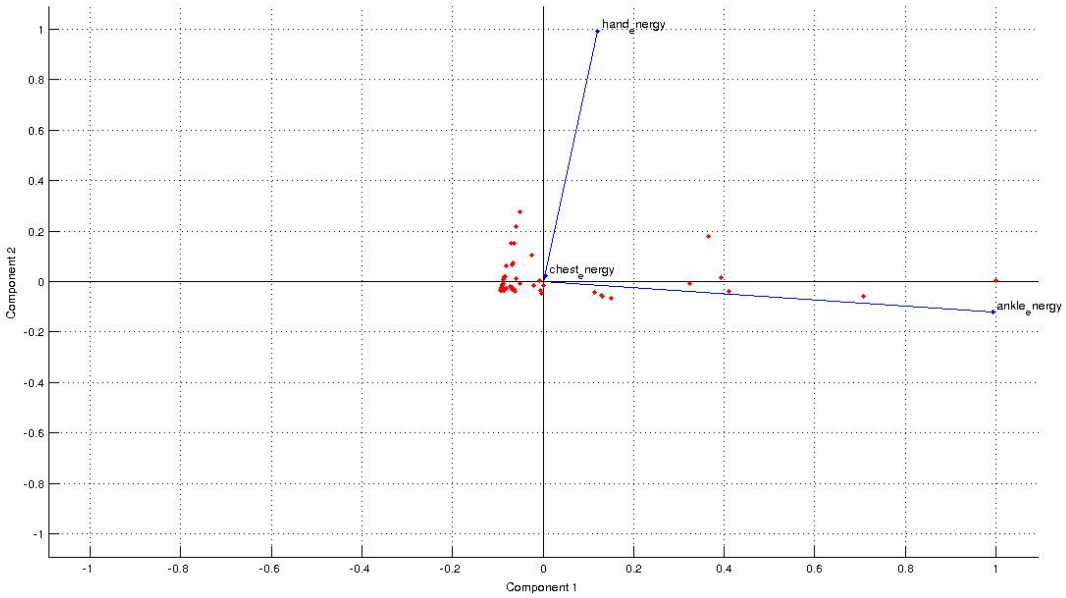


Figure 78 Principle Component Analysis Bi-Plot

Using this method, selecting the optimal number of features is usually problematic (Bins & Draper 2001). However, a scree plot can be used to overcome this. A scree plot plots the *Eigenvalues* in descending order and the appropriate number of features can be determined when the plot levels off. Figure 79, below, illustrates the scree plot that has been generated. Using *PCA*, and the resulting scree plot, the dataset has initially been reduced to eight features.

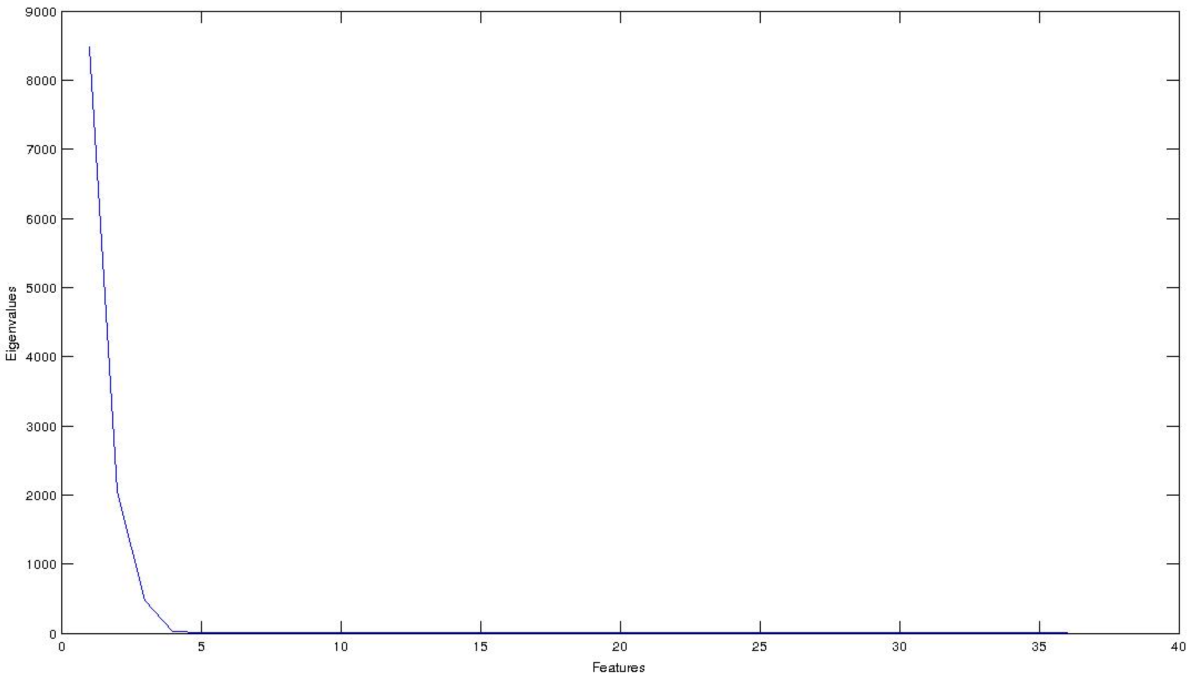


Figure 79 Scree Plot

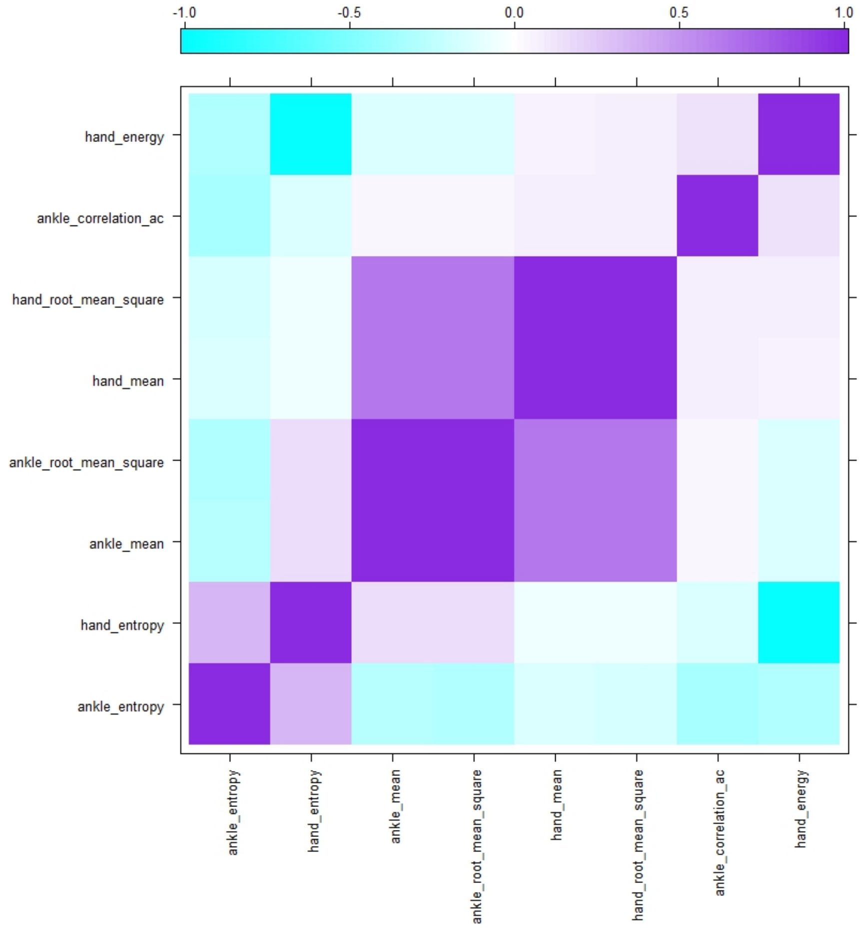
Analysing the features in stages provides a clearer view of the features with the best discriminative capabilities, as the graphs are not distorted with overlapping labels, which can be difficult to read. This process enables the dataset to be drastically reduced so that further analysis can occur. Using the selected features from the *PCA* process, a new dataset is formed called the *PCAFeature* dataset. Table 4, below, illustrates the number of generated features in the *PCAFeature* dataset.

In summary, reducing the number of features has resulted in 504 features remaining. This has been calculated by adding the *Total* number of records, per feature, together. The *PCAFeature* dataset now contains the features that have the most information. However, at this stage in the process, it is still unknown how these features relate to each other. A correlation matrix can be used to illustrate which features are highly associated with each other and thus have been used to reduce the feature set further.

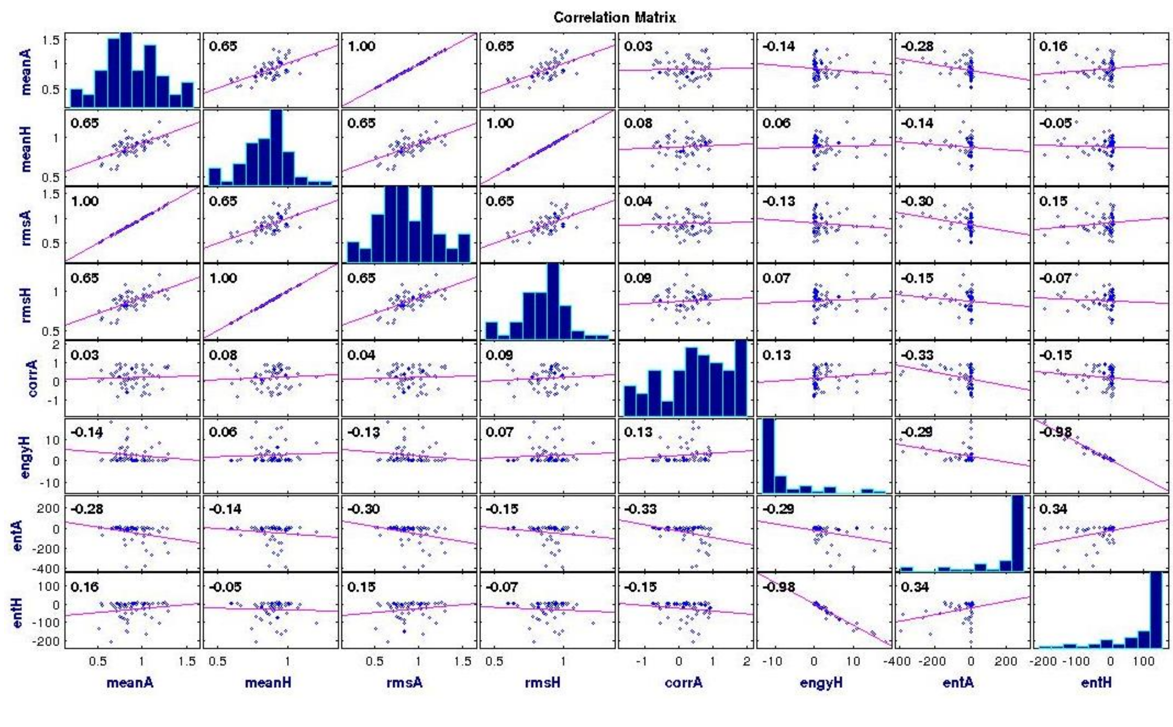
Activity	Total Number of Days	Mean	Root Mean Square	Correlation	Energy	Entropy
		Ankle & Hand	Ankle & Hand	Ankle-Chest	Hand	Ankle & Hand
Lying	7	14 records	14 records	7 records	7 records	14 records
Sitting	7	14 records	14 records	7 records	7 records	14 records
Standing	7	14 records	14 records	7 records	7 records	14 records
Walking	7	14 records	14 records	7 records	7 records	14 records
Running	7	14 records	14 records	7 records	7 records	14 records
Ascending Stairs	7	14 records	14 records	7 records	7 records	14 records
Descending Stairs	7	14 records	14 records	7 records	7 records	14 records
Vacuum Cleaning	7	14 records	14 records	7 records	7 records	14 records
Ironing	7	14 records	14 records	7 records	7 records	14 records
Total	**63 Memory Blocks**	**126 records**	**126 records**	**63 records**	**63 records**	**126 records**

Table 4 *PCAFeature* Set

Figure 80 a) below, illustrates a heat map that depicts the relationship between the features. The pair of features that are darker are more correlated, as opposed to the lighter shades. Figure 80 b) validates Figure 80 a) and illustrates the correlation coefficients for each pair of features. As it can be seen, from both graphs, features that are correlated with themselves are always represented diagonally. As denoted by the coefficients in Figure 80 b), separate features also share 100% correlation as they are measuring the same axis. As it can be seen, from both graphs, *ankle_mean* and *ankle_rms* have 100% correlation, as do *hand_mean* and *hand_rms*. Of all the features, *hand_entropy* has the worst correlation with *hand_energy* at -98%. Therefore, when clustering and classifying the data, only the features with 65% correlation are selected because these features have the highest correlation coefficients in Figure 80 b), without being associated with each other. For the purpose of the evaluation, in chapter seven, this dataset is known as the *CorrFeature* set.



a) Heat Map Correlation Matrix



b) Correlation Matrix

Figure 80 Correlation Matrixes of the *PCAFeature* Feature Set

Through the process of *PCA* and by establishing correlation, the *CorrFeature* set now includes only the features that contain the most information, which are highly associated with each other. This method has removed thirty-two redundant features and has reduced a large dataset to a more manageable set of useful information. Table 5, below, illustrates the number of generated features in the *CorrFeature* set.

Activity	**Total Number of Days**	**Ankle Mean**	**Hand Mean**	**Ankle Root Mean Square**	**Hand Root Mean Square**
Lying	7	7 records	7 records	7 records	7 records
Sitting	7	7 records	7 records	7 records	7 records
Standing	7	7 records	7 records	7 records	7 records
Walking	7	7 records	7 records	7 records	7 records
Running	7	7 records	7 records	7 records	7 records
Ascending Stairs	7	7 records	7 records	7 records	7 records
Descending Stairs	7	7 records	7 records	7 records	7 records
Vacuum Cleaning	7	7 records	7 records	7 records	7 records
Ironing	7	7 records	7 records	7 records	7 records
Total	**63 Memory Blocks**	**63 records**	**63 records**	**63 records**	**63 records**

Table 5 *CorrFeature* dataset

In summary, the number of memory blocks remains the same (63), as the number of rows has not been removed. However, further reducing the number of features has resulted in 252 features remaining.

6.3 Classification

Classification has been used to answer life questions about the user's data. Following an analysis of the literature, simple, yet powerful algorithms, which give good results, have been selected. The classifiers considered include the linear discriminant classifier (*LDC*), quadratic discriminant classifier (*QDC*), uncorrelated normal density based classifier (*UDC*), polynomial classifier (*POLYC*), logistic classifier (*LOGLC*), k-nearest neighbour (*KNNC*), decision tree (*TREEC*), parzen classifier (*PARZENC*), support vector classifier (*SVC*) and Naive Bayes classifier (*NAIVEBC*) (van der Heijden et al. 2004). *LDC*, *QDC* and *UDC* are all normal density based classifiers (Duin et al. 2007). *LDC* is a simple method that finds the linear combination of features which best separate two or more classes of objects and works best when the measurements made on each observation are continuous quantities (Kotsiantis 2007). In contrast to *LDC*, *QDC* estimates mean and covariance for each class, allowing the classifier to find more suitable discriminant functions (Leitner et al. 2003). *UDC* uses a quadratic Bayes classifier assuming normal densities and uncorrelated features (Sirlantzis et al. 2002).

POLYC and *LOGLC* are both linear-based classifiers (Duin et al. 2007). *POLYC* takes the binomial terms of reduced subspace features as inputs and has shown superior performance to multilayer neural networks, in pattern classification (Liu & Sako 2006). Meanwhile, *LOGLC* is a maximum likelihood based classifier using the logistic (sigmoid) function (Sirlantzis et al. 2002).

KNNC, *TREEC*, *PARZENC*, *SVC* and *NAIVEBC* are nonlinear classifiers (Duin et al. 2007). *KNNC* and *PARZENC* are similar in the sense that the classifiers they build still include all training objects and that their parameter (the number of neighbours or the smoothing parameter) can be user supplied or can be optimized over the training set using a leave-one-out error estimation (Duin et al. 2007). *KNNC* is based on the principle that the instances within a dataset will generally exist in close proximity to other instances that have similar properties. It locates the *k* nearest instances to the query instance and determines its class by identifying the single most frequent class label (Kotsiantis 2007). Meanwhile, *TREEC* classifies instances by sorting them based on feature values. One of the most useful characteristics of decision trees is their comprehensibility; they tend to perform better when dealing with discrete or categorical features (Kotsiantis 2007). *PARZENC* uses a Gaussian kernel with optimised width to estimate class densities and calculate posterior class probabilities (Sirlantzis et al. 2002). The object is to obtain estimates of the conditional probability densities. Each sample in the training set contributes in a like manner to the estimate. The estimation process is space-invariant (van der Heijden et al. 2004). *SVC* revolves around the notion of a “margin” – either side of a hyper-plane that separates two data classes. Generally, *SVC*s tend to perform much better when dealing with multi-dimensions and continuous features. On the other hand, logic-based systems tend to perform better when dealing with discrete or categorical features (Kotsiantis 2007). *NAIVEBC* are composed of directed acyclic graphs with only one parent (representing the unobserved node) and several children (corresponding to observed nodes) with a strong assumption of independence among child nodes in the context of their parent. Moreover, *NAIVEBC* and the *KNNC* can be easily used as incremental learners whereas rule algorithms cannot. *NAIVEBC* is naturally robust to missing values since these are simply ignored in computing probabilities and hence have no impact on the final decision. On the contrary, *KNNC* and neural networks require complete records to do their work (Kotsiantis 2007).

6.3.1 Validation Methods

The *holdout* method is used to partition the dataset into two independent sets, a training set and a test set (Adhvaryn & Panchal 2012). The test set is used to estimate only one parameter, i.e. the error rate, and the training set is used to train all other parameters of the classifier, therefore the training set must be larger than the test set (van der Heijden et al. 2004). Using a common approach, and to avoid overfitting, the *CorrFeature* dataset has been separated such that 80% of the whole dataset is designated for training and the remaining 20% for testing (van der Heijden et al. 2004; Adamopoulos & Tuzhilin 2013). In order to maintain generalisation, the learning and testing stages are repeated. The average performance obtained from 100 cycle simulations is utilised. This number is considered, by statisticians, to be an adequate number of iterations to obtain an average (Salkind 2008). After each repetition, the error rate for each classifier is stored and the learning experience of the algorithm cleared so that it does not influence the next test. Carrying out several repetitions provides mean *error rates*, *standard deviations* and performance values for each classifier.

In order to estimate the accuracy of the classifiers, the *k-fold* cross-validation technique is also used. In this instance, the dataset is randomly partitioned into k mutually exclusive subsets. Training and testing is performed k times (Adhvaryn & Panchal 2012). During this evaluation, k has been set to five, using 1 and 100 repetitions, respectively. Setting $k = 5$ is a typical choice as there is a lower variance in the data (Hastie et al. 2009). These results are then compared with those from the 80/20 holdout approach. *Sensitivity* (true positives) and *Specificity* (true negatives) measure the predictive capabilities of the classifiers in the binary classification tests. In this instance, *Sensitivity* refers to the system's ability to recall a memory and that there is a high probability that the memory occurred. On the other hand, *Specificity* refers to the probability that the memory did not occur. For example, when asking the question "Have I been running?" a higher *Sensitivity* rate indicates that the system is able to correctly classify running and establish that this activity did occur. However, if the user has never run before then a higher *Specificity* rate would correctly indicate that the user has never run before.

A Receiver Operating Characteristics (*ROC*) graph is also used to summarise the classifier's performance and for visualization and organization (Fawcett 2006). It works by calculating the trade-offs between true positive and true negative error rates. In order to directly compare the classifiers, their *ROC* performances have been reduced to a single scalar value (representing expected performance) by calculating the Area Under the Curve (*AUC*) (Fawcett 2006). This calculation is another accepted performance metric that provides a value equal to the probability that a classifier will rank a randomly chosen positive instance higher than a randomly chosen negative one (this obviously assumes that positive ranges higher than negative) (Fawcett 2006).

The *PRTools*[15] pattern recognition toolbox has been used to implement the classification algorithms and the validation methods, in chapter seven. Each classifier has been evaluated to determine its overall performance, and accuracy, in finding information in human digital memory datasets.

6.3.2 Data Over-Sampling

In the *CorrFeature* set, there are 63 rows of memory blocks, with each activity only containing 28 features (seven days x four features), thus, the number of instances that the classifiers can learn from, per activity, is limited. Hence, the minority of true positive results (*Sensitivity*) is lower. Learning from data sets that contain very few instances of the minority class usually produces biased classifiers that have a higher predictive accuracy over the majority class, but poorer predictive accuracy over the minority class (Chawla et al. 2003). For example, in the case of answering the question, "Have I been running?" given a random sample taken from the dataset, the probability of a classifier classifying a memory as "running" is quite low. This is because the algorithms have a limited number of instances to learn about the features of this activity as training 80% of the data results in only six memory blocks of data, per activity, being trained. The system then has merely one chance to correctly identify running from the test data. The probability of accurately identifying running out of the majority of incorrect activities is quite low. However, in the case of oversampling the data, the system would have more information to learn off and to test against, thus increasing the probability of correctly identifying the activity.

In order to address this problem, it is necessary to resample the dataset. Various resampling techniques are available, including under-sampling and over-sampling (Chawla et al. 2002). Under-sampling reduces the number of records from the majority class to make it equal to the minor class. Meanwhile, whilst over-sampling augments the minority class by exactly duplicating the examples of the minority class and thus enlarges the dataset (Han et al. 2005). In this instance, the synthetic minority over-sampling technique (SMOTE) is used rather than reducing the dataset further (Chawla et al. 2002). Using SMOTE, the entire dataset is oversampled using each activity label, in order to generate new synthetic records. This approach is an accepted technique for solving the problems related to unbalanced datasets (Chawla et al. 2002).

Using SMOTE, the dataset has been doubled in size to 126 memory blocks. It has then been doubled for a second time to 252 memory blocks and again to 504. It has been necessary to increase the size of the dataset to see if the results can be improved. These oversampled datasets are now known as *CorrFeature_Oversampled1*

[15] http://prtools.org/

(126 memory blocks), *CorrFeature_Oversampled2* (252 memory blocks) and *CorrFeature_Oversampled3* (504 memory blocks).

6.4 Clustering

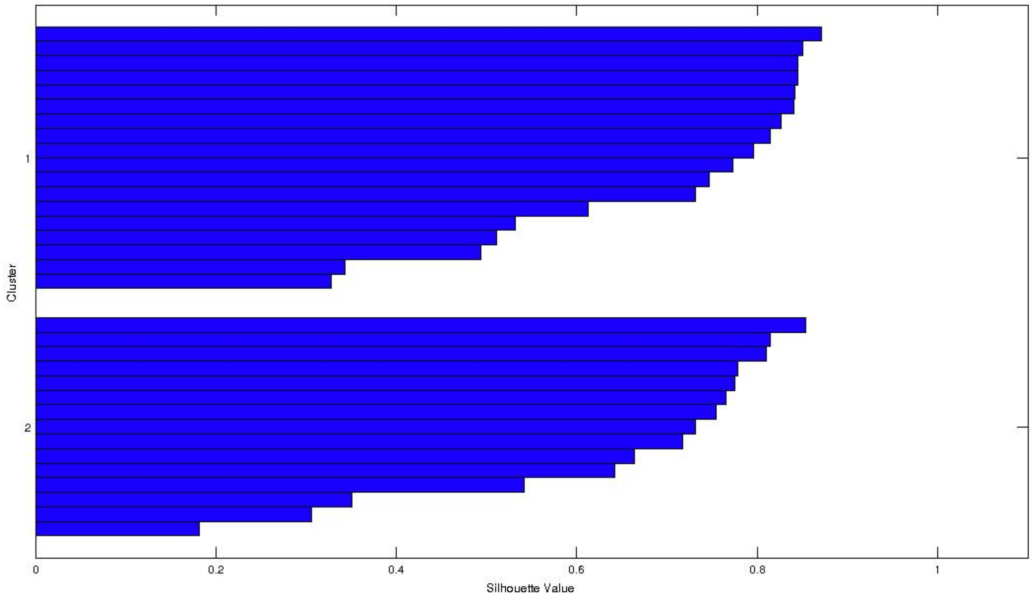
K-means clustering has been chosen to create memory boxes. This method is being explored as an alternative to traditional searching methods. Instead of defining specific queries, data is grouped together based on its similarities. As with defining search queries, a pre-existing knowledge of the data is required. However, using this method, data can be explored and natural groupings occur. *K-means* was chosen because of its simplicity, and because it is the most widely used clustering algorithm in practice, which has been used in a variety of application domains (Wagstaff et al. 2001; Wu et al. 2007; Forero et al. 2004). As previously stated, defining the number of clusters (K) can be challenging. In this instance, a silhouette plot has been used to overcome this challenge.

In order to first determine the appropriate number of clusters to use, the silhouette values are calculated and plotted. This has been undertaken in Matlab, using the *silhouette* function. The silhouette value $S(i)$ quantifies the similarity of an object i to the others in its own cluster, compared to the objects in other clusters (Forero et al. 2004; Rousseeuw 1987). These values range from +1, indicating points that are very distant from neighbouring clusters, through to 0, indicating points that are not distinctly in one cluster or another, to -1, indicating points that are probably assigned to the wrong cluster (MathWorks 2013). The silhouette average (SA) is then calculated and is used as a measurement of the quality of the resulting clusters (Forero et al. 2004). The value of k that has the largest SA indicates the most appropriate value to use.

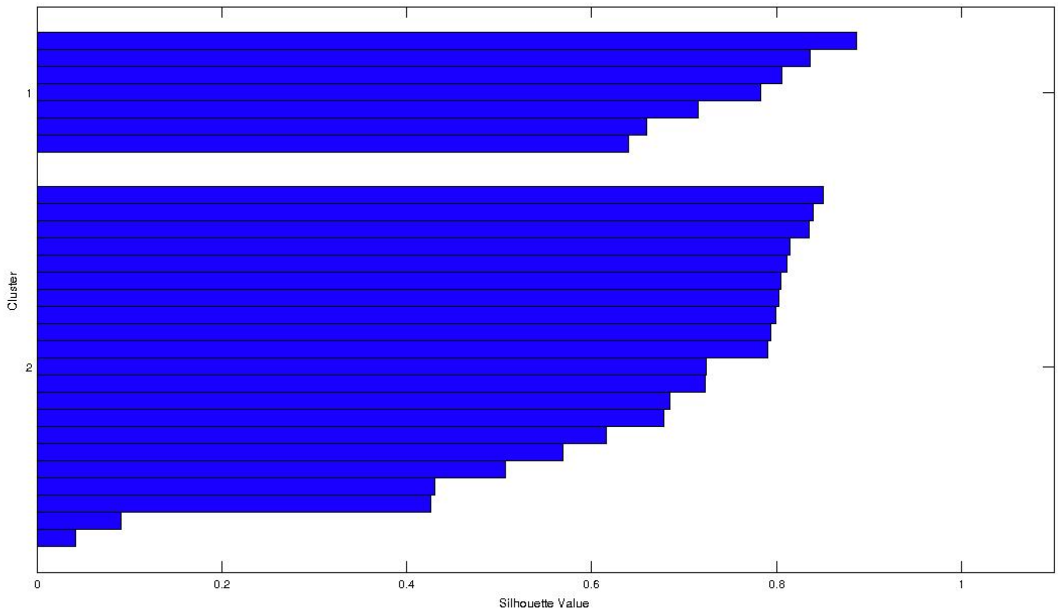
Using the *CorrFeature* set and these steps, the value of k has been increased, from 2 to 5, and evaluated using the *silhouette* function. This has been executed for the questions, “When have I been the most active?” and “When have I been the least active?” In these examples, all ‘high-energy’ and ‘low-energy’ activities have been filtered and their SA’s calculated. In this instance, ‘high-energy’ activity refers to walking, running, ascending and descending stairs and vacuum cleaning, whilst ‘low-energy’ activity denotes lying down, sitting, standing and ironing. Table 6, below, illustrates the SA’s of the various cluster sizes for high and low energy activities. As can be seen from this table, for this dataset, 2 seems to be the most appropriate number of clusters to use. Figure 81, below, illustrates the resulting silhouette plot where k = 2, for both high and low energy activities.

Cluster (*k*)	Silhouette Average (*SA*)	Silhouette Average (*SA*)
	High Energy	Low Energy
2	0.675606236	0.677199149
3	0.60073687	0.492206707
4	0.596855235	0.51111136
5	0.586823923	0.494339558

Table 6 Silhouette Averages for *k* Clusters



a) High-energy



b) Low-energy

Figure 81 Silhouette Plot of the *CorrFeature* Dataset for high and low-energy activities

As it can be seen in Figure 81, the clusters are strong, as the silhouette values are greater than 0.6. This value is then used in chapter seven to implement the clustering algorithms.

6.5 Summary

This chapter describes the evaluation methodology that has been used to pre-process raw human digital memory data. This is essential so that noise is removed from the raw signals and so that the size of the dataset can be reduced. The chapter has then described how features have been extracted and analysed, using *PCA* and correlation matrixes. The redundant features have been removed and the resulting *CorrFeature* dataset is composed of only the features that contain the most information and that relate to each other. This feature set has then been oversampled, using SMOTE. Oversampling the entire dataset set enables more records to be available for clustering and classification. The chapter then concludes with a discussion about the validation methods, as well as the classification and clustering algorithms, which will be used to evaluate DigMem's searching methods

Using the discussed methodology, and the resulting *CorrFeature* dataset, the following chapter presents an evaluation of DigMem's searching methods. In this way, the results from both the supervised and unsupervised machine learning algorithms are presented. The chapter also provides a comparison between the state-of-the-art lifelogging devices and DigMem to illustrate the advantages that the system has over current systems.

Chapter 7

Evaluation

In the previous chapter, the evaluation methodology has been discussed, which detailed the approach that has been used to pre-process raw human digital memory data so that features can be extracted. These features have then been analysed so that redundant information is removed. This process resulted in the creation of the *CorrFeature* dataset. Using the SMOTE over-sampling technique, this dataset has then been oversampled numerous times, so that the machine learning algorithms have more data to learn from. Using the *CorrFeature* and oversampled datasets, this chapter evaluates supervised and unsupervised machine learning algorithms, as a viable search tool for searching human digital memory data. Supervised algorithms have been utilised to evaluate the system's ability to answer the question, "Have I been running", whilst unsupervised algorithms have been used to answer the questions, "When have I been the most active?" and "When have I been the least active?" As well as these results, a comparison between the state-of-the-art lifelogging devices and DigMem has also been made to illustrate how DigMem compares with other solutions.

7.1 Comparison Against Existing Approaches

DigMem offers a flexible solution that embraces the use of pervasive mobile devices, cloud computing, Peer-to-Peer networking, and machine learning. The system provides a method of creating human digital memories in any environment, using any DigMem compliant devices. This has many advantages over other systems. Systems, such as (Doherty, Moulin, et al. 2011; Lee et al. 2008; Doherty, Caprani, et al. 2011; Sellen et al. 2007), use Microsoft's *SenseCam* to create memory browsers within lifelogging research, whilst (Almeida et al. 2011; Dwyer et al. 2009; Taraldsen et al. 2011; Lord et al. 2011) use the *activPAL* and *SWA* devices to monitor behaviour. The disadvantage of these systems is that specialist, and expensive equipment are often needed and the data is quite limited. However, the DigMem system overcomes these shortcomings by using any DigMem compliant device, for data collection. The system is not limited to collecting photos; any number of services can be included in a memory. Table 7 provides a comparison between the state-of-the-art lifelogging devices and DigMem, whilst Table 8 provides a comparison of DigMem's system functionality against these devices.

	Data Collected	Automatic Upload of Data	Query Technique	Expensive	Extendable	Adaptive (Flexible)	Storage	Open Source	Ability to Answer Life Questions
DigMem	Unlimited from ad-hoc services*	Yes	Probabilistic	No	Yes	Yes	Cloud (unlimited)	Yes	Yes
SenseCam	Photos & Sensor Readings (light, temperature & accelerometer)	No	Keyword	Yes	No	No	1GB Flash Memory - 30,000 images (8 days of data approx.)	No	No
SenseWear Armband (SWA)	Motion, Step Count, Sweat Rates, Skin Temperature, Body Heat	No	Time-Based	Yes	No	No	28 days worth of data (data rate is 32 times/second)	No	No
ActivPAL	Activity Data (Sitting, Standing, Walking), Step Count, Transitions, Energy Expenditure	No	Time-Based	Yes	No	No	4MB	No	No

*Any device that supports DigMem can be used for data collection, thus not limiting the data that is collected

Table 7 Comparison of DigMem against the state-of-the-art lifelogging devices

	Create & connect to P2P Network	Ability to publish services	Allow external devices to connect to gather information
DigMem	Yes	Yes	Yes
SenseCam	Closed System	Closed System	Closed System
SenseWear Armband (SWA)	Closed System	Closed System	Closed System
ActivPAL	Closed System	Closed System	Closed System

Table 8 Comparison of DigMem System Functionality

As it can be seen, DigMem offers a number of advantages over current systems. The data collected is not limited, it is automatically uploaded to the cloud and searching is based on features. Memory boxes are created with a probabilistic confidence value that the search features, and the features in the memory block, are similar. It is also able to run on any Android (open source) device, thus expensive equipment is not needed. Its plug-and-play middleware is also extendable. When services are not available, replacement devices are found, thus making it extremely adaptive and flexible. The use of cloud storage also provides an unlimited amount of space, depending on which cloud service is used. It is also able to create, and connect devices to, a P2P network, which enables such devices to publish their services and allow external devices to collect data. Memories are created in an unobtrusive manner and are collected from environment specific devices, instead of being device specific (for example, only using a *SenseCam* to collect photos), which is a novel aspect of the work. Data, collected over a lifetime, is semantically linked and any time of our lives can be re-constructed. This has profound implications in the use of such a system and, currently, it is being used to investigate its effectiveness in monitoring and measuring sedentary behaviour (Dobbins et al. 2013). The ability to answer life questions is also a feature that is unique to the system.

In terms of capturing data, there are several notable features that make this system a viable alternative to the systems described in Table 7 and Table 8. Most notably the data-gathering platform typically on the tablet device, as well as the previously implemented MoDM middleware (Dobbins, Merabti, et al. 2012b), are built using the open-source Android operating system. This allows the hardware features of devices to be freely accessed, ensuring the data can be gathered in a non-proprietary format. This is very important as the information can be manipulated at a later time and the longevity of the data is increased.

In addition, the DigMem system is able to provide adequate flexibility in a number of environments, and allows for any number of ad-hoc services to be used, for gathering information for the human digital memory. The devices that are available shape the memories. Devices present in one environment will differ to those of another, thus altering the information that is available. A P2P network can handle this dynamically changing set of peers. This P2P architecture is used to gain access to the services that devices have to offer. The system is also adaptive to accommodate peers leaving the network. For example, if a camera suddenly becomes unavailable, perhaps if it is placed inside a bag or pocket, or runs out of battery, then another camera service can be chosen. For instance, a Closed-circuit television (CCTV) camera in the city centre would be recording images continually. If the original camera service has become unavailable, then the CCTV camera, if it was DigMem compliant, could be used instead. With recent advances within the field of facial recognition (Dadgostar et al. 2011; Introna & Wood 2004; Lee et al. 2006), this idea could soon be a possibility. In the future, this technology could be used, within DigMem, so that the system knows what its users look like. If CCTV images were being used, then DigMem would be able to gather these images, identify the user, match the captured images to their profile and add the images to their record. The original service, which is now unavailable, is replaced with another that is available. By retrieving data from these pervasive devices, memories are richer in detail. Information is incorporated into the memory that necessarily would not have been available otherwise. For example, a mobile phone would not be able to take the temperature reading of the room; however, by connecting to the thermostat this information could be retrieved and sent back, to enhance the memory.

Transferring the data to the cloud enables it to be distributed across many cloud storage areas, and accessed from anywhere in the world. The storage capacity of mobile devices is also limited. Therefore, transferring data to the cloud enables more material to be collected, and more complex processing can be carried out, at a later time. This is a very simple and effective way of using Cloud Computing technology to send data, via the Internet, to different locations (Dikaiakos et al. 2009). This method of data transfer eliminates the need for the user to manually upload their content, which is the current method used in other systems. Regarding the storage of the raw data, the decision to store this information in a database allows developers the freedom to choose the data processing tools that they require, i.e. Excel, SPSS, or Matlab.

Transferring the raw data into the Raw Dataset ensures that every piece of information is processed. This element grows with the user and becomes their human digital memory data space. Extracting features enables probabilistic searches to be performed. This is important in order to create the memory boxes and for the system to be able to learn about the user. This is beneficial, compared to traditional searching methods, such as SPARQL, where explicit queries are required. SPARQL is a complex language that relies on the user understanding the domain before queries can be constructed. However, if the user is unfamiliar with the underlying syntax then finding information can be very difficult, due to the complexity of the language.

Memories are composed of much more than merely photographs and location. As the amount of data increases so does the difficulty in searching it. This is highlighted by Fuller *et al.* (Fuller et al. 2008), who states, "*HDM data is highly heterogeneous and unstructured, therefore, it is difficult to form search queries*". However, the use of machine learning algorithms can alleviate this problem as they are capable of analysing large sets of data. Instead of focusing on producing queries to search this enormous set of "*heterogeneous and unstructured"* (Fuller et al. 2008) data, DigMem focuses on treating the searches as a machine learning problem. Therefore, using the DigMem system, this mass of varied data can be easily searched, without the user having a pre-existing knowledge of the dataset. This method enables much richer and more-detailed memories to be created, than previously seen and enables life questions to be answered. In response to this challenge, the evaluation of the system relates to assessing how human digital memory data can be searched effectively.

7.2 Supervised Machine Learning Results

This section presents the results for classifying human digital memory data, using several supervised machine learning algorithms (discussed in section 6.3 Classification). The previously created *CorrFeature* dataset is considered, using an 80% *holdout* technique and *k-fold* cross-validation (discussed in section 6.3.1 Validation Methods). The primary focus is to demonstrate how the system is able to recognise activities, for the purpose of answering life questions. In this demonstration, the question “Have I been running?” has been asked to determine if the system is able to recall the memories of when the user has been running. In this way, we describe the features of an activity and use probabilistic reasoning to filter human digital memory data that contain similar features to those described.

7.2.1 Results Using Original *CorrFeature* dataset

The first evaluation uses the original *CorrFeature* dataset to answer the question “Have I been running?” The performance for each classifier is evaluated, using the *sensitivity*, *specificity*, mean error, standard deviation and *AUC* values with 100 simulations and randomly selected training and testing sets for each simulation.

7.2.1.1 Classifier Performance

To evaluate the performance, of each classifier, the *classperf* function, within *PRTools*, is used. Table 9, below, illustrates the mean averages obtained over 100 simulations for the *sensitivity*, *specificity*, and *AUC*.

	Sensitivity	**Specificity**	**AUC**
Classifier	*HDM*	*HDM*	*HDM*
LDC	0.6700	0.9613	75.15%
QDC	0.4000	0.9438	88.09%
UDC	0.7700	0.8350	74.47%
POLYC	0.9800	0.8350	82.07%
LOGLC	0.7300	0.9250	82.34%
KNNC	0.3600	0.8450	85.30%
TREEC	0.2800	0.9225	93.73%
PARZENC	0.6000	0.8425	76.59%
SVC	0.2600	0.8938	81.62%
NAIVEBC	0.5000	0.8138	82.58%

Table 9 Averages of Classifier Performance for the Original *CorrFeature* Dataset

As illustrated in Table 9, *sensitivities* (ability to recall an activity), in this initial test, are quite low for most of the classifiers. However, this was expected because the dataset does not have enough records for the classifiers to learn from and test against. In order to determine the accuracy of the classifiers, the *k-fold* cross-validation technique has also been used. This was performed using the *crossval* function, in *PRTools*, to determine whether the results, obtained from the 80% *holdout* method, could be improved. Table 10, below, illustrates these results.

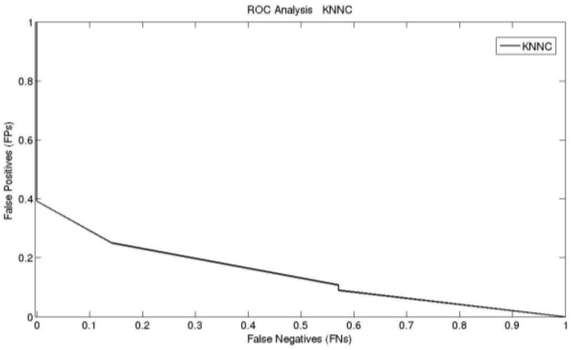
	80% Holdout: 100 Repetitions		**Cross Val, 5 Folds, 1 Repetition**	**Cross Val, 5 Folds, 100 Repetitions**	
Classifier	*Mean Error*	*SD*	*Mean Error*	*Mean Error*	*SD*
LDC	0.6989	0.1343	0.6984	0.6890	0.0336
QDC	0.7589	0.1234	0.7619	0.7805	0.0333
UDC	0.7667	0.1307	0.7143	0.7570	0.0318
POLYC	0.6956	0.1055	0.6667	0.6868	0.0255
LOGLC	0.6889	0.1283	0.6984	0.6903	0.0302
KNNC	0.8233	0.1162	0.8413	0.8090	0.0379
TREEC	0.8200	0.1158	0.9206	0.8367	0.0353
PARZENC	0.7678	0.1025	0.7143	0.7752	0.0357
SVC	0.8089	0.0894	0.7778	0.8148	0.0232
NAIVEBC	0.8278	0.0899	0.8413	0.8392	0.0225

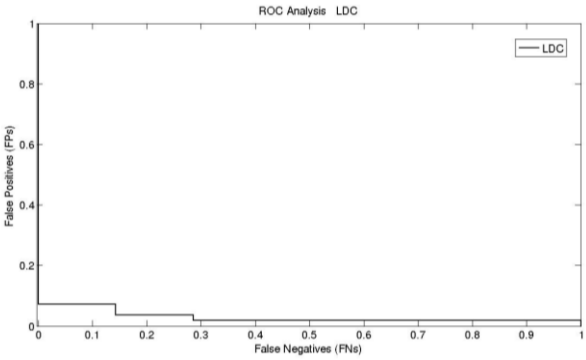
Table 10 Cross Validation Results for the Original *CorrFeature* Dataset

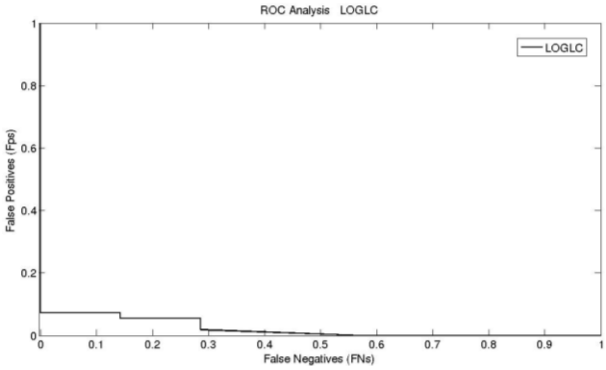
The *k-fold* cross-validation results, using *five* folds and *one* and *one hundred* repetitions illustrates that the error rates have slightly improved, for most of the classifiers. However, the error rates are still relatively high. Furthermore, the lowest error rates could not be improved below the minimum expected error rate of 11.11% (7 days per activity / 63 total records). This could be attributed to the fact that the dataset was rather small.

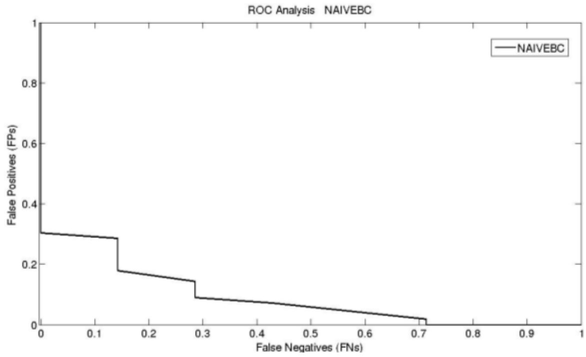
7.2.1.2 Model Selection

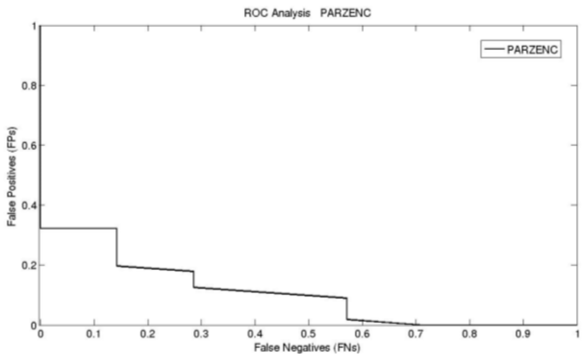
The *ROC* curves (see Figure 82) show the cut-off values for the *false negative* and *false positive* rates, for each of the classifiers used. In terms of accuracy, several of the classifiers performed well, such as the *POLYC* and *LOGLC*. The high *AUC sensitivity* and *specificity* values in Table 9 support these findings; *POLYC* has an accuracy of 82.07%, *sensitivity* of 98% and *specificity* of 83.5%, whilst *LOGLC*’s accuracy is 82.34%, *sensitivity* was 73% and *specificity* 92.5%.

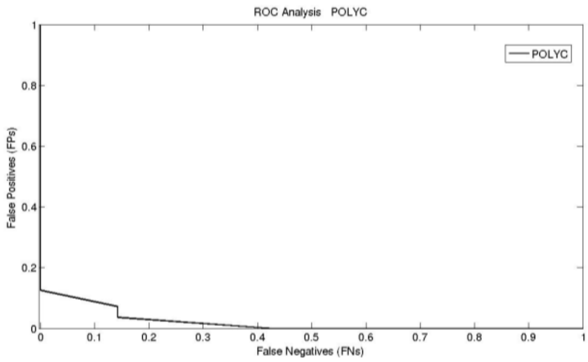

ROC Analysis KNNC
KNNC
False Positives (FPs)
False Negatives (FNs)

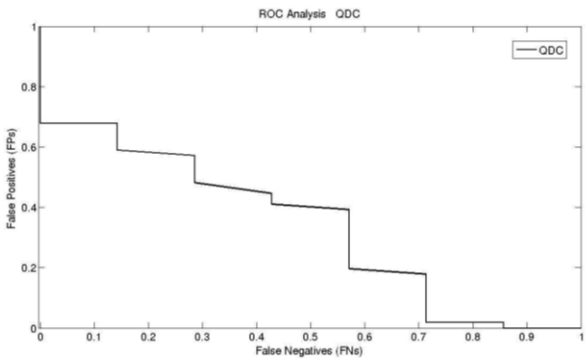

ROC Analysis LDC
LDC
False Positives (FPs)
False Negatives (FNs)

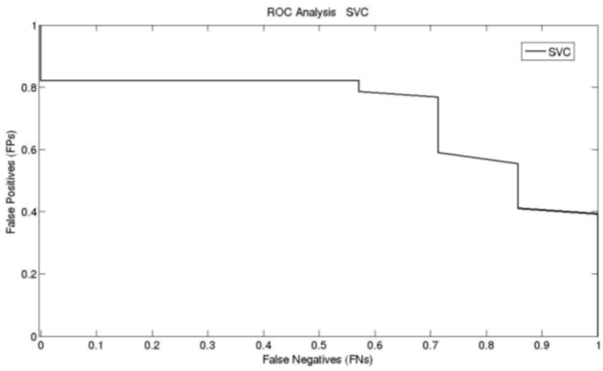

ROC Analysis LOGLC
LOGLC
False Positives (FPs)
False Negatives (FNs)


ROC Analysis NAIVEBC
NAIVEBC
False Positives (FPs)
False Negatives (FNs)


ROC Analysis PARZENC
PARZENC
False Positives (FPs)
False Negatives (FNs)


ROC Analysis POLYC
POLYC
False Positives (FPs)
False Negatives (FNs)


ROC Analysis QDC
QDC
False Positives (FPs)
False Negatives (FNs)


ROC Analysis SVC
SVC
False Positives (FPs)
False Negatives (FNs)

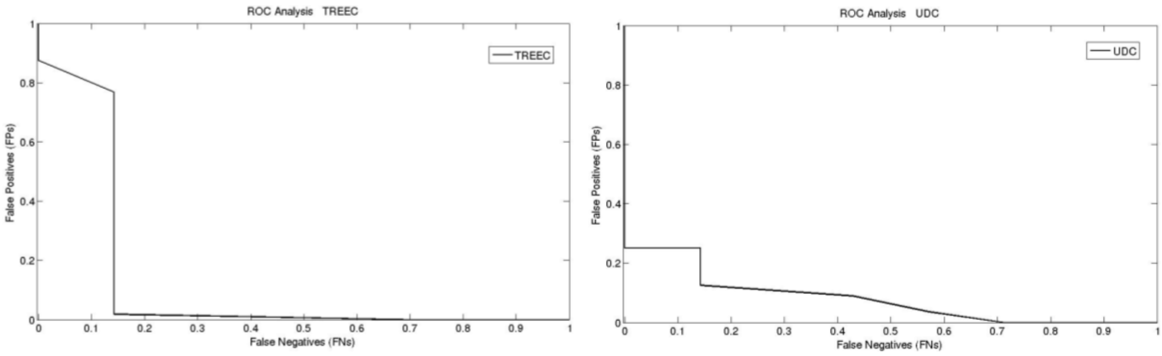


Figure 82 Receiver Operator Curves for the Original *CorrFeature* dataset

Although these results are quite good, they can be improved upon. Consequently, for most of the classifiers, *sensitivities* are quite low while *specificities* are higher. In this evaluation, higher *sensitivities* are more important. As previously stated, *sensitivity* refers to the system's ability to recall a memory and that there is a high probability that the memory occurred. On the other hand, *specificity* refers to the probability that the memory did not occur. Therefore, for the purpose of the evaluation, it is more important to correctly recall a memory.

As previously stated in section 6.3.2 Data Over-Sampling, the *CorrFeature* dataset is relatively small as there are only 63 rows of memory blocks, with each activity only containing 28 features (7 days x 4 features). Therefore, since the original *CorrFeature* dataset is limited it can be presumed that this is the reason that the results are quite low, as the classification algorithms do not have enough records to learn from and test against, in order to accurately answer the question. Re-sampling the dataset is a conventional way to rectify this problem (Tong et al. 2011). In the next section the results from oversampling the *CorrFeature* are presented to try and improve the results.

7.2.2 Results Using the *CorrFeature_Oversampled1* dataset

Using the SMOTE technique (Chawla et al. 2002), the *CorrFeature* dataset has been re-sampled to 126 memory blocks. The algorithm allows a new dataset to be generated that contains double the amount of memory blocks.

7.2.2.1 Classifier Performance

Table 11, below, illustrates the mean averages obtained over 100 simulations for the *sensitivity*, *specificity*, and *AUC*. These results are encouraging. The *sensitivity*, *specificity*, and *AUC* values, for all the algorithms, have increased. In particular, *POLYC*'s *sensitivity* has increased to 99.5%, *specificity* to 92.44% and accuracy to 87.85%. Doubling the size of the dataset increased the algorithms ability to classifying running.

	Sensitivity	**Specificity**	**AUC**
Classifier	*HDM*	*HDM*	*HDM*
LDC	0.9050	0.9775	88.13%
QDC	0.5850	0.9756	95.53%
UDC	0.8050	0.8819	74.79%
POLYC	0.9950	0.9244	87.85%
LOGLC	0.9300	0.9556	89.23%
KNNC	0.7150	0.8950	96.44%
TREEC	0.6900	0.9544	98.58%
PARZENC	0.7750	0.8938	94.27%
SVC	0.6200	0.8488	87.57%
NAIVEBC	0.7900	0.8938	89.59%

Table 11 Averages of Classifier Performance for the *CorrFeature_Oversampled1* dataset

Table 12, below, illustrates the corresponding *k-fold* validation results, which have also been improved. However, whilst the *mean error* rates have been significantly reduced, they are still higher than the minimum expected error rate of 11.11% (14 days per activity / 126 total records). Further improvement is still required.

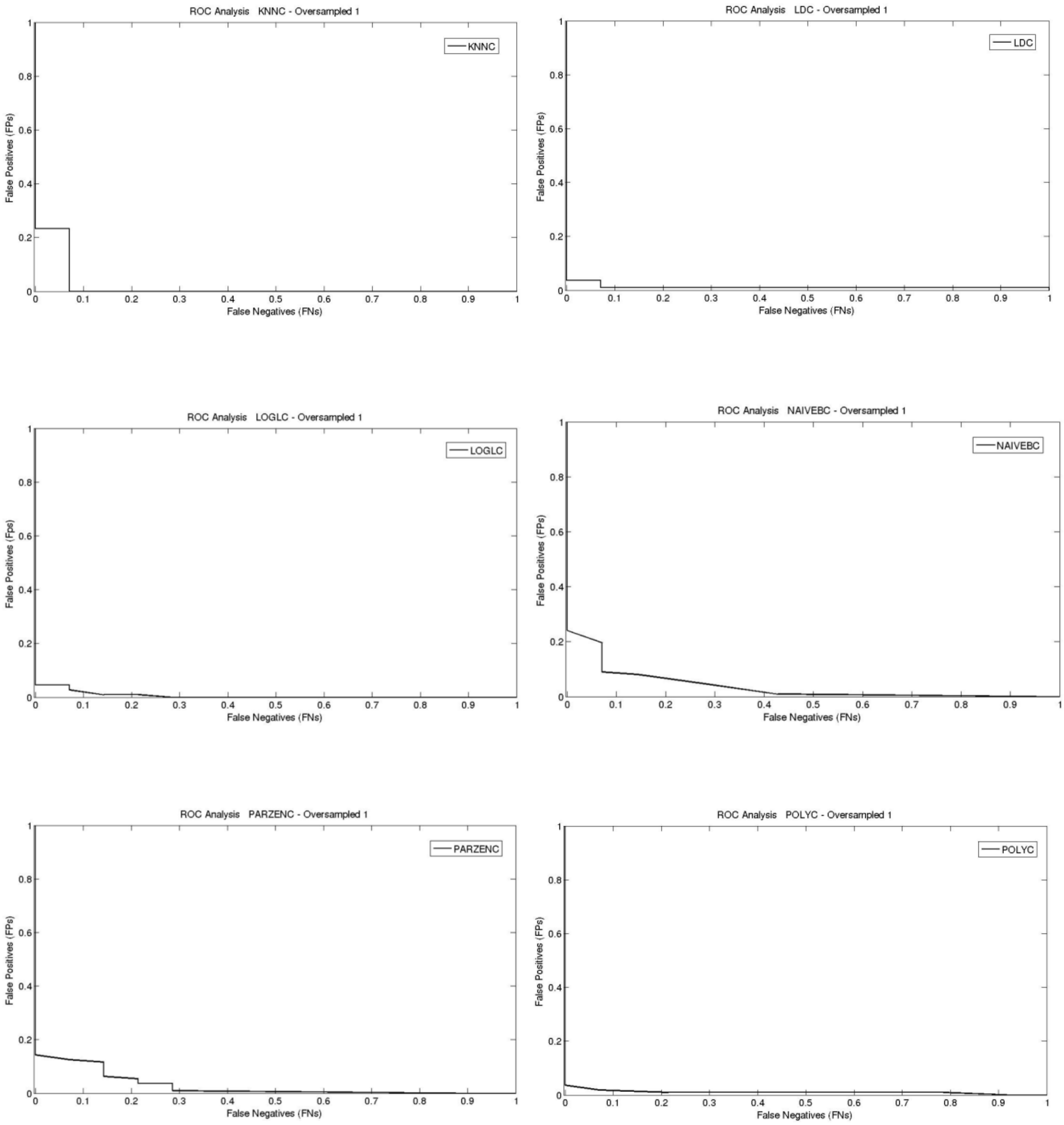
	80% Holdout: 100 Repetitions		**Cross Val, 5 Folds, 1 Repetition**	**Cross Val, 5 Folds, 100 Repetitions**	
Classifier	*Mean Error*	*SD*	*Mean Error*	*Mean Error*	*SD*
LDC	0.3578	0.1069	0.3254	0.3677	0.0219
QDC	0.3272	0.1122	0.3651	0.3942	0.0425
UDC	0.7100	0.0881	0.6825	0.7032	0.0257
POLYC	0.4228	0.1100	0.4286	0.4115	0.0314
LOGLC	0.3028	0.1061	0.3016	0.2963	0.0163
KNNC	0.3694	0.1164	0.3810	0.3562	0.0133
TREEC	0.3528	0.1084	0.3730	0.3545	0.0178
PARZENC	0.3744	0.1025	0.4127	0.3886	0.0275
SVC	0.6961	0.0566	0.7143	0.7120	0.0138
NAIVEBC	0.4422	0.0999	0.4524	0.4445	0.0151

Table 12 Cross Validation Results for the *CorrFeature_Oversampled1* dataset

The results indicate that increasing the size of the dataset has improved the algorithms ability to classify activities. This was expected as there is more data for the algorithms to learn from and test against.

7.2.2.2 Model Selection

Again, the ROC curves show the cut-off values for the false negative and false positive rates. Compared to Figure 82, Figure 83 shows a significant improvement, for most of the classifiers. Once again, the high *AUC*, *sensitivity* and *specificity* values in Table 11 support these findings. In comparison to Table 9, *PARZENC*'s accuracy has improved dramatically by 17.68%, whilst *LDC*'s has improved by 12.98%. *TREEC*'s *sensitivity* has improved by 41%, whilst *SVC*'s has improved by 36%. *Specificity* has improved slightly, with POLYC improving by 8.94% and *NAIVEBC* by 8%



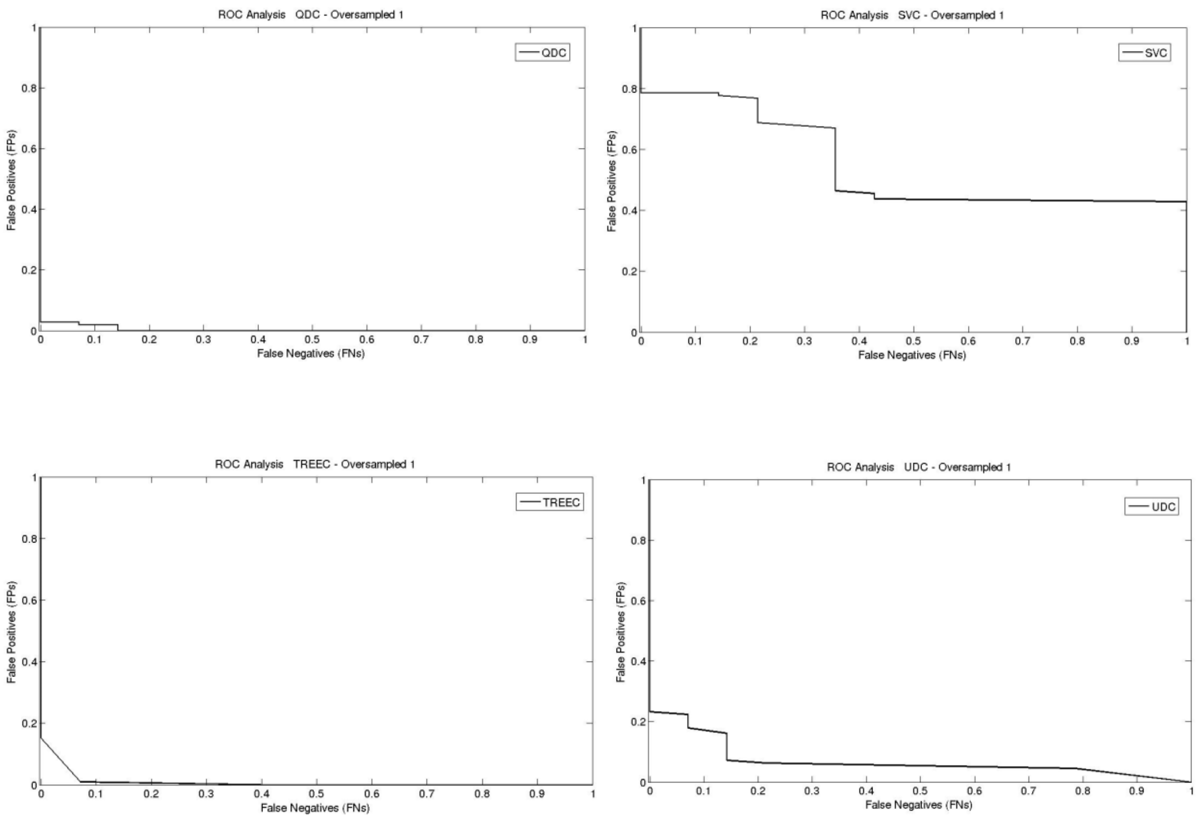


Figure 83 Receiver Operator Curves for the *CorrFeature_Oversampled1* dataset

7.2.3 Results Using the *CorrFeature_Oversampled2* dataset

In order to establish if these results can be further improved, the *CorrFeature_Oversampled1* dataset has been re-sampled again. In this instance, the dataset has been doubled from 126 to 252 memory blocks.

7.2.3.1 Classifier Performance

Table 13, below, illustrates the mean averages obtained over 100 simulations for the *sensitivity*, *specificity*, and *AUC*. These results are encouraging. The *sensitivity*, *specificity*, and *AUC* values, for all the algorithms, have increased again. Enlarging the dataset has further increased the algorithms ability to classify running. Table 14, illustrates the corresponding *k-fold* cross-validation results, which have also been improved. However, whilst the *mean error* rates have been significantly reduced, they are still higher than the minimum expected error rate of 11.11% (28 days per activity / 252 total records), although the rates are closer to 11.11% than in previous tests. Nevertheless, further improvement is required. The results indicate that further increasing the size of the dataset has improved the algorithms ability to classifying activities. Again, this has been expected, as there is more data for the algorithms to learn from and test against.

	Sensitivity	Specificity	AUC
Classifier	*HDM*	*HDM*	*HDM*
LDC	0.9620	0.9853	93.95%
QDC	0.7660	0.9783	97.79%
UDC	0.8700	0.9378	85.09%
POLYC	0.9820	0.9695	92.89%
LOGLC	0.9740	0.9738	94.00%
KNNC	0.8320	0.9445	98.43%
TREEC	0.8420	0.9745	99.42%
PARZENC	0.8800	0.9430	96.67%
SVC	0.8200	0.7915	91.78%
NAIVEBC	0.8420	0.9433	92.58%

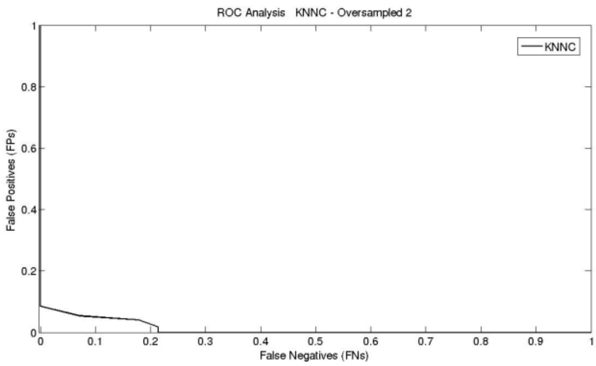
Table 13 Averages of Classifier Performance for the *CorrFeature_Oversampled2* dataset

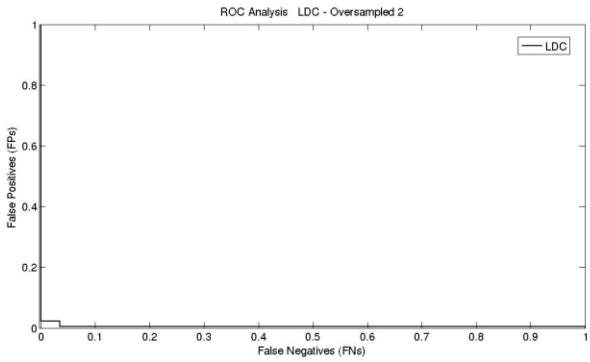
	80% Holdout: 100 Repetitions		Cross Val, 5 Folds, 1 Repetition	Cross Val, 5 Folds, 100 Repetitions	
Classifier	*Mean Error*	*SD*	*Mean Error*	*Mean Error*	*SD*
LDC	0.1671	0.0454	0.1587	0.1620	0.0042
QDC	0.2071	0.1053	0.2937	0.2236	0.0383
UDC	0.4827	0.0812	0.4921	0.4859	0.0233
POLYC	0.1689	0.0547	0.1706	0.1704	0.0048
LOGLC	0.1447	0.0458	0.1468	0.1485	0.0135
KNNC	0.1858	0.0477	0.1746	0.1796	0.0069
TREEC	0.1744	0.0513	0.1825	0.1717	0.0073
PARZENC	0.1818	0.0578	0.1825	0.1806	0.0056
SVC	0.6853	0.0326	0.6825	0.6782	0.0193
NAIVEBC	0.2844	0.0494	0.2817	0.2834	0.0058

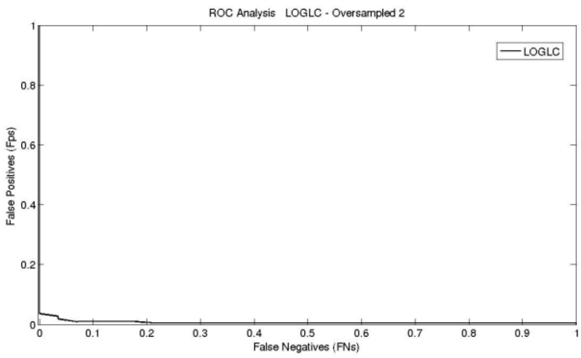
Table 14 Cross Validation Results for the *CorrFeature_Oversampled2* dataset

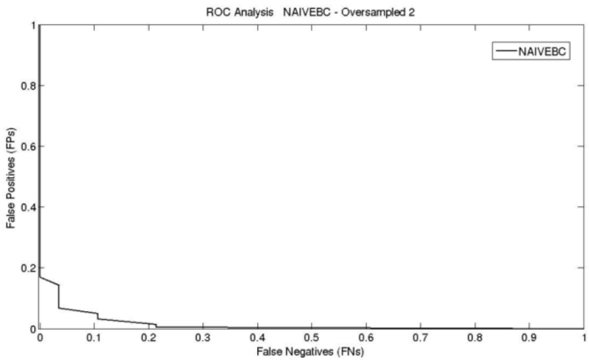
7.2.3.2 Model Selection

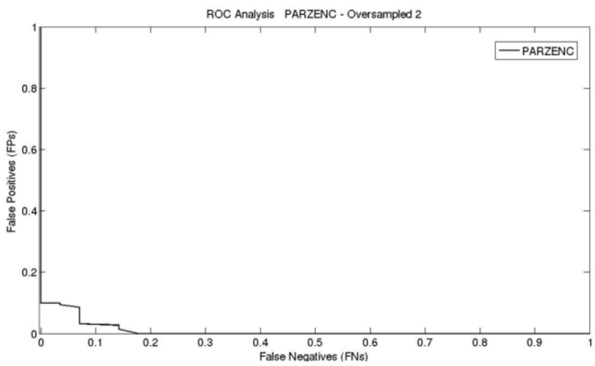
Again, the ROC curves show the cut-off values for the false negative and false positive rates. Compared to Figure 83, Figure 84 shows a significant improvement, for all of the classifiers.

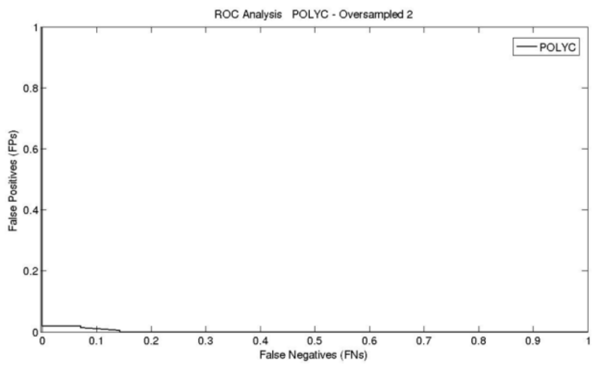












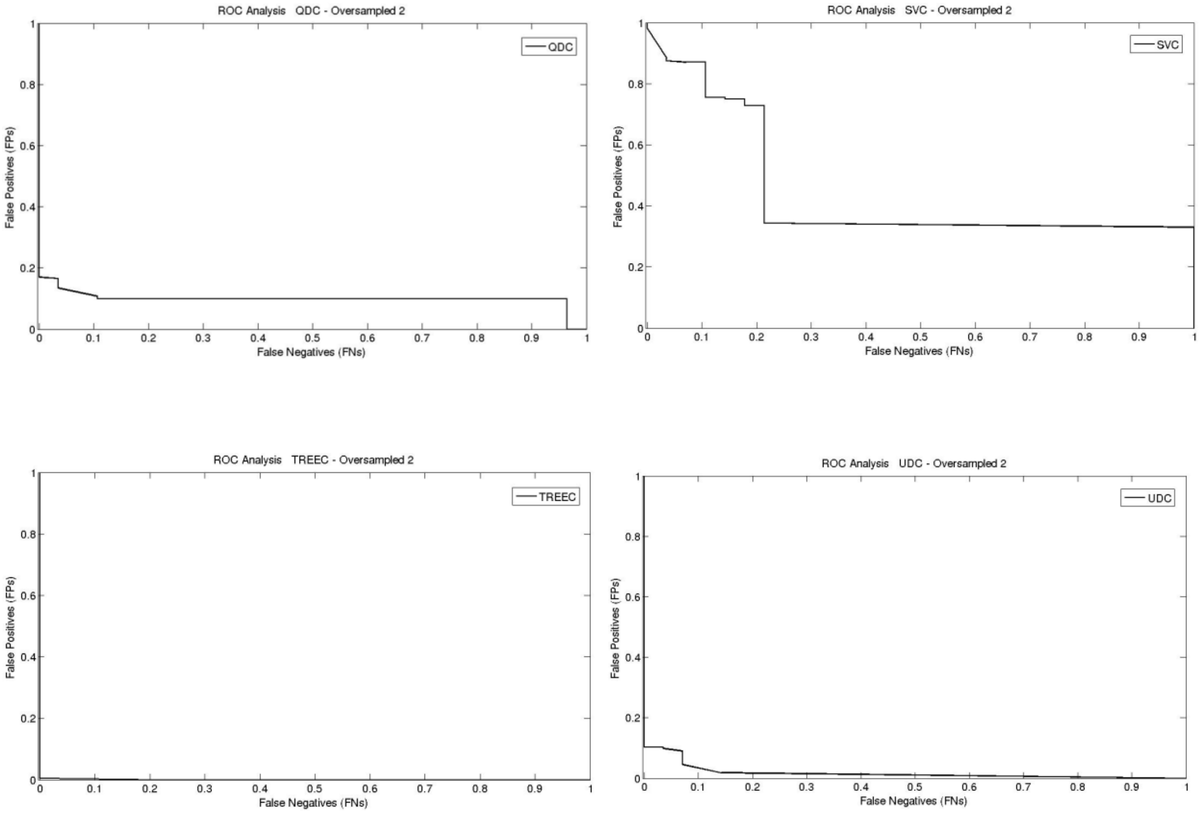


Figure 84 Receiver Operator Curves for the *CorrFeature_Oversampled2* dataset

Once again, the high *AUC*, *sensitivity* and *specificity* values in Table 13 support these findings. In comparison to Table 11, *UDC*'s accuracy has improved by 10.30%, whilst *LDC*'s has improved by 5.82%. *SVC*'s *sensitivity* has improved by 20%, whilst *QDC*'s has improved by 18.10%. *Specificity* has improved slightly, with *UDC*'s improving by 5.59% and *KNNC* by 4.95%.

7.2.4 Results Using the *CorrFeature_Oversampled3* dataset

In order to establish if these results can be further improved, the *CorrFeature_Oversampled2* dataset has been re-sampled again. In this instance, the dataset has been doubled once more from 252 to 504 memory blocks.

7.2.4.1 Classifier Performance

Table 15, below, illustrates the mean averages obtained over 100 simulations for the sensitivity, *specificity*, and *AUC*. These results are encouraging. The *sensitivity*, *specificity*, and *AUC* values, for most of the algorithms, have increased again. However, this approach did not work too well for *SVC*, whose *sensitivity* was 0.0000. Nevertheless, further enlarging the size of the dataset further increased most algorithms ability to classifying running.

	Sensitivity	Specificity	AUC
Classifier	*HDM*	*HDM*	*HDM*
LDC	0.9827	0.9894	96.78%
QDC	0.9118	0.9907	98.75%
UDC	0.9164	0.9634	90.68%
POLYC	0.9836	0.9907	96.06%
LOGLC	0.9818	0.9848	96.98%
KNNC	0.9391	0.9718	99.32%
TREEC	0.9364	0.9860	99.80%
PARZENC	0.9273	0.9711	98.29%
SVC	0.0000	0.9955	95.46%
NAIVEBC	0.9209	0.9694	95.35%

Table 15 Averages of Classifier Performance for the *CorrFeature_Oversampled3* dataset

Table 16, below, illustrates the corresponding *k-fold* validation results, which have been improved. However, another anomaly has occurred with *POLYC*, whose standard deviation result was slightly uncoordinated with the rest of the results. Nevertheless, the *mean error* rate, for most of the classifiers, is below the minimum expected error rate of 11.11% (56 days per activity / 504 total records).

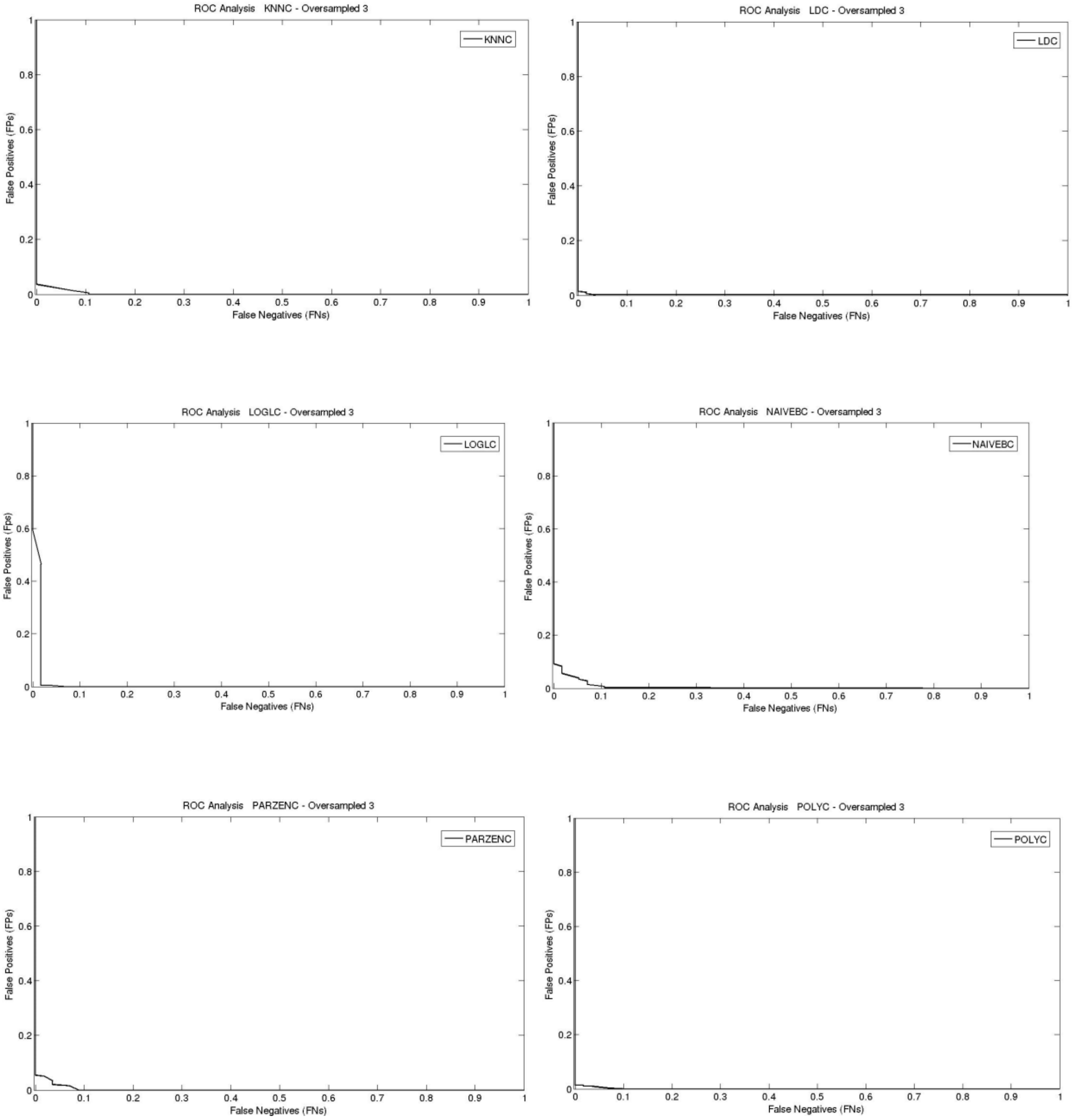
	80% Holdout: 100 Repetitions		**Cross Val, 5 Folds, 1 Repetition**	**Cross Val, 5 Folds, 100 Repetitions**	
Classifier	*Mean Error*	*SD*	*Mean Error*	*Mean Error*	*SD*
LDC	0.0821	0.0225	0.0833	0.0815	0.0019
QDC	0.1172	0.1074	0.2460	0.1297	0.0494
UDC	0.3444	0.0675	0.3214	0.3319	0.0254
POLYC	0.0872	0.0223	0.0873	0.0873	1.3948E-16
LOGLC	0.0734	0.0223	0.0754	0.0879	0.0335
KNNC	0.0879	0.0239	0.0893	0.0899	0.0032
TREEC	0.0831	0.0296	0.0853	0.0877	0.0038
PARZENC	0.0939	0.0291	0.0972	0.0923	0.0032
SVC	0.5642	0.0674	0.5734	0.5672	0.0286
NAIVEBC	0.1997	0.0244	0.1964	0.1972	0.0030

Table 16 Cross Validation Results for the *CorrFeature_Oversampled3* dataset

The results indicate that enlarging the size of the dataset has improved the algorithms ability to classifying activities. Again, this has been expected as there is more data for the algorithms to learn from and test against.

7.2.4.2 Model Selection

Again, the *ROC* curves show the cut-off values for the false negative and false positive rates. Figure 85, shows a significant improvement, for all of the classifiers, compared to previous evaluations.



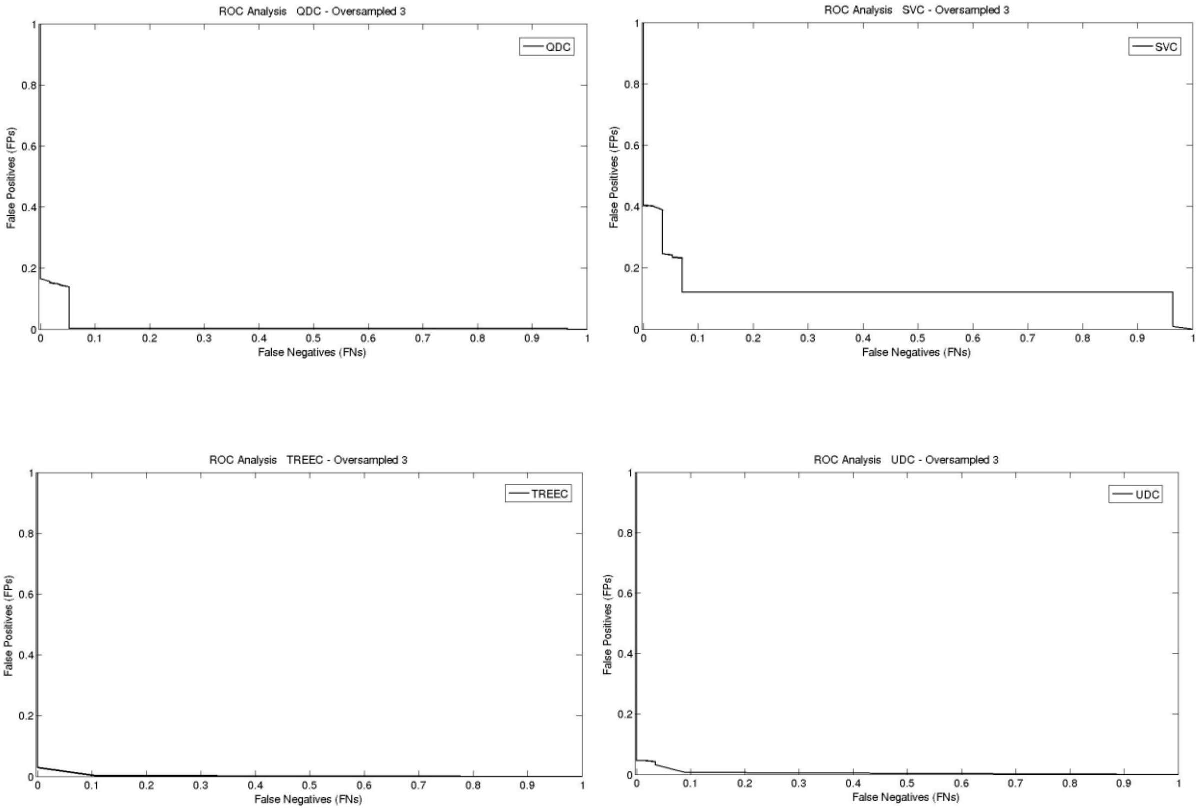


Figure 85 Receiver Operator Curves for the *CorrFeature_Oversampled3* dataset

Once again, the high *AUC*, *sensitivity* and *specificity* values in Table 15 support these findings. In comparison to Table 13, *UDC*'s accuracy has improved by 5.59%, whilst *POLYC*'s has improved by 3.17%. *QDC*'s sensitivity has improved by 14.58%, whilst *KNNC*'s has improved by 10.71%. Specificity has improved slightly, with *PARZENC*'s improving by 2.81% and *KNNC* by 2.73%.

7.2.5 Summary of Results

As it was expected, oversampling the data has drastically improved all of the performance results. This is due to more records being available for learning and testing. As the size of the dataset increased, the overall accuracy, *sensitivity* and *specificity* rates have improved. Comparing the original dataset to the final dataset, *PARZENC*'s accuracy has improved by 21.70%, whilst *LDC*'s has improved by 21.63%. In terms of *sensitivity*, *TREEC*'s has improved dramatically by 65.64%, whilst *KNNC*'s has increased by 57.91%. Regarding *specificity*, *POLYC* and *NAIVEBC* have increased by 15.57%, whilst *PARZENC*'s has improved 12.86%. The *mean error* rates have also dropped significantly, with the final oversampled dataset achieving an error rate below the expected 11.11%, for most of the classifiers.

The results indicate that the use of supervised machine learning techniques is encouraging. The ability to ask questions of our data is very interesting. Simply searching this data with specific queries, or keyword searches, does not produce the level of detail that is required or enable such specific questions to be answered. As demonstrated, supervised machine learning algorithms are able to treat the challenge of searching this data as a classification problem and to retrieve information based on features. However, whilst this method is a good starting point in relation to answering life questions, constructing memory boxes requires a different approach

7.3 Unsupervised Machine Learning Results

This section presents the results for searching human digital memory data, using the *CorrFeature* dataset and *k-means* clustering. In this example, questions are asked about our data in order to cluster periods of high or low energy. This enables the user to gather more of an insight into their behaviours. In order to demonstrate this idea, the first question that was asked was, "When have I been the most active?" and the second being "When have I been the least active?"

7.3.1 Results Using Original *CorrFeature* dataset

The first unsupervised evaluation uses the original *CorrFeature* dataset to answer the question "When have I been the most active?" The value of *k* has been defined in section 6.4 Clustering. Figure 86, below, illustrates the results from this question. In all examples, the centre of each cluster has been marked by the "Centroids" symbol.

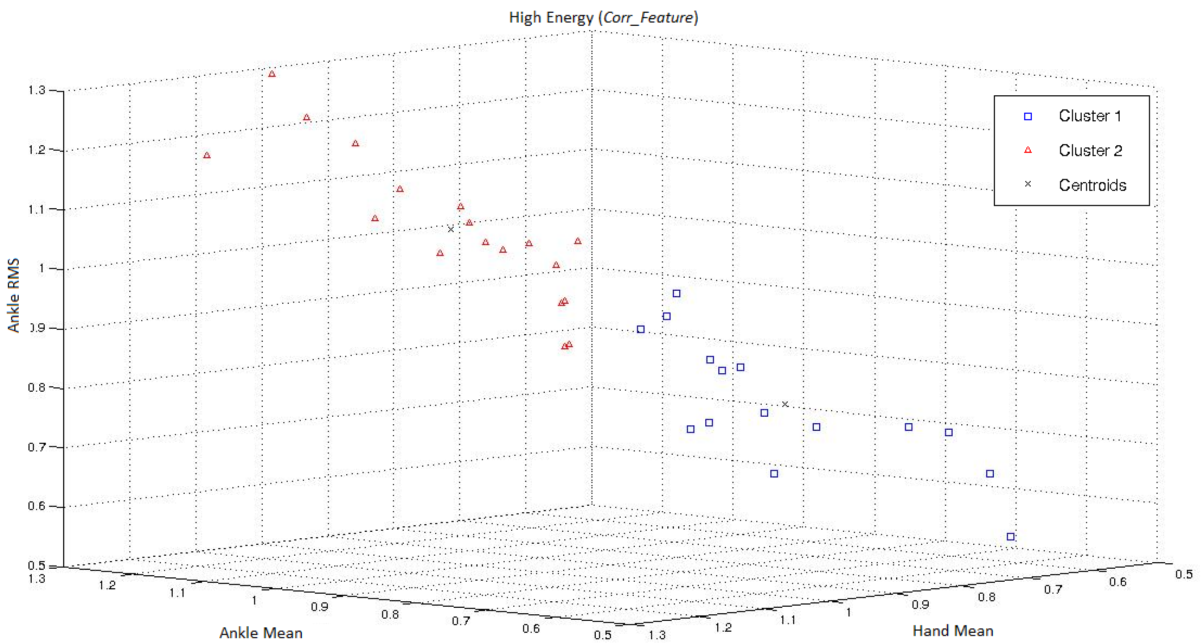


Figure 86 *K-means* Analysis of the Original *CorrFeature* Dataset – When Was I Most Active?

As can be seen, there is a clear divide in the data. The majority of the activities are located within cluster two and relate to walking, running and ascending stairs. Cluster one refers to descending stairs and vacuum cleaning. Since cluster two's activities do exert more energy than cluster one, the algorithm has correctly grouped the information into activities that have higher energy, since walking, running and ascending stairs use additional energy as opposed to descending stairs and vacuum cleaning. This has also been confirmed by analysing the dataset directly.

Using the original *CorrFeature* dataset, the question "When have I been the least active?" has also been asked. Again, the value of *k* has been defined in section 6.4 Clustering. Figure 87, below illustrates the results from this question.

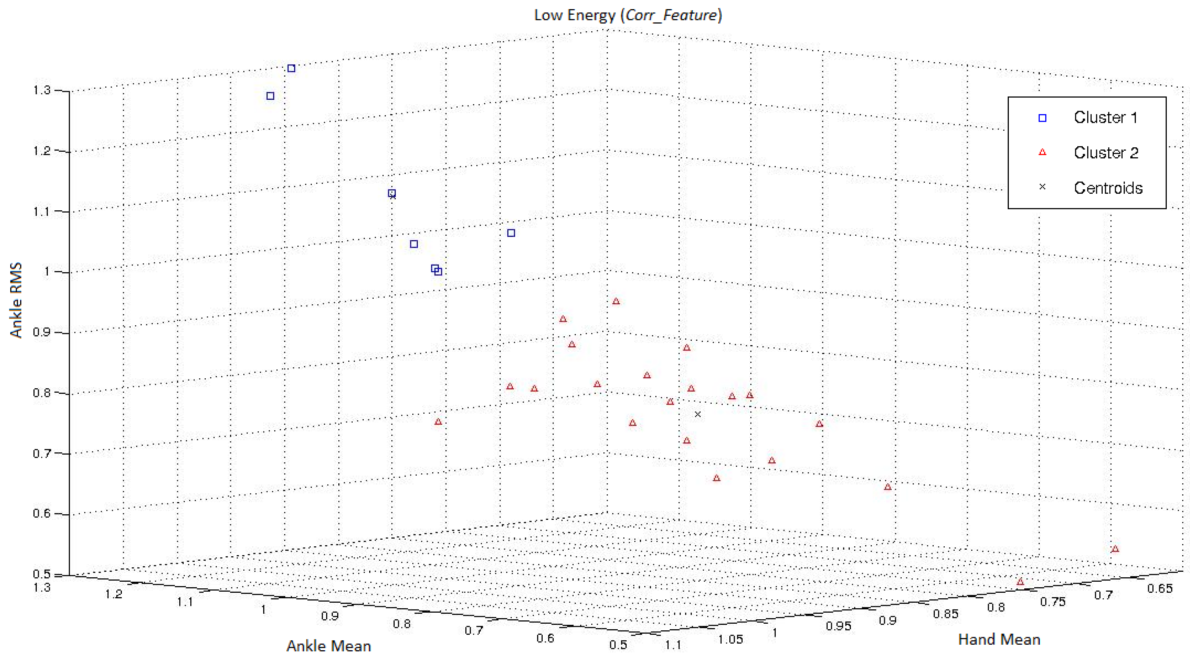


Figure 87 *K-means* Analysis of the Original *CorrFeature* Dataset – When Was I Least Active?

As can be seen, there is a clear divide in the data. The majority of the activities are located within cluster two and relate to sitting, standing and ironing. Cluster one refers to lying down only. Once again, the algorithm has correctly grouped the information into activities. Cluster two's activities, although classed as 'low-energy', all have higher energy than cluster one (lying down). It should be noted that for all clustering results, the activity labels were withheld, as clustering uses unlabelled data, and were only later added after as references to ascertain the clusters of each activity. This has also been confirmed by also analysing the dataset directly.

In order to present a well-rounded evaluation, the oversampled SMOTE datasets (used in section 7.2 Supervised Machine Learning Results), have also been evaluated with the clustering algorithm.

7.3.2 Results Using the *CorrFeature_Oversampled1* dataset

The next unsupervised evaluation uses the *CorrFeature_Oversampled1* dataset to answer the same question "When have I been the most active?" In order to find the optimum number of clusters, the *SA*'s have been calculated, for two to ten clusters. Please refer to Appendix 4.1.1 for the *SA* table and corresponding silhouette plot. Figure 88, below, illustrates the results from this question.

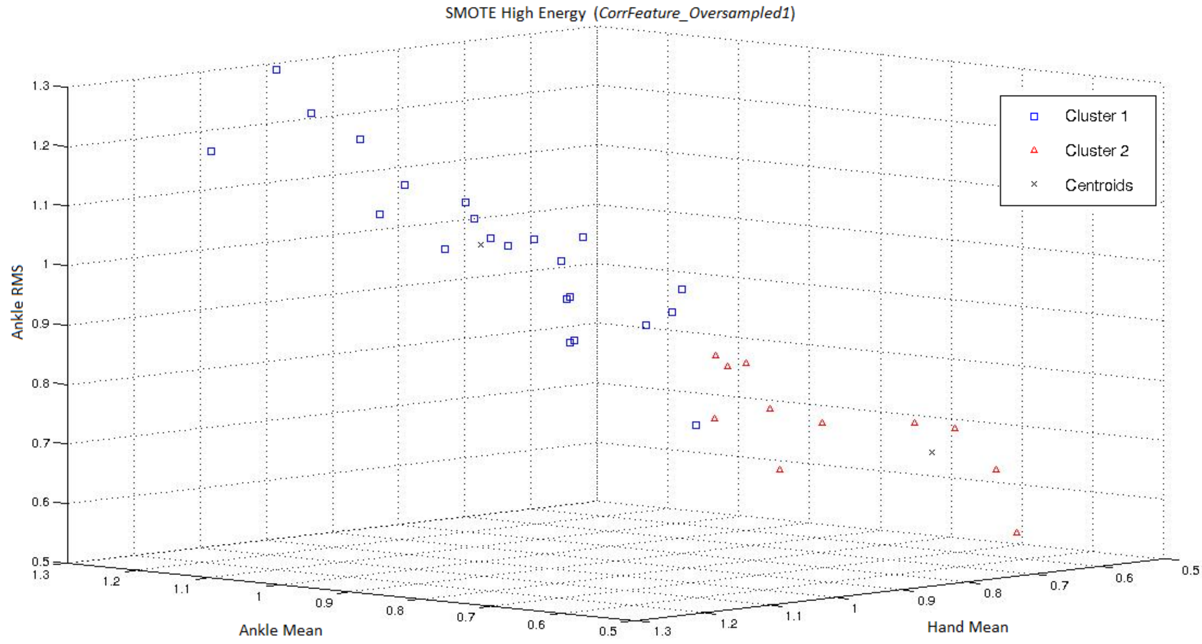


Figure 88 *K-means* Analysis of the *CorrFeature_Oversampled1* dataset – When Was I Most Active?

As it can be seen, again there is a clear divide. Once again, the majority of the activities are located within cluster two and primarily relate to descending stairs and vacuum cleaning, with a few instances of walking and ascending stairs. Cluster one mainly refers to running, walking, and ascending or descending stairs. Even though there is some overlap in the clusters and there is repetition in the data, cluster one's activities do exert more energy than cluster two; therefore, the algorithm has correctly grouped the information into activities that have higher energy. This has also been confirmed by analysing the dataset directly.

Using the *CorrFeature_Oversampled1* dataset, the question "When have I been the least active?" has also been asked. Please refer to Appendix 4.1.2 for the *SA* table and corresponding silhouette plot. Figure 89, below, illustrates the results from this question. As it can be seen, again, there is a clear divide in the data. The majority of the activities are located within cluster two and primarily relate to lying down, with an overlap of the other activities. Cluster one refers to sitting, standing and ironing. This has also been confirmed by analysing the dataset directly. Cluster one's activities, although classed as 'low-energy', all have higher energy than cluster two and so it was expected that cluster one should have higher averages. In this occurrence, the algorithm did not perform exactly as expected.

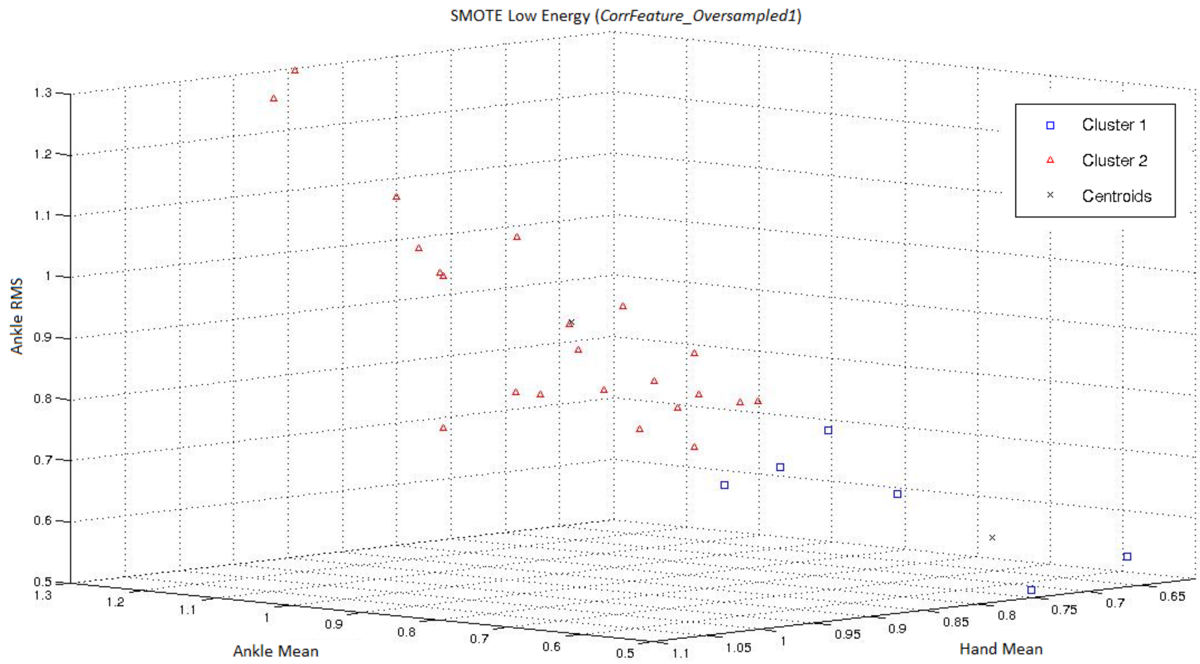


Figure 89 *K-means* Analysis of the *CorrFeature_Oversampled1* dataset – When Was I Least Active?

Upon directly analysing the dataset it has been established that the majority of activities in cluster one have repeated values. For this reason, cluster one is smaller than cluster two as there are few unique values to cluster on, with lower values. Oversampling the dataset has resulted in a significant portion of the data being repeated, with lower values.

7.3.3 Results Using the *CorrFeature_Oversampled2* dataset

In order to establish if these results can be further improved, the *CorrFeature_Oversampled2* dataset has been used to answer the question, "When have I been the most active?" Please refer to Appendix 4.2.1 for the *SA* table and corresponding silhouette plot. Figure 90, below, illustrates the results from this question.

In this demonstration, cluster one primarily relates to running, with a few instances of other activities. Cluster two relates to walking, ascending or descending stairs, and vacuuming. Again, even though there is some overlap in the clusters and there is repetition in the data, cluster one's activities do exert more energy than cluster two; therefore, the algorithm has correctly grouped the information into activities that have higher energy. This has also been confirmed by analysing the dataset directly.

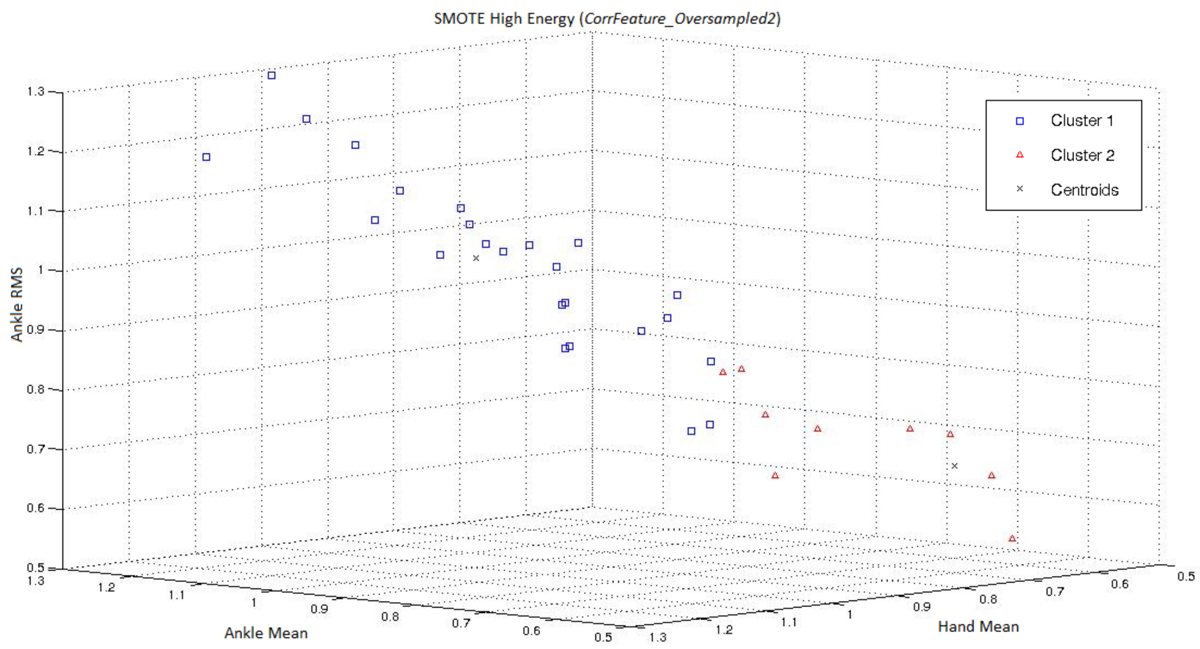


Figure 90 *K-means* Analysis of the *CorrFeature_Oversampled2* dataset – When Was I Most Active?

Using the *CorrFeature_Oversampled2* dataset, the question "When have I been the least active?" has also been asked again. Please refer to Appendix 4.2.2 for the *SA* table and corresponding silhouette plot. Figure 91, below, illustrates the results from this question.

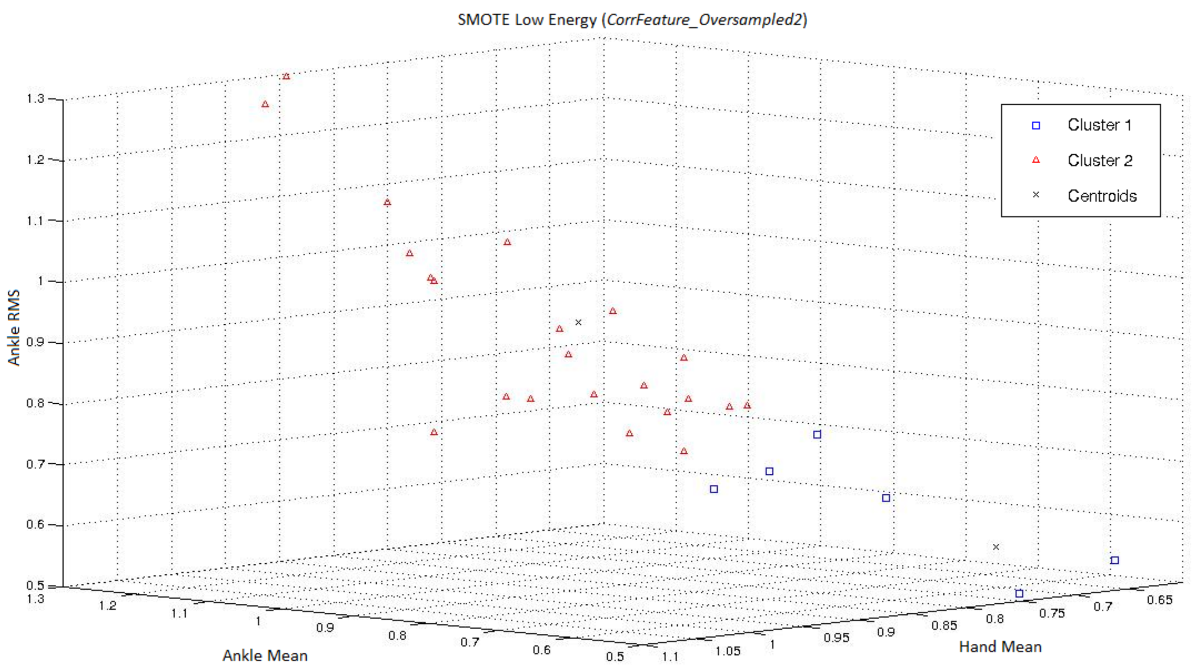


Figure 91 *K-means* Analysis of the *CorrFeature_Oversampled2* dataset – When Was I Least Active?

In this demonstration, cluster two primarily relates to lying down, with an overlap of other activities. Cluster one refers to sitting, standing and ironing. This has been confirmed by analysing the dataset directly. Again, cluster one's activities, although classed as 'low-energy', all have higher energy than cluster two and so it was expected that cluster one should have higher averages. Again, the algorithm did not perform exactly as expected. Similarly, like the previous oversampled example, the majority of activities in cluster one have repeated values. For this reason, cluster one is smaller than cluster two as there are few unique values to cluster on, with lower values. Oversampling the dataset again has resulted in a significant portion of the data being repeated and the resulting data having lower values.

7.3.4 Results Using the *CorrFeature_Oversampled3* dataset

Again, in order to establish if these results can be further improved, the *CorrFeature_Oversampled3* dataset has been used to again answer the question, "When have I been the most active?" Please refer to Appendix 4.3.1 for the *SA* table and corresponding silhouette plot. Figure 92, below, illustrates the results from this question.

In this demonstration, cluster two primarily relates to running, with a few instances of other activities. Cluster one relates to walking, ascending or descending stairs, and vacuuming. Once more, even though there is some overlap in the clusters and there is repetition in the data, cluster two's activities do exert more energy than cluster one; therefore, the algorithm has correctly grouped the information into activities that have higher energy. This has been confirmed by analysing the dataset directly.

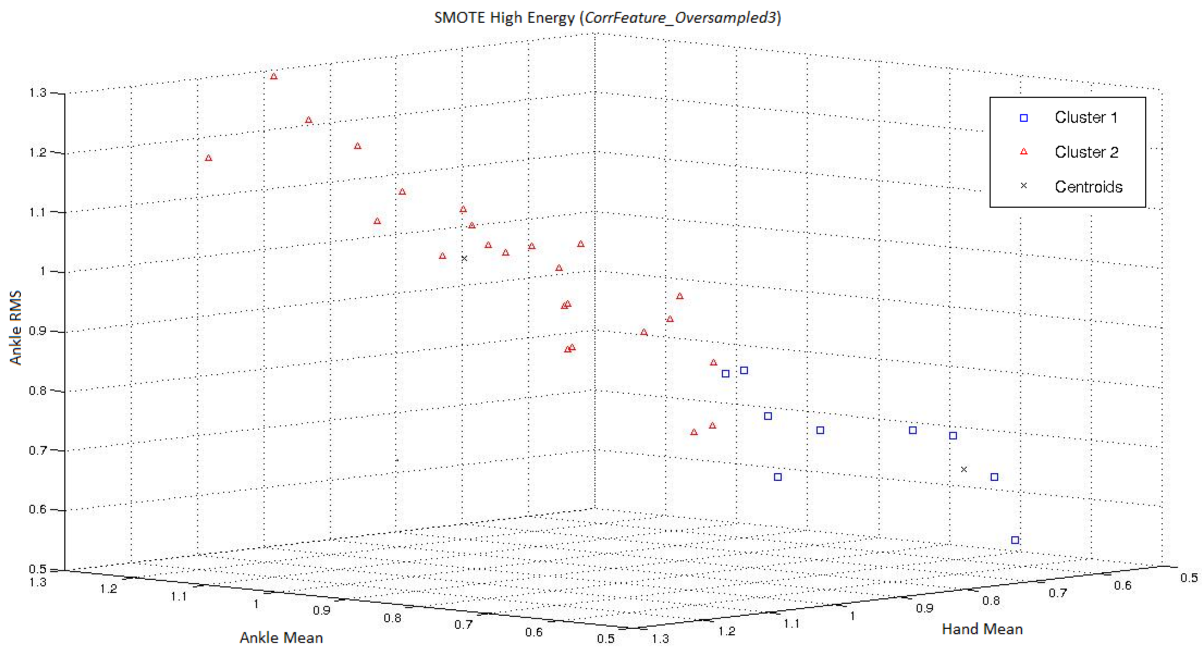


Figure 92 *K-means* Analysis of the *CorrFeature_Oversampled3* dataset – When Was I Most Active?

Using the *CorrFeature_Oversampled3* dataset, the question "When have I been the least active?" has also been asked again. Please refer to Appendix 4.3.2 for the *SA* table and corresponding silhouette plot. Figure 93, below, illustrates the results from this question.

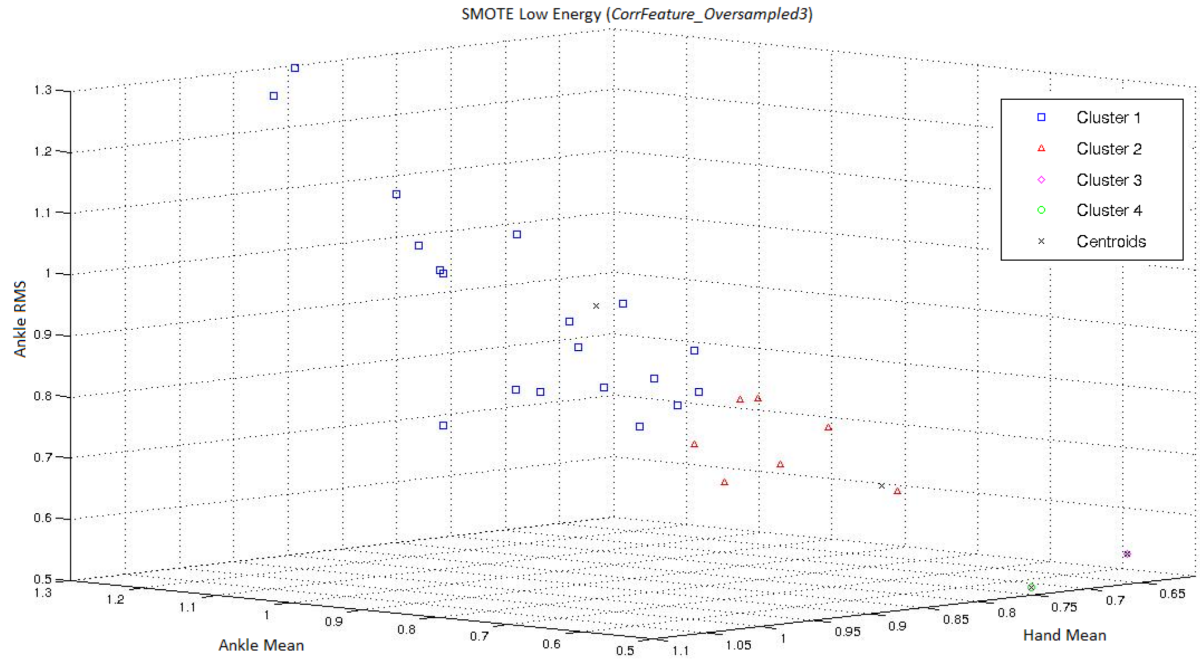


Figure 93 *K-means* Analysis of the *CorrFeature_Oversampled3* dataset – When Was I Least Active?

In this demonstration, four clusters have been used, as this has been deemed the appropriate number to use by the *SA* table (see Appendix 4.3.2). After analysing the dataset directly, cluster one mainly represents lying down, cluster two mainly sitting, cluster three refers only to standing and cluster four only to ironing. However, due to oversampling the data, the data within clusters three has been replicated, as has the data in cluster four. This is illustrated within the graph, as the clusters are not very varied.

7.3.5 Summary of Results

Overall, these results support the idea that clustering data is a viable method of creating memory boxes. However, concerning clustering, oversampling the data has not proven to enhance the results, as it did for classification. This is because, in classification, more results are required to enable the system to learn from and test against to distinguish activities. Alternatively, in clustering, duplicating the results serves little purpose and distorts the clusters as there are not many unique values to create individual points from which the algorithm is able to use to create bigger clusters. By using, the oversampled datasets in both evaluations for classification and clustering provides a well-rounded evaluation of both methods.

Referring to the results from the original dataset (see section 7.3.1 Results Using Original *CorrFeature* dataset), these graphs clearly illustrate the periods of time that the user has spent being active or inactive. Simply searching this data with specific queries, or keywords, does not produce the level of detail required. As demonstrated, unsupervised machine learning is able to treat the challenge of searching this data as a clustering problem and to retrieve information based on features. The algorithm has also been successful in dividing the data into clear groups of high and low energy activities. Although these times can now be clearly seen, we still do not know exactly what the user was doing. In this instance, the information that has been obtained is transformed into temporal memory boxes of human experiences, as previously discussed in section 4.2. Prototype Configuration.

7.4 Discussion

Most human digital memory systems search data based on complicated queries. As the amount of information increases, these methods become increasingly limited. This surge in both the volume and the variety of data requires advances in methodologies that automatically understand, process, and summarize the data (Jain 2010). Machine learning is seen as a technique to overcome the shortcomings of traditional search techniques so that this collection of 'big data' can be effectively analysed.

This chapter demonstrates the results that have been obtained by treating searching of human digital memory data as both a classification and clustering problem. In terms of classification, the dataset has been separated with 80% for training and 20% for testing. As it can be seen from the results in section 7.2 Supervised Machine Learning Results, this approach yields some positive and interesting results. Using the SMOTE (Chawla et al. 2002) technique to oversample the dataset enabled the classification results to be improved. The system has more data to learn about the features of each activity and test this against the training set. Overall, several of the classifiers performed well. These include *TREEC*, with 93.64% *sensitivity*, and an overall accuracy of 99.80% and *KNNC* with 93.91% *sensitivity* and an overall accuracy of 99.32%. These results also produced significantly better results than those in (Qiu et al. 2011) who report an accuracy ranging from 90% to 98%. In this approach, various activities have been undertaken whilst the user is wearing a *SenseCam*. Using the data from the on-board single tri-axil accelerometer activity is then classified, whilst the photos provide the visual evidence of these events. For each of the three axes, the authors have used the features raw acceleration data, standard deviation and range to classify the activities, totalling nine features. This is in contrast to our approach where three tri-axil accelerometers have been placed around the body to capture the acceleration of the ankle, chest and hand. This provides a better illustration of the entire body's movements, as opposed to a single reading from an accelerometer positioned around the user's neck. Furthermore, location and heart data have also been utilised, which provides more information to reason over. Our method also generates a larger feature set and enables significant features to be extracted. Although we have classified activity using only four features (ankle mean, hand mean, ankle root mean square and hand root mean square) these features have been selected from the original dataset and are deemed to contain the most information, which can be used to describe the entire dataset. In contrast, the authors use only the raw data from each accelerometer's axis, standard deviation and range, which produces a feature set that is limited. The data has not been analysed to determine if these features are significant. Furthermore, the inclusion of raw accelerometer data is susceptible to noise and generally needs to be pre-processed before classification occurs.

The approach then used the Support Vector Machine (*SVM*) algorithm to classify the data. This algorithm was chosen "*given its widespread use in classifying accelerometer-based activity*" (Qiu et al. 2011). However, it is not known if the results could have been improved if other algorithms had been tested. In our approach the evaluation of a number of algorithms provides a better method of determining the best algorithm to use. In contrast to the *SVM*, our approach produced better results using the *TREEC* and *KNNC* algorithms. As opposed to *KNNC* and *TREEC*, *SVM*s suffer from overfitting, which can cause poor performance (Kotsiantis 2007). By utilizing more data sources, pre-processing techniques, features selection and different machine learning algorithms have helped to produce better results.

As the dataset increased the results improved, which was expected as there was more data for the algorithm to learn from. In terms of answering life questions, these results are promising and provide a valid method of classifying behaviour that can be extended, for instance, to include emotion data. In this instance, once the system learns the features of being happy, sad, angry, etc. emotional data can be classified and a greater understanding of our data and ourselves will begin to emerge. However, in terms of creating memory boxes, classification is not a viable approach since the method relies on using labelled training data and predetermined classes in order to structure information. In the case of creating memory boxes, and to find information without having prior knowledge of its features, another method is required. In this instance, unsupervised machine learning algorithms can be used to explore the data, without the need for predetermined categories.

In terms of the clustering approach, the SMOTE (Chawla et al. 2002) technique did not improve the results. The size and shapes of the clusters were not improved. However, the original dataset did provide some interesting results. As it can be seen, there is a clear divide in the data at times of high and low activity. These results can be used to determine, for example, periods of inactivity and the memory box provides the visual illustrations of these times and the factors that contribute to these periods of inactivity. By treating the searching of human digital memory data as a clustering problem removes the need to label the feature vectors. Letting the algorithm cluster similar pieces of data together eliminates the need to have a pre-existing knowledge of the data. Moreover, the system is not learning about the user's memories; therefore, testing and training sets are not required, as in the case of classification. Additionally, this method is beneficial as it overcomes the limitations of searching data with complex query languages such as SPARQL. By transforming the raw data into human digital memory vectors and treating the searching of this data as a clustering problem eliminates the need to have a pre-existing knowledge of the dataset. Furthermore, these vectors can become extremely large, especially if a lifetime of data is being recorded. In spite of this, clustering algorithms are able to deal with these sets of big data quite easily. By transforming extremely large datasets of raw data into features enables the human digital memory vectors to be rich with information, the bigger the feature space, the more detailed a memory is.

These two methods complement each other perfectly. Classification is used to ask initial questions about the data, such as "Have I been running?" Once an answer has been retrieved, for example, "yes the user has been running," then clustering is used to bring together those periods of running so that a memory box can be generated. These results are very exciting and demonstrate the validity of the approach. By enabling, the user to ask questions and cluster these moments in time allows a deeper reflection of ourselves to be undertaken.

Overall, the results support the use of machine learning as a viable approach for searching and analysing human digital memory data. Classification is more useful for analysing data so that the system can learn about our lives, which could then be used to predict behaviour. However, in terms of constructing memory boxes, clustering offers a better approach. This method is especially useful as the human digital memory vectors increase in size and it becomes more difficult to keep track of the memory features. By letting the clustering algorithm find the natural groupings within the data ensures that all related pieces of information are brought together.

7.5 Summary

In this chapter, the results from our evaluation have been demonstrated. This evaluation focused on treating the searching of human digital memory data as a machine learning problem and has been successful in addressing the searching challenges identified in section 1.3 Research Challenges. This work paves the way for creating fully interactive human digital memories. Digital memories grow alongside their human counterparts. A lifetime of information can be collected, thus producing extraordinarily large datasets, which can be reasoned over. In this field, more advanced search techniques are needed. Machine learning techniques are seen as a viable solution, where dealing with big datasets is not unusual. These results are encouraging and support the use of machine learning in searching human digital memory data. Arguably, a greater monitoring period is needed. Future work would need to consider using a far larger dataset, which would contain more human digital memories, over a much bigger time span, for example, a months' worth of data rather than days or hours. This chapter has also demonstrated how DigMem compares to other systems. This comparison has illustrated that

DigMem outperforms current methods. In particular, its extendable nature, the types of data that are able to be collected, and the supported search methods outperform other approaches.

The following chapter concludes the thesis by presenting a summary of the project, and the future direction that the work can take. The chapter also discusses how this work has contributed to knowledge and provides concluding remarks about the project.

Chapter 8

Summary and Future Work

In the previous chapter, the results from the evaluation have been presented. The evaluation focused on treating the searching of human digital memory data as a machine learning problem, as such, the results from several supervised and unsupervised algorithms have been presented. The chapter also demonstrated how oversampling the data improved some of the results. Furthermore, a comparison of DigMem against current lifelogging systems has been made. This comparison has illustrated that DigMem outperforms existing methods. In particular, its extendable nature, the types of data that are able to be collected, and the supported search methods outperform other approaches.

This chapter concludes the thesis and provides a summary of the work, discusses the contribution to knowledge that has been made and how the challenges identified in section 1.3, Research Challenges have been addressed. The future research directions of the project are then discussed and the thesis is concluded with some final remarks about the project.

8.1. Thesis Summary

Human memories link past experiences with the future and are a very powerful tool that people have at their disposal. However, as people get older the ability to remember events with the same precision, for example, as they did ten years ago, decreases. Digitizing this process enables a completely new way in which memories can be retained and provides a platform where people can relive their past. As technology advances, the landscape of our world is rapidly changing. Devices prevalent within the environment are now able to capture a multitude of data, both physiological and ecological. Sensors embedded in everyday objects, and biological monitoring devices produce a phenomenal amount of information; however, there has been very little development in bringing these items together for the purpose of building human digital memories. Furthermore, current implementations are limited in the data that can be collected. In order to provide a well-rounded human digital memory a variety of data is required so that a more vivid memory can be created. In realising these challenges, this thesis has focused on presenting the DigMem system, which is able to learn about the user and to produce rich and interactive human digital memories (memory boxes). This has been achieved by using a variety of devices and technologies, including pervasive devices, P2P networks, cloud computing, semantic web technologies and machine learning.

Chapter one provided an overview of human memories and the concept of human digital memories and lifelogging. It also describes how the generation of data has advanced and examines how such digital memories are created. As well as an overview of the area, this chapter describes the research challenges that have been identified. These include the challenge of identifying what a human digital memory is composed of, how a lifetime can be effectively searched and the structure of a human digital memory. It also describes the novel contributions that the project has provided and the structure of the thesis.

The background has been presented in chapter two. This included examining how memory data is captured, through the use of wearable lifelogging systems, mobile technologies, physiological and smart devices. Further research has been undertaken into examining how data is currently being searched. This included providing an overview into examining linked data and machine learning. Furthermore, an examination into how memory data has been organised is also presented. In summary, this chapter provides an overview of the state-of-the-art into capturing, searching and creating human digital memories.

The design of the DigMem system has been presented in chapter three. This has included an overview of the design specifications and architecture of the system. The chapter also includes a detailed design of the MoDM, DigMem Server and DigMem Web Application components of the system. Various Unified Modelling Language (UML) diagrams have been constructed to illustrate the design of each element of the system. As well as the UML design, the DigMem Web Application has also been modelled with the use of a site map and several storyboards that illustrate the design of each page.

In chapter four, the implementation of the system has been described. This chapter focused on describing how the main components of the system were implemented. This section describes the tools that were used to implement DigMem and presents a demonstration of how memory boxes are created. This chapter has demonstrated the idea that human digital memories can be created using pervasive devices, linked data and machine learning. The main aim of this chapter has been to address the research challenges, outlined in Chapter one, and to validate the Design (chapter three).

In chapter five, a case study has been presented that illustrates how DigMem collects data in a smart home and demonstrates a memory box and its capabilities. These studies are encouraging and do validate the design, and idea, that human digital memories can be created using every day, pervasive, devices and linked data. Other than recalling specific experiences, this chapter also examines how the system can be used in other areas. It illustrates how DigMem can be used to monitor and reduce sedentary behaviour, as well as aiding autobiographical memory in patients with cognitive impairments, as well as how activity and emotional states can be recognised and inferred.

Chapter six has discussed the evaluation methodology that has been used to pre-process the raw human digital memory data. Features have then been extracted and analysed, using *PCA* and a correlation matrix. The resulting *CorrFeature* Dataset has then been oversampled, using SMOTE, to generate an enlarged data set. This chapter has also discussed the supervised and unsupervised machine learning algorithms that have been chosen for the evaluation.

The system has been evaluated in chapter seven. This has been done by presenting the results of several supervised and unsupervised machine learning algorithms. These results demonstrate how the system searches data to answer life questions and to create memory boxes. As well as these results, a comparison between the state-of-the-art lifelogging devices and DigMem has also been made. This chapter highlights the benefits of the DigMem system and illustrates how human digital memory data sets can be effectively searched.

8.2. Contribution to Knowledge

A great deal can be learnt from this project. The MoDM middleware platform offers a number of advantages over current, proprietary, systems (Dobbins, Merabti, et al. 2012b). Currently, devices, such as the *SenseCam* or *SenseWear Armband (SWA)*, focus on only collecting specific information, such as photos or physiological information. These approaches limit the resulting memory. As previously stated in section 1.3 Research Challenges "*Instead of focusing on collecting a specific set of data items, the process of capturing human digital memory data needs to be flexible enough to adapt to the user's current situation, utilizing a variety of information sources.*" The MoDM middleware addresses this point by adapting to the user's present environment and enabling any MoDM compliant device to be used as a source of information, thus not limiting the data that is collected and enables the components of a human digital memory to be increased. This has been achieved by creating a P2P network where the user's mobile device can search for services from external devices. Once connected, information is automatically sent to the user and saved in an Internet-connected folder. Therefore, devices present in one environment will differ to those of another. This dynamically changing set of peers can easily be accommodated. Furthermore, this implementation illustrates that the use of pervasive mobile devices can support the memory structures required. This is achieved by enabling a flexible open-source plug-and-play platform, for memory data sources, which can be exploited by the DigMem application.

This implementation is also more cost-effective than buying specialist equipment (for example, *SenseCam*) and reduces the number of devices that the user has to carry around with them. For example, as demonstrated in (Kikhia et al. 2010), this system "*requires the user to bring the relevant equipment and to have it turned on*". This is a very cumbersome method, especially if the user is older and has trouble remembering all the items that they would need. However, the DigMem system reduces the burden that is placed on the user, by using mobile devices, such as tablets and smartphones, and devices within the environment. As previously discussed, these are items that almost everyone carries around with them, daily. There is no need to buy specialist equipment as the data, required for the memories, is gathered remotely, from devices prevalent within the surrounding environment. This method is adaptable and extendable enough to adjust to any environment, thus allowing memories to be created over a lifetime, and with as little user interaction as possible. Furthermore, the ability to create and connect to a P2P network, publish services and allow external devices to connect to one another to gather and share data is also a unique feature that is not available with other implementations, due to them being closed systems. Additionally, the benefit of automatically uploading collected data into an Internet-connected directory has numerous benefits. Firstly, user intervention is drastically reduced. Once the MoDM application is started, it automatically collects data, saves this information in an Internet-connected directory, transfers this information into the Raw Data Store and frees up space on the device. This is in contrast to current devices where data has to be manually uploaded onto the system and storage is limited.

The use of RDF also enables data to be incorporated into a human digital memory, irrespective of its format and enables the longevity of these boxes to be increased and constructed, using a lifetime of data (Dobbins, Merabti, Fergus, Llewellyn-Jones, et al. 2013). As new standards become available their data, as well as data collected twenty years ago, for instance, are still able to be incorporated in a human digital memory. This flexible and very important feature is especially useful because, as Fitzgibbon and Reiter (Fitzgibbon & Reiter 2005) question, "*How can we ensure that data is still accessible in 50 years time, despite inevitable changes in software, hardware and formats?*" As demonstrated in this work, the use of RDF is seen as a way to address this question. As time goes on and new devices and formats emerge, they can still be incorporated into the memory box, even if the underlying device has become obsolete. This is another flexible and very important feature and as reiterated by the W3C (W3C 2004a), "*RDF has features that facilitate data merging even if the underlying schemas differ, and it specifically supports the evolution of schemas over time without requiring all the data* *consumers to be changed*". Memories created, in this way, are a "mash–up" of all the data amassed over an extensive period of time. Human digital memories grow alongside their human counterpart and the DigMem system is able to achieve this, using this method of data transformation.

When humans remember events, different information is processed, subconsciously, and without too much effort. As defined in this work, memories can be imagined as little "boxes" or "episodes" of events, which can be recalled in the mind. For example, in the case of recalling a birthday party that happened last week, recalling attendees, location, feelings, temperature (if it was particularly hot or cold), what was eaten and other pieces of information are quite easily remembered. This type of memory is known as "episodic memory", and relates to the memory of temporal periods of experienced events and episodes (Sutton 2010). However, trying to remember a birthday party that happened thirty years ago, recalling the same level of detail is a lot harder, as opposed to an event that happened recently. DigMem bridges this gap, so that any time of life can be easily recollected (Dobbins, Merabti, Fergus & Llewellyn-Jones 2013a). The DigMem system is flexible to adapt over time and a lifetime of data can be captured and turned into memory boxes. This method enables any piece of recorded information to be included in the human digital memory. This technique enables a richer human digital memory to be created, as the pool of resources widens, and more data can be included. Furthermore, memory boxes allow us to obtain inferences of events that are impossible from a single data source, reduces data overload and allows a lot of data to be transformed into a smaller amount of more meaningful information.

The use of machine learning algorithms enables a new way to explore human digital memory data (Dobbins, Merabti, Fergus & Llewellyn-Jones 2013b). Collecting data over an extensive period of time (for example, over a lifetime) yields an unprecedented amount of information. Due to its vastness, searching human digital memory data is challenging. This increase in both the volume and the variety of data requires advances in methodologies to automatically understand, process, and summarize the data (Jain 2010). Keyword searching is no longer a viable option for finding rich information. In overcoming this challenge, the evaluation of several supervised and unsupervised machine learning algorithms has been undertaken. Raw data is converted into a matrix of features, which is then classified and clustered. Classification enables questions to be asked about our data. This development has enabled an intelligent system to be created that is able to learn about its user in order to answer life questions, which is unique to this project. In terms of creating memory boxes, clustering is beneficial as data is extracted based on similarities. In this way, all related pieces of information are brought together to form a memory box. Feature-rich interactive memory boxes are constructed without the need to define complex and specific queries (Dobbins, Merabti, Fergus & Llewellyn-Jones 2013a). This is a very important and unique feature of the system that re-defines how human digital memory data is searched. As the human digital memory vectors increase in size, it becomes progressively more difficult to keep track of accumulated data, especially from ten or twenty years ago. Therefore, when defining explicit queries, it is easy to overlook pieces of information. However, the use of unsupervised algorithms overcomes this by clustering related pieces of data together. In this way, data items are not overlooked. The results indicate that both methods are a viable alternative to searching and deriving information from this type of data. The DigMem system is learning about the user and using what it has learnt to create memory boxes.

Furthermore, DigMem is mimicking, to a degree, the Serial Parallel Independent (SPI) model of human memory, which postulates process-specific relations among human memory systems (Tulving 1972; Tulving 2002). In this model, Tulving (Tulving 1972) states that, "*Information is encoded into systems serially, and encoding in one system is contingent on the successful processing of the information in some other system, that is, the output from one system provides the input into another. Information is stored in different systems in parallel*". DigMem mirrors this concept, to an extent. Creating a memory box relies on data successfully being stored, transferred and transformed from one layer to the next (Dobbins, Merabti, Fergus, Llewellyn-Jones, et al. 2013). The system is not intended to precisely re-create how human memory works, but as it can be seen, certain aspects are mirrored.

The ability to see various pieces of information, recorded from any time of our lives is also another unique feature of the system, as is having a system that learns and knows about its user. Data, collected over a lifetime, is semantically linked and any instance of our lives can be re-constructed. The DigMem system illustrates that by incorporating data from any pervasive device a greater level of detail is achieved. For example, the data from smart fridges can be used to quantify dietary habits, whilst the human body provides us with physiological data. Any object can become a data source, and their information used in a memory. This is particularly important because a richer level of detail is achieved and can be used to reason over behaviour and help us to understand aspects about our health, level of activity and physical wellbeing (Dobbins, Merabti, et al. 2012a). Questions about ourselves will begin to emerge, such as "*How was I feeling at x point in my life?*" or "*What factors made me feel like this?*" and "*How were others around me feeling at the same time?*" Any time, throughout our lives, can be reconstructed and our feelings, from those times, reasoned over. This is a very powerful feature of the system.

8.3. Future Work

This work paves the way for creating fully interactive human digital memories that can be used for reflection, over a lifetime. Our digital memories grow alongside their human counterparts. Fragmented pieces of data are semantically linked and transformed into visual items. This work provides exciting results in terms of how fully interactive human digital memories can be created and how such a system can be used to influence various aspects of our lives. However, whilst the results are exciting further work remains. The following is an overview of the future direction of this research.

8.3.1 DigMem Automation

Currently, the process of creating memory boxes or answering life questions is not an automated and seamless process. Regarding the web interface, additional work is required to integrate the classification and clustering algorithms into this aspect of the system. In this way, upon logging in, the user can select their question, and an instant response or memory box is presented. The evaluation proves that these algorithms are a viable method of searching data; therefore, integration into the user interface is the next step.

8.3.2 Privacy and Security

In any system that records personal information, privacy becomes an issue. Whenever Memories for Life is discussed with the general public or the non-scientific media, privacy and security issues are the most frequently raised talking points (Fitzgibbon & Reiter 2005). The European Network and Information Security Agency's (ENISA) report (European Network and Information Security Agency (ENISA) 2011) into the benefits and risks of lifeogging reiterates this point. In this report, they comment that, "*The top risk for individuals utilising life-logging devices and scenarios is the threat to privacy that accompany using them. Loss of control over this data might result in individuals being subjected to financial fraud or unauthorised access might result in reputational harm or discrimination and exclusion*" (European Network and Information Security Agency (ENISA) 2011). Fitzgibbon and Reiter (Fitzgibbon & Reiter 2005) also question, "*How should privacy be protected, especially when an individual is present in someone else's memory? For example, if Joe has a digital picture of Mary and he is holding hands, what rights does Mary have over the use and distribution of this picture?*" This is a very important point to consider, especially when photographic data is concerned, as inevitably, this type of information will contain the images of multiple users.

Furthermore, collecting a lifetime's worth of information undoubtedly produces a vast amount of data. Securing this collection becomes harder as it grows in size. In comparison to how much data is capable of being captured over a lifetime, the demonstrated system collects a small amount of information that is related to one user, which is retained in secure data stores. However, the issue of security will need to be addressed as the amount of data, and users, increases. Private areas will need to be established for each user, where they can store their information and choose what memories to share with others. Another issue, regarding privacy, relates to identity theft. If devices are stolen, and false memories created, then this affects the user's entire human digital memory store. Future work aims to look further into the issues regarding privacy and security so that memory data can be safely collected and stored.

8.3.3 Cataloguing Important Events

Collecting a lifetime's worth of human digital memory information undoubtedly produces a vast amount of data. Even with intelligent search, sifting through all of this data to find important moments can seem impossible. Highlighting and automatically tagging significant memories (events) poses quite a problem. How does a system know what is important? Semantic annotation can be a way around this. By automatically annotating the boxes, so that the machine can "understand" the meaning and value of them, enables data to be flagged as "important" and worth highlighting. Future work aims to examine this issue further so that each year of accumulated memories can be summarised with the key memory boxes of that time. Once the system "knows" what is deemed important, automating this process is a separate challenge. Reducing the burden on the user is a key requirement.

8.3.4 Interpreting Emotional Data

Development of the emotional classification algorithms is another exciting direction for this work. As stated previously, recent research into the area of affective computing has demonstrated that emotional states of people can be recognized from their physiological signals (Ivonin et al. 2012; van den Broek et al. 2009; McDuff et al. 2012). Currently, the system utilizes this type of information in the form of heartbeat and acceleration signals, and links this data with photographic and location information to infer emotions from events. Whilst this is a good starting point, further examination into this point is required so that the system knows the features of being happy, sad, angry, etc. so that data can be classified and clustered in this way. Questions such as, "When have I been happiest" or "Does this location make me sad?" can begin to be answered.

8.3.5 Querying Photographic Data

One limitation of the system is that the questions approach only considers data that is straightforward to measure (location, accelerometer and heartbeat). Machine learning algorithms can easily classify or cluster this type of data. Future work would consider expanding the range of questions so that photographic data could be queried, instead of being linked in at a later time. Executing such queries requires sophisticated interpretation, such as "*Find a picture of me playing with Peter when he was a toddler*" (Fitzgibbon & Reiter 2005). This type of query places considerable focus on computer vision and image understanding (Delaitre et al. 2010; Bicocchi et al. 2012; Nakajima et al. 2000). In order to execute this query, an innate understanding of who the people in the picture are and activity recognition are required.

8.4 Concluding Remarks

The accumulation of personal digital memories is inevitable (Fitzgibbon & Reiter 2005). As technology advances, the landscape of our world is rapidly changing. Devices, prevalent within the environment, are now able to capture a multitude of data, both physiological and ecological. As stated previously, a consequence of living in the digital age is the abundance of information that is available. In today's society, it is common practice to capture, store, upload and share almost every moment of daily life. However, as we accumulate more data, the risk of "information overload" increases. Harnessing and creating human digital memories, composed of this information, is useful for not only reminiscing but also has the potential to influence almost every aspect of people's lives. Furthermore, having a system that understands our data is extremely exciting and opens up a whole new way to interact with these devices. In recognition of these points, this thesis presents the DigMem system, which creates rich and interactive human digital memories (memory boxes) using a variety of distributed mobile services, linked data and machine learning to create such memories, whilst enabling life questions to be answered simultaneously. This has been achieved with the development of the three components of the system – MoDM, the DigMem Server and the DigMem Web Application, which each provide their own novelties.

The MoDM (Dobbins, Merabti, et al. 2012b) middleware platform has been developed as a method of obtaining the use of device-specific services for the purpose of collecting human digital memory data. This development is a promising step towards using everyday devices as sources of information, which are capable of providing information for a memory box. Any MoDM compliant device is able to be used, thus eliminating the need for specialist, and expensive, devices that provide only limited data. The development of this component is the first step in collecting and sharing data, via ubiquitous devices, in order to create human digital memories. By retrieving data from pervasive devices, memories are richer in their context. Information is incorporated into the memory that necessarily would not have been available on the current device. For example, the user's mobile device is not able to take a temperature reading of the room, however, by connecting to the thermostat this information could be retrieved and sent back to enhance the memory. A wider range of information is able to be incorporated into the memory box.

The DigMem Server (Dobbins, Merabti, Fergus, Llewellyn-Jones, et al. 2013) aspect of the system has been developed as a way to store collected raw human digital memory data and transform this information into a set of features. This implementation provides a unique method of storing accumulated human digital memory data items and provides a method of creating a universal human digital memory feature set. Items from distributed sources are brought together and, combined to form a single search space. This allows us to treat the searching of these documents as a clustering and classification problem. The feature set contains every data item that has been collected. As time passes this element grows with the user. When memory boxes are created this human digital memory space is searched and clusters of data are extracted or behaviour classified.

The DigMem web application (Dobbins, Merabti, Fergus & Llewellyn-Jones 2013a) aspect of the system provides a web interface for the user so that they can register and log in to search their human digital memory data. This interface provides a bridge between the DigMem Server and the user. This aspect of the system enables the user to interact with their memories. Information that has been found, within the previously described human digital memory space, are organised within memory boxes of temporal human-life experiences. Interaction with such memories is essential and is unique to this research. Users are able to delve into their data to see various pieces of information. By not overloading the user with all of their information simultaneously prevents information overload and enables them to choose the items of information that they wish to view. The system has the ability to enable users to reason over their behaviour and to retrieve any time of their lives. Our actions and how our bodies have changed over time are vividly portrayed. This aspect of the system also enables life questions about the user to be answered. This is extremely exciting as aspects about ourselves, which we might not know, can begin to be uncovered.

The evaluation results, in chapter seven, support the idea of using machine learning algorithms, and the universal set of human digital memory feature data, as a viable method of storing and searching human digital memory data. Furthermore, the case studies illustrate the system's capabilities in a practical environment. Overall, this project provides a flexible and extensible solution that is capable of accumulating and creating human digital memories over a lifetime, whilst also making a significant contribution to the emerging area of human digital memories. Its application in a variety of areas is evident and has the potential to influence almost every aspect of our lives.

References

Abdullah, S., Lane, N.D. & Choudhury, T., 2012. Towards Population Scale Activity Recognition: A Framework for Handling Data Diversity. In *Twenty-Sixth AAAI Conference on Artificial Intelligence*. pp. 851–857.

Adamopoulos, P. & Tuzhilin, A., 2013. Recommendation Opportunities: Improving Item Prediction UsingWeighted Percentile Methods in Collaborative Filtering Systems. In *Proceedings of the 7th ACM conference on Recommender systems - RecSys '13*. New York, New York, USA: ACM Press, pp. 351–354.

Adhvaryn, P.S. & Panchal, M., 2012. A Review on Diverse Ensemble Methods for Classification. *IOSR Journal of Computer Engineering*, 1(4), pp.27–32.

Agency for Healthcare Research and Quality (AHRQ), 2012. Evidence-based Practice Center Systematic Review Protocol Pressure Ulcer Risk Assessment and Prevention: A Comparative Effectiveness Review. , pp.1–18. Available at: http://www.effectivehealthcare.ahrq.gov/ehc/products/309/926/Pressure-Ulcer-Prevention_Protocol_20120110.pdf.

Agrawal, R. & Srikant, R., 1994. Fast Algorithms for Mining Association Rules. In *Proceedings of the 20th International Conference on Very Large Data Bases (VLDB)*. pp. 487–499.

Ahmed, N., Natarajan, T. & Rao, K.R., 1974. Discrete Cosine Transform. *IEEE Transactions on Computers*, C-23(1), pp.90–93.

Almeida, G.J.M. et al., 2011. Physical activity measured by the SenseWear Armband in women with rheumatoid arthritis. *Physical therapy*, 91(9), pp.1367–1376.

Alzheimer's Research UK, 2012. Dementia Statistics. , pp.1–8. Available at: http://www.alzheimersresearchuk.org/dementia-statistics/ [Accessed April 19, 2013].

Andre, D. et al., 2006. The Development of the SenseWear ® armband , a Revolutionary Energy Assessment Device to Assess Physical Activity and Lifestyle. , pp.1–19. Available at: http://www.dotfit.com/sites/63/templates/categories/images/1783/Dev_SenseWear_article.pdf.

Araujo, S. & Houben, G.-J., 2010. Linking Personal Data: Towards a Web of Digital Memories. In *Proceedings of the WebSci10: Extending the Frontiers of Society On-Line*.

Atzori, L., Iera, A. & Morabito, G., 2010. The Internet of Things: A survey. *Computer Networks*, 54(15), pp.2787–2805.

Au, W.-H., Chan, K.C.C. & Yao, X., 2003. A Novel Evolutionary Data Mining Algorithm With Applications to Churn Prediction. *IEEE Transactions on Evolutionary Computation*, 7(6), pp.532–545.

Aud, M.A. et al., 2010. Smart Carpet: Developing a Sensor System to Detect Falls and Summon Assistance. *Journal of Gerontological Nursing*, 36(7), pp.8–12.

Auer, S. et al., 2007. DBpedia: A Nucleus for a Web of Open Data. In *Proceeding ISWC'07/ASWC'07 Proceedings of the 6th international The semantic web and 2nd Asian conference on Asian semantic web conference*. pp. 722–735.

Bakar, Z.A. et al., 2012. Classification of Parkinson's disease Based on Multilayer Perceptrons (MLPs) Neural Network and ANOVA as a Feature Extraction. In *8th IEEE International Colloquium on Signal Processing and its Applications (CSPA)*. IEEE, pp. 63–67.

Banks, R., Kirk, D. & Sellen, A., 2012. A Design Perspective on Three Technology Heirlooms. *Human–Computer Interaction*, 27(1-2), pp.63–91.

Bao, L. & Intille, S.S., 2004. Activity Recognition from User-Annotated Acceleration Data. *Pervasive Computing*, 3001, pp.1–17.

Becker, C. & Bizer, C., 2008. DBpedia Mobile : A Location-Enabled Linked Data Browser. In *Linked Data on the Web (LDOW2008)*. p. 369.

Belimpasakis, P., Roimela, K. & You, Y., 2009. Experience Explorer: A Life-Logging Platform Based on Mobile Context Collection. In *2009 Third International Conference on Next Generation Mobile Applications, Services and Technologies*. IEEE, pp. 77–82.

Bennett, G., Dealey, C. & Posnett, J., 2004. The cost of pressure ulcers in the UK. *Age and Ageing*, 33(3), pp.230–235.

Berners-Lee, T., 1996. WWW: Past, Present, and Future. *Computer*, 29(10), pp.69–77.

Berners-Lee, T., 2009. Linked Data. Available at: http://www.w3.org/DesignIssues/LinkedData.html [Accessed August 15, 2012].

Berners-Lee, T., Hendler, J. & Lassila, O., 2001. The Semantic Web. *Scientific American*, 284(5), pp.28–37.

Berry, E. et al., 2007. The use of a wearable camera, SenseCam, as a pictorial diary to improve autobiographical memory in a patient with limbic encephalitis: A preliminary report. *Neuropsychological Rehabilitation*, 17(4-5), pp.582–601.

Bicocchi, N., Lasagni, M. & Zambonelli, F., 2012. Bridging Vision and commonsense for Multimodal Situation Recognition in Pervasive Systems. In *IEEE International Conference on Pervasive Computing and Communications (PerCom)*. IEEE, pp. 48–56.

Biddle, S.J.H. et al., 2010. Tracking of Sedentary Behaviours of Young People: A Systematic Review. *Preventive Medicine*, 51(5), pp.345–51.

Bilton, N., 2012. Behind the Google Goggles, Virtual Reality. *The New York Times*.

Bins, J. & Draper, B.A., 2001. Feature Selection from Huge Feature Sets. In *Proceedings of the Eighth IEEE International Conference on Computer Vision (ICCV '01)*. IEEE Comput. Soc, pp. 159–165.

Birant, D. & Kut, A., 2007. ST-DBSCAN: An algorithm for clustering spatial–temporal data. *Data & Knowledge Engineering*, 60(1), pp.208–221.

Bisignano, M., Modica, G. Di & Tomarchio, O., 2005. JMobiPeer : a middleware for mobile peer-to-peer computing in MANETs. In *First International Workshop on Mobility in Peer-to-Peer Systems (MPPS) (ICDCSW'05)*.

Bizer, C., 2009. The Emerging Web of Linked Data. *IEEE Intelligent Systems*, 24(5), pp.87–92.

Bizer, C., Heath, T. & Berners-Lee, T., 2009. Linked Data - The Story So Far. *International Journal on Semantic Web and Information Systems*, 5(3), pp.1–22.

Bizer, C., Lehmann, J., et al., 2009. DBpedia - A crystallization point for the Web of Data. *Web Semantics: Science, Services and Agents on the World Wide Web*, 7(3), pp.154–165.

Blum, M., Pentland, A. & Troster, G., 2006. InSense: Interest-Based Life Logging. *IEEE Multimedia*, 13(4), pp.40–48.

Bonato, P., 2010. Wearable Sensors and Systems. From Enabling Technology to Clinical Applications. *IEEE Engineering in Medicine and Biology Society Magazine*, 29(3), pp.25–36.

Bonomi, A.G. et al., 2009. Detection of Type, Duration, and Intensity of Physical Activity Using an Accelerometer. *Medicine and science in sports and exercise*, 41(9), pp.1770–7.

Bowen, S. & Petrelli, D., 2011. Remembering Today Tomorrow: Exploring the Human-Centred Design of Digital Mementos. *International Journal of Human-Computer Studies*, 69(5), pp.324–337.

Brindley, R., Bateman, A. & Gracey, F., 2011. Exploration of use of SenseCam to support autobiographical memory retrieval within a cognitive-behavioural therapeutic intervention following acquired brain injury. *Memory*, 19(7), pp.745–757.

Browne, G. et al., 2011. SenseCam improves memory for recent events and quality of life in a patient with memory retrieval difficulties. *Memory*, 19(7), pp.713–722.

Burns, A. et al., 2010. SHIMMER^{TM} – A Wireless Sensor Platform for Noninvasive Biomedical Research. *IEEE Sensors Journal*, 10(9), pp.1527–1534.

Bush, V., 1945. As We May Think. *The Atlantic Monthly*, (JULY 1945).

Byrne, D. & Jones, G.J.F., 2008. Towards Computational Autobiographical Narratives through Human Digital Memories. *Proceeding of the 2nd ACM international workshop on Story representation, mechanism and context - SRMC '08*, p.9.

Byrne, D. & Jones, G.J.F., 2009. Exploring Narrative Presentation for Large Multimodal Lifelog Collections through Card Sorting. In I. A. Iurgel, N. Zagalo, & P. Petta, eds. *Proceedings of Interactive Storytelling - Second Joint International Conference on Interactive Digital Storytelling, ICIDS 2009, Guimarães, Portugal, December 9-11, 2009.* Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 92–97.

Cavalheri, V. et al., 2011. Energy expenditure during daily activities as measured by two motion sensors in patients with COPD. *Respiratory Medicine*, 105(6), pp.922–929.

Chang, M. et al., 2009. Tracing for Web 3.0. In *Proceedings of the 2009 ACM SIGPLAN/SIGOPS international conference on Virtual execution environments - VEE '09*. New York, New York, USA: ACM Press, pp. 71–80.

Chawla, N. V. et al., 2002. SMOTE: Synthetic Minority Over-sampling Technique. *Journal of Artificial Intelligence Research*, 16(1), pp.321–357.

Chawla, N. V. et al., 2003. SMOTEBoost: Improving Prediction of the Minority Class in Boosting. *In Proceedings of the Principles of Knowledge Discovery in Databases, PKDD-2003*, pp.107–119.

Chen, Y. & J.F. Jones, G., 2008. Integrating Memory Context into Personal Information Re-finding. In *The 2nd BCS IRSG Symposium: Future Directions in Information Access*. pp. 14–21.

Chen, Y. & Jones, G.J.F., 2009. An Event-Based Interface to Support Personal Lifelog Search. In *HCI International 2009, July 2009, San Diego, U.S.A.*

Chen, Y. & Jones, G.J.F., 2010. Augmenting human memory using personal lifelogs. In *Proceedings of the 1st Augmented Human International Conference*. New York, New York, USA: ACM Press.

Chennuru, S. et al., 2010. Mobile Lifelogger-recording, indexing, and understanding a mobile user's life. In *Proceedings of MobiCase*.

Cheung, K.-H. et al., 2005. YeastHub: a semantic web use case for integrating data in the life sciences domain. *Bioinformatics*, 21 Suppl 1, pp.i85–96.

Cisco, 2012. Cisco Visual Networking Index Forecast Projects 18-Fold Growth in Global Mobile Internet Data Traffic From 2011 to 2016. Available at: http://newsroom.cisco.com/press-release-content?type=webcontent&articleId=668380 [Accessed August 1, 2012].

Cisco, 2013. Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2012–2017. , pp.1–34.

Clark, B.K. et al., 2009. Validity and Reliability of Measures of Television Viewing Time and Other Non-Occupational Sedentary Behaviour of Adults: A Review. *Obesity Reviews : An Official Journal of the International Association for the Study of Obesity*, 10(1), pp.7–16.

Coyle, S. et al., 2010. BIOTEX—Biosensing Textiles for Personalised Healthcare Management. *IEEE Transactions on Information Technology in Biomedicine*, 14(2), pp.364–370.

Crete-Nishihata, M. et al., 2012. Reconstructing the Past: Personal Memory Technologies Are Not Just Personal and Not Just for Memory. *Human–Computer Interaction*, 27(1-2), pp.92–123.

Curone, D. et al., 2010. Smart Garments for Emergency Operators: The ProeTEX Project. *IEEE Transactions on Information Technology in Biomedicine*, 14(3), pp.694–701.

Cyganiak, R. & Jentzsch, A., 2011. Linking Open Data cloud diagram. Available at: http://lod-cloud.net [Accessed July 23, 2013].

Dadgostar, F., Bigdeli, A. & Smith, T., 2011. Demo: An Automated Face Enrolment and Recognition System Across Multiple Cameras On CCTV Networks. In *2011 Fifth ACM/IEEE International Conference on Distributed Smart Cameras (ICDSC)*. IEEE, pp. 1–2.

Davies, R.J. et al., 2009. A user driven approach to develop a cognitive prosthetic to address the unmet needs of people with mild dementia. *Pervasive and Mobile Computing*, 5(3), pp.253–267.

DCU, 2011. Centre for Digital Video Processing Dublin City University, Ireland. Available at: http://www.cdvp.dcu.ie/ [Accessed November 15, 2012].

De Jong, K.A., Spears, W.M. & Gordon, D.F., 1993. Using Genetic Algorithms for Concept Learning. *Machine Learning*, 13(2-3), pp.161–188.

De Leo, G., Brivio, E. & Sautter, S.W., 2011. Supporting Autobiographical Memory in Patients with Alzheimer's Disease Using Smart Phones. *Applied Neuropsychology*, 18(1), pp.69–76.

Delaitre, V., Laptev, I. & Sivic, J., 2010. Recognizing human actions in still images: a study of bag-of-features and part-based representations. *Procedings of the British Machine Vision Conference (BMVC)*, 2(5), p.7.

Dias, A. et al., 2009. Measuring Physical Activity with Sensors : A Qualitative Study. *Medical Informatics*, pp.475–479.

Dickie, C. et al., 2004. Augmenting and Sharing Memory with eyeBlog. In *Proceedings of the the 1st ACM workshop on Continuous archival and retrieval of personal experiences - CARPE'04*. New York, New York, USA, NY, USA: ACM Press, p. 105.

Dikaiakos, M.D. et al., 2009. Cloud Computing: Distributed Internet Computing for IT and Scientific Research. *IEEE Internet Computing*, 13(5), pp.10–13.

Dobbins, C., Fergus, P., Merabti, M., & Llewellyn-Jones, D., 2012a. Monitoring and Measuring Sedentary Behaviour with the Aid of Human Digital Memories. In *Proceedings of The 9th Annual IEEE Consumer Communications & Networking Conference (CCNC'12)*. Las Vegas, Nevada, USA: IEEE, pp. 395–398.

Dobbins, C., Fergus, P., Merabti, M., & Llewellyn-Jones, D., 2012b. Remotely Monitoring and Preventing the Development of Pressure Ulcers with the Aid of Human Digital Memories. In *Proceedings of 47th Annual IEEE International Conference on Communications (ICC'12)*. Ottawa, Canada: IEEE, pp. 6101 – 6105.

Dobbins, C., Fergus, P., Stratton, G., Rosenberg, M., & Merabti, M., 2013. Monitoring and reducing sedentary behavior in the elderly with the aid of human digital memories. *Telemedicine and e-Health*, 19(3), pp.173–185.

Dobbins, C., Merabti, M., Fergus, P., & Llewellyn-Jones, D., 2011. Towards a Framework for Capturing and Distributing Rich Interactive Human Digital Memories. In *Proceedings of 12th Annual PostGraduate Symposium on the Convergence of Telecommunications, Networking and Broadcasting (PGNet'11)*. Liverpool, UK, pp. 1–8.

Dobbins, C., Merabti, M., Fergus, P., & Llewellyn-Jones, D., 2012a. Augmenting Human Digital Memories with Physiological Data. In *Proceedings of 3rd IEEE International Conference on Network Embedded Systems for Every Application (NESEA'12)*. Liverpool, UK: IEEE, pp. 35–41.

Dobbins, C., Merabti, M., Fergus, P., & Llewellyn-Jones, D., 2012b. Capturing and Sharing Human Digital Memories with the Aid of Ubiquitous Peer– to–Peer Mobile Services. In *Proceedings of 10th Annual IEEE International Conference on Pervasive Computing and Communications (PerCom'12)*. Lugano, Switzerland: IEEE, pp. 64–69.

Dobbins, C., Merabti, M., Fergus, P., & Llewellyn-Jones, D., 2013a. Creating Human Digital Memories for a Richer Recall of Life Experiences. In *Proceedings of 10th IEEE International Conference on Networking, Sensing and Control (ICNSC'13)*. Evry, France: IEEE, pp. 246–251.

Dobbins, C., Merabti, M., Fergus, P., & Llewellyn-Jones, D., 2013b. Creating Human Digital Memories with the Aid of Pervasive Mobile Devices. *Pervasive and Mobile Computing (In Press)*.

Dobbins, C., Merabti, M., Fergus, P., Llewellyn-Jones, D., & Bouhafs, F., 2013. Exploiting linked data to create rich human digital memories. *Computer Communications*, 36(15-16), pp.1639–1656.

Dodge, M. & Kitchin, R., 2007. "Outlines of a world coming into existence": pervasive computing and the ethics of forgetting. *Environment and Planning B: Planning and Design*, 34(3), pp.431–445.

Doherty, A.R., 2005. Human Digital Memories. *Health Promotion*, (July). Available at: http://www.computing.dcu.ie/~cgurrin/ca168/seminar7.pdf [Accessed August 1, 2013].

Doherty, A. R., Caprani, N., Conaire, C. Ó., Kalnikaite, V., Gurrin, C., Smeaton, A. F., & O'Connor, N. E., 2011. Passively Recognising Human Activities Through Lifelogging. *Computers in Human Behavior*, 27(5), pp.1948–1958.

Doherty, A. R., Moulin, C. J. A., & Smeaton, A. F., 2011. Automatically assisting human memory: A SenseCam browser. *Memory*, 19(7), pp.785–95.

Doherty, A. R., Pauly-Takacs, K., Caprani, N., Gurrin, C., Moulin, C. J. A., O'Connor, N. E., & Smeaton, A. F., 2012. Experiences of Aiding Autobiographical Memory Using the SenseCam. *Human–Computer Interaction*, 27(1-2), pp.151–174.

Doherty, A. R., Pauly-Takacs, K., Gurrin, C., Moulin, C. J. A., & Smeaton, A. F., 2009. Three Years of SenseCam Images: Observations on Cued Recall. In *Invited speech at SenseCam 2009, Symposium at the 39th Annual Meeting of the Society for Neuroscience: Neu- roscience 2009, Chicago, USA, 16-17 October.*

Doherty, A.R. & Smeaton, A.F., 2010. Automatically Augmenting Lifelog Events Using Pervasively Generated Content from Millions of People. *Sensors*, 10(3), pp.1423–1446.

Dopazo, J. & Carazo, J.M., 1997. Phylogenetic Reconstruction Using an Unsupervised Growing Neural Network That Adopts the Topology of a Phylogenetic Tree. *Journal of Molecular Evolution*, 44(2), pp.226–233.

Duin, R.P.W. et al., 2007. PRTools4: A Matlab Toolbox for Pattern Recognition. *PRTools …*, (August), pp.1–61.

Dwyer, T.J. et al., 2009. Evaluation of the SenseWear activity monitor during exercise in cystic fibrosis and in health. *Respiratory Medicine*, 103(10), pp.1511–1517.

European Network and Information Security Agency (ENISA), 2011. *TO L0G OR NOT TO L0G ? Risks and benefits of emerging life-logging applications*,

Facebook.com, 2012. One Billion Fact Sheet. Available at: http://newsroom.fb.com/download-media/4227 [Accessed November 10, 2012].

Fawcett, T., 2006. An introduction to ROC analysis. *Pattern Recognition Letters*, 27(8), pp.861–874.

Fensel, D. et al., 2011. *Semantic Web: Why, What, and How?*, Berlin, Heidelberg: Springer Berlin Heidelberg.

Fergus, P., Attwood, A., Dobbins, C., Stratton, G., Hussain, A., Al-Jumeily, D., & Randles, M., 2012. Monitoring and measuring physical activity and sedentary behaviour. *International Journal of Healthcare Technology and Management*, 13(5/6), pp.283–303.

Fergus, P., Iram, S., Al-Jumeily, D., Randles, M., & Attwood, A., 2012. Home-Based Health Monitoring and Measurement for Personalised Healthcare. *Journal of Medical Imaging and Health Informatics*, 2(1), pp.35–43.

Figo, D. et al., 2010. Preprocessing Techniques for Context Recognition from Accelerometer Data. *Personal and Ubiquitous Computing*, 14(7), pp.645–662.

Figueiredo, C.P. et al., 2010. Low Power Wireless Acquisition Module for Wearable Health Monitoring Systems. In *32nd Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. pp. 704–707.

Fischer, B. & Buhmann, J.M., 2003. Bagging for path-based clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(11), pp.1411–1415.

Fitzgibbon, A. & Reiter, E., 2005. Grand Challenges in Computing Research: GC3 Memories for life: managing information over a human lifetime. In *Conference on Grand Challenges in Computing Research*. pp. 13–16.

Forero, M.G., Sroubek, F. & Cristóbal, G., 2004. Identification of tuberculosis bacteria based on shape and color. *Real-Time Imaging*, 10(4), pp.251–262.

Frohlich, D. & Murphy, R., 2000. The Memory Box. *Personal Technologies*, 4(4), pp.238–240.

Fukui, R., Mori, T. & Sato, T., 2012. An Electrostatic Capacitive Floor Sensor System for Human Position Monitoring in a Living Space. *Advanced Robotics*, 26(10), pp.1127–1142.

Fuller, M., Kelly, L. & Jones, G.J.F., 2008. Applying contextual memory cues for retrieval from personal information archives. In *Personal Information Management (PIM), in conjunction with CHI 2008 Workshop*.

Gemmell, J. & Bell, G., 2009. *Total Recall: How the E-Memory Revolution Will Change Everything*, Dutton, New York.

Gemmell, J., Bell, G., Lueder, R., Drucker, S., & Wong, C., 2002. MyLifeBits: Fulfilling the Memex Vision. In *Proceedings of the 10th ACM International Conference on Multimedia - MULTIMEDIA ’02*. New York, New York, USA: ACM Press, pp. 235–238.

Gemmell, J., Williams, L., Wood, K., Lueder, R., & Bell, G., 2004. Passive Capture and Ensuing Issues for a Personal Lifetime Store. In *Proceedings of the the 1st ACM workshop on Continuous archival and retrieval of personal experiences - CARPE’04*. New York, New York, USA: ACM Press, pp. 48–55.

Gilleade, K. & Fairclough, S.H., 2010. Physiology as XP – Body Blogging to Victory. In *BioS-Play Workshop at Fun and Games, Leuven, Belgium*.

Goldberg, D.E., 1989. *Genetic Algorithms in Search, Optimization and Machine Learning*, Reading, MA, USA: Addison-Wesley.

Gong, L., 2001. JXTA: A Network Programming Environment. *IEEE Internet Computing*, 5(3), pp.88–95.

Google, 2012. Google Glasses - Project Glass. Available at: https://plus.google.com/+projectglass/about [Accessed October 10, 2012].

Gozali, J.P., Kan, M.-Y. & Sundaram, H., 2012. How Do People Organize Their Photos in Each Event and How Does It Affect Storytelling, Searching and Interpretation Tasks? In *Proceedings of the 12th ACM/IEEE-CS joint conference on Digital Libraries - JCDL ’12*. New York, New York, USA: ACM Press, pp. 315–324.

Graham, A. et al., 2002. Time as Essence for Photo Browsing Through Personal Digital Libraries. In *Proceedings of the second ACM/IEEE-CS joint conference on Digital libraries - JCDL ’02*. New York, New York, USA, NY, USA: ACM Press, p. 326.

Grossi, F. et al., 2008. An Assistive Home Automation and Monitoring System. In *International Conference on Consumer Electronics (ICCE) 2008 - Digest of Technical Papers*. IEEE, pp. 1–2.

Guo, B., Zhang, D. & Wang, Z., 2011. Living with Internet of Things: The Emergence of Embedded Intelligence. In *2011 International Conference on Internet of Things and 4th International Conference on Cyber, Physical and Social Computing*. IEEE, pp. 297–304.

Gurrin, C., 2012. Living with SenseCam: Experiences, Motivations and Advances. In *Invited talk at SenseCam 2012, 3-4 April 2012, Oxford, UK*.

Gurrin, C., Byrne, D., O’Connor, N., Jones, G. J. F., & Smeaton, A. F., 2008. Architecture and Challenges of Maintaining a Large-scale, Context-aware Human Digital Memory. In *5th International Conference on Visual Information Engineering*. pp. 158–163.

Haartsen, J., 1998. Bluetooth - The universal radio interface for ad hoc, wireless connectivity. *Ericsson Review*, 1(3), pp.110–117.

Hadi, A. S., Fergus, P., Dobbins, C., & Al-Bakry, A. M., 2013. A Machine Learning Algorithm for Searching Vectorised RDF Data. In *The 27th IEEE International Conference on Advanced Information Networking and Applications Workshops (WAINA'13)*. Barcelona, Spain: IEEE, pp. 613–618.

Hallberg, J. & Kikhia, B., 2009. Reminiscence process using life-log entities for persons with mild dementia'. In *Proceedings of the First International Workshop on Reminiscence Systems (RSW-2009)*. pp. 16–21.

Hamm, J. et al., 2013. Automatic Annotation of Daily Activity from Smartphone-based Multisensory Streams. In *Mobile Computing, Applications, and Services*. pp. 328–342.

Han, H., Wang, W.-Y. & Mao, B.-H., 2005. Borderline-SMOTE: A New Over-Sampling Method in Imbalanced Data Sets Learning D.-S. Huang, X.-P. Zhang, & G.-B. Huang, eds. *Advances in Intelligent Computing*, 3644(1), pp.878–887.

Harada, S. et al., 2004. Lost in Memories: Interacting With Photo Collections On PDAs. In *Proceedings of the 2004 joint ACM/IEEE conference on Digital libraries - JCDL '04*. New York, New York, USA, NY, USA: ACM Press, p. 325.

Harle, R. et al., 2011. Towards real-time profiling of sprints using wearable pressure sensors. *Computer Communications*, 35(6), pp.650–660.

Hastie, T., Tibshirani, R. & Friedman, J., 2009. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction* 2nd Editio., Springer Press.

Healey, J. & Picard, R.W., 1998. StartleCam: A Cybernetic Wearable Camera. *Digest of Papers. Second International Symposium on Wearable Computers (Cat. No.98EX215)*, pp.42–49.

Hendler, J., 2009. Web 3.0 Emerging. *Computer*, 42(1), pp.111–113.

Herrero, J., Valencia, A. & Dopazo, J., 2001. A hierarchical unsupervised growing neural network for clustering gene expression patterns. *Bioinformatics*, 17(2), pp.126–136.

Hirt, W., 2003. Ultra-wideband radio technology: overview and future research. *Computer Communications*, 26(1), pp.46–52.

Hodges, S., Berry, E. & Wood, K., 2011. SenseCam: A wearable camera that stimulates and rehabilitates autobiographical memory. *Memory*, 19(7), pp.685–696.

Hodges, S., Williams, L., Berry, E., Izadi, S., Srinivasan, J., Butler, A., … Wood, K., 2006. SenseCam: A Retrospective Memory Aid P. Dourish & A. Friday, eds. *UbiComp 2006: Ubiquitous Computing*, 4206, pp.177–193.

Holmes, E., Nicholson, J.K. & Tranter, G., 2001. Metabonomic Characterization of Genetic Variations in Toxicological and Metabolic Responses Using Probabilistic Neural Networks. *Chemical Research in Toxicology*, 14(2), pp.182–191.

Hotho, A. & Stumme, G., 2010. Towards the ubiquitous Web. *Semantic Web*, 1(1-2), pp.117–119.

Huang, H.-H. et al., 2012. Toward a Memory Assistant Companion for the Individuals with Mild Memory Impairment. In *IEEE 11th International Conference on Cognitive Informatics and Cognitive Computing (ICCI*CC)*. IEEE, pp. 295–299.

International Council of Nurses (ICN) & Pfizer Inc., 2009. A Global Survey of Nurses (May 2009). *Nurses in the Workplace: Expectations and Needs*. Available at: http://www.icn.ch/news/nurses-in-the-workplace-expectations-and-needs/a-global-survey-of-nurses-may-2009-1335.html [Accessed December 2, 2011].

Introna, L.D. & Wood, D., 2004. Picturing Algorithmic Surveillance: The Politics of Facial Recognition Systems. *Surveillance & Society*, 2(2/3), pp.177–198.

Ismail, A. et al., 2009. Identifying Personal Memories and Shared Experience Using P2P Networks. In *The 10th Annual Conference on the Convergence of Telecommunications, Networking & Broadcasting (PGNet 2009), Liverpool, UK, 22-23 June 2009*.

Ivonin, L. et al., 2012. Unconscious emotions: quantifying and logging something we are not aware of. *Personal and Ubiquitous Computing*, 17(4), pp.663–673.

Jain, A.K., 2010. Data Clustering: 50 years beyond K-means. *Pattern Recognition Letters*, 31(8), pp.651–666.

Json.org, Introducing JSON. Available at: http://www.json.org/ [Accessed August 5, 2013].

Kalnikaite, V. et al., 2010. Now Let Me See Where I Was: Understanding How Lifelogs Mediate Memory. In *Proceedings of the 28th International Conference on Human Factors in Computing Systems - CHI '10*. New York, New York, USA: ACM Press, p. 2045.

Kalnikaite, V. & Whittaker, S., 2011. A Saunter Down Memory Lane: Digital Reflection on Personal Mementos. *International Journal of Human-Computer Studies*, 69(5), pp.298–310.

Kalnikaite, V. & Whittaker, S., 2012. Synergetic Recollection: How to Design Lifelogging Tools That Help Locate the Right Information. *Human-Computer Interaction: The Agency Perspective*, 396, pp.329–348.

Kamel Boulos, M.N. & Wheeler, S., 2007. The emerging Web 2.0 social software: an enabling suite of sociable technologies in health and health care education. *Health Information and Libraries Journal*, 24(1), pp.2–23.

Karaman, S. et al., 2011. Activities of Daily Living Indexing by Hierarchical HMM for Dementia Diagnostics. In *9th International Workshop on Content-Based Multimedia Indexing (CBMI)*. IEEE, pp. 79–84.

Kawamura, T. et al., 2007. Ubiquitous Memories: a memory externalization system using physical objects. *Personal and Ubiquitous Computing*, 11(4), pp.287–298.

Kelly, L., 2007. The Information Retrieval Challenge of Human Digital Memories. In *BCS IRSG Symposium: Future Directions in Information*. British Computer Society.

Kelly, L. & Jones, G.J.F., 2007. Venturing into the labyrinth: the information retrieval challenge of human digital memories. In *Workshop on Supporting Human Memory with Interactive Systems, Lancaster, UK*. pp. 37–40.

Kelly, L. & Jones, G.J.F., 2009. Examining the utility of affective response in search of personal lifelogs. In *5th Workshop on Emotion in Human-Computer Interaction, held at the 23rd BCS HCI Group conference 2009, 1 September 2009, Cambridge, UK.*

Kelly, P. et al., 2011. Can We Use Digital Life-Log Images To Investigate Active And Sedentary Travel Behaviour? Results From A Pilot Study. *The International Journal of Behavioral Nutrition and Physical Activity*, 8(1), p.44.

Kelly, P. et al., 2012. Evaluating the feasibility of measuring travel to school using a wearable camera. *American Journal of Preventive Medicine*, 43(5), pp.546–550.

Kikhia, B., Boytsov, A., Hallberg, J., Sani, Z. ul H., Jonsson, H., & Synnes, K., 2011. Structuring and Presenting Lifelogs based on Location Data. *Image*, 4, pp.5–24.

Kikhia, B., Hallberg, J., Bengtsson, J. E., Sävenstedt, S., & Synnes, K., 2010. Building Digital Life Stories For Memory Support. *International Journal of Computers in Healthcare*, 1(2), pp.161–176.

Kikhia, B., Hallberg, J., Synnes, K., & Sani, Z. ul H., 2009. Context-aware Life-Logging for Persons with Mild Dementia. In *Annual International Conference of the IEEE Engineering in Medicine and Biology Society*. pp. 6183–6186.

Kim, M. & Kotz, D., 2011. Identifying Unusual Days. *Journal of Computing Science and Engineering*, 5(1), pp.71–84.

Klack, L. et al., 2010. Future Care Floor: A Sensitive Floor for Movement Monitoring and Fall Detection in Home Environments J. C. Lin & K. S. Nikita, eds. *Wireless Mobile Communication and Healthcare: Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*, 55, pp.211–218.

Kohonen, T., 1990. The self-organizing map. *Proceedings of the IEEE*, 78(9), pp.1464–1480.

Kohonen, T., 2013. Essentials of the self-organizing map. *Neural Networks*, 37, pp.52–65.

Korhonen, I. & Bardram, J.E., 2004. Guest Editorial Introduction to the Special Section on Pervasive Healthcare. *IEEE Transactions on Information Technology in Biomedicine*, 8(3), pp.229–234.

Kortuem, G. et al., 2010. Smart objects as building blocks for the Internet of things. *IEEE Internet Computing*, 14(1), pp.44–51.

Kotsiantis, S.B., 2007. Supervised Machine Learning: A Review of Classification Techniques. *Informatica*, 31(3), pp.249–268.

Krishnan, N., 2001. The Jxta solution to P2P. *JavaWorld.com*. Available at: http://www.javaworld.com/javaworld/jw-10-2001/jw-1019-jxta.html?page=2 [Accessed September 1, 2012].

Krishnan, N.C. & Panchanathan, S., 2008. Analysis of low resolution accelerometer data for continuous human activity recognition. In *2008 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, pp. 3337–3340.

Kumar, V., Furuta, R. & Allen, R.B., 1998. Metadata Visualization for Digital Libraries: Interactive Timeline Editing and Review. In *Proceedings of the third ACM conference on Digital libraries - DL '98*. New York, New York, USA: ACM Press, pp. 126–133.

Lalanne, D. & van den Hoven, E., 2007. Supporting Human Memory with Interactive Systems. In *Workshop at the 2007 British HCI Conference September 4th, Lancaster, UK*.

Langereis, G. et al., 2007. ConText: Contactless sensors for body monitoring incorporated in textiles. In *IEEE International Conference on Portable Information Devices*. pp. 1–5.

Latchoumane, C.-F. V. et al., 2012. Multiway array decomposition analysis of EEGs in Alzheimer's disease. *Journal of Neuroscience Methods*, 207(1), pp.41–50.

Lee, H. et al., 2008. Constructing a SenseCam Visual Diary As A Media Process. *Multimedia Systems*, 14(6), pp.341–349.

Lee, K.M. et al., 2011. Context-Aware Support for Assistive Systems and Services. In *IET Seminar on Assisted Living*. IET Digital Library.

Lee, M.L. & Dey, A.K., 2009. Wearable Experience Capture for Episodic Memory Support. In *12th IEEE International Symposium on Wearable Computers (ISWC)*. pp. 107–108.

Lee, M.W., Khan, A.M. & Kim, T.-S., 2011. A single tri-axial accelerometer-based real-time personal life log system capable of human activity recognition and exercise information generation. *Personal and Ubiquitous Computing*, 15(8), pp.887–898.

Lee, Sang-Woong, Park, J. & Lee, Seong-Whan, 2006. Low resolution face recognition based on support vector data description. *Pattern Recognition*, 39(9), pp.1809–1812.

Lee, Y.-S. & Cho, S.-B., 2012. Recognizing multi-modal sensor signals using evolutionary learning of dynamic Bayesian networks. *Pattern Analysis and Applications*.

Leitner, R., Mairer, H. & Kercek, A., 2003. Real-time classification of polymers with NIR spectral imaging and blob analysis. *Real-Time Imaging*, 9(4), pp.245–251.

Lewis, D.D., 1998. Naive (Bayes) at Forty: The Independence Assumption in Information Retrieval. In C. Nédellec & C. Rouveirol, eds. *Proceedings of the 10th European Conference on Machine Learning (ECML '98)*. Berlin/Heidelberg: Springer-Verlag, pp. 4–15.

Lindley, S.E. et al., 2009. Narrative, Memory and Practice: Tensions and Choices in the Use of a Digital Artefact. In *Proceedings of the 23rd British HCI Group Annual Conference on People and Computers: Celebrating People and Technology*. pp. 1–9.

Lindström, M. et al., 2006. Affective Diary – Designing for Bodily Expressiveness and Self-Reflection. In *CHI '06 extended abstracts on Human factors in computing systems*. New York, New York, USA: ACM Press, pp. 1037–1042.

Liu, C.-L. & Sako, H., 2006. Class-specific feature polynomial classifier for pattern classification and its application to handwritten numeral recognition. *Pattern Recognition*, 39(4), pp.669–681.

Liu, R. et al., 2008. Signal Processing and Accelerometer-based Design for Portable Small Displacement Measurement Device. In *2008 International Conference on Embedded Software and Systems*. IEEE, pp. 575–579.

Long, X., Yin, B. & Aarts, R.M., 2009. Single-Accelerometer-Based Daily Physical Activity Classification. In *Annual International Conference of the IEEE Engineering in Medicine and Biology Society*. pp. 6107–6110.

López, G., Custodio, V. & Moreno, J.I., 2010. LOBIN: E-textile and wireless-sensor-network-based platform for healthcare monitoring in future hospital environments. *IEEE Transactions on Information Technology in Biomedicine*, 14(6), pp.1446–1458.

Lord, S. et al., 2011. Exploring Patterns of Daily Physical and Sedentary Behaviour in Community-Dwelling Older Adults. *Age and Ageing*, 40(2), pp.205–210.

Loveday, C. & Conway, M.A., 2011. Using SenseCam with an amnesic patient: Accessing inaccessible everyday memories. *Memory*, 19(7), pp.697–704.

Luo, S., Jin, J.S.. & Li, J., 2009. A Smart Fridge with an Ability to Enhance Health and Enable Better Nutrition. *International Journal of Multimedia and Ubiquitous Engineering*, 4(2), pp.69–80.

Lymberis, A. & Paradiso, R., 2008. Smart Fabrics and Interactive Textile Enabling Wearable Personal Applications: R&D State of the Art and Future Challenges. In *30th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*. pp. 5270–5273.

Lyons, G.M. et al., 2005. A description of an accelerometer-based mobility monitoring technique. *Medical Engineering & Physics*, 27(6), pp.497–504.

Machajdik, J. et al., 2011. Affective Computing for Wearable Diary and Lifelogging Systems: An Overview. In *35th Annual Workshop of the Austrian Association for Pattern Recognition (OAGM/AAPR)*.

Mainetti, L., Patrono, L. & Vilei, A., 2011. Evolution of wireless sensor networks towards the Internet of Things: A survey. In *19th International Conference on Software, Telecommunications and Computer Networks (SoftCOM)*. pp. 1–6.

Maner, W.L. et al., 2003. Predicting term and preterm delivery with transabdominal uterine electromyography. *Obstetrics and Gynecology*, 101(6), pp.1254–1260.

Mann, S., 1997. Wearable Computing: A First Step Toward Personal Imaging. *Computer*, 30(2), pp.25–32.

Mann, S., 2004. Continuous Lifelong Capture of Personal Experience with EyeTap. In *Proceedings of the the 1st ACM workshop on Continuous archival and retrieval of personal experiences - CARPE'04*. New York, New York, USA: ACM Press, pp. 1–21.

Mann, S. et al., 2005. Designing EyeTap Digital Eyeglasses for Continuous Lifelong Capture and Sharing of Personal Experiences. In *Proceedings of CHI 2005 Conference on Human Factors in Computer Systems*. Portland, OR, USA.

Mannini, A. & Sabatini, A.M., 2010. Machine learning methods for classifying human physical activity from on-body accelerometers. *Sensors*, 10(2), pp.1154–1175.

Manohar, A. & Bhatia, D., 2008. Pressure Detection and Wireless Interface for Patient Bed. In *2008 IEEE Biomedical Circuits and Systems Conference (BioCAS)*. IEEE, pp. 389–392.

MathWorks, 2013. k-Means Clustering. Available at: http://www.mathworks.co.uk/help/stats/k-means-clustering.html [Accessed June 25, 2013].

Matthews, R. et al., 2007. A wearable physiological sensor suite for unobtrusive monitoring of physiological and cognitive state. In *29th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBS)*. pp. 5276–5281.

Mayagoitia, R.E., Nene, A. V. & Veltink, P.H., 2002. Accelerometer and rate gyroscope measurement of kinematics: an inexpensive alternative to optical motion analysis systems. *Journal of Biomechanics*, 35(4), pp.537–542.

McArdle, G., Tahir, A. & Bertolotto, M., 2012. Spatio-temporal clustering of movement data: An application to trajectories generated by human-computer interaction. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 1(2), pp.147–152.

McCarthy, S. et al., 2007. MemoryLane: An Intelligent Mobile Companion for Elderly Users. In *The 7th Information Technology and Telecommunication Conference IT&T 2007*. Dublin, Ireland, pp. 72–82.

McDuff, D. et al., 2012. AffectAura: An Intelligent System for Emotional Memory. In *Proceedings of the 2012 ACM annual conference on Human Factors in Computing Systems - CHI '12*. New York, New York, USA: ACM Press, pp. 849–858.

Microsoft Research, 2011. Reviewing and Sharing SenseCam Images. Available at: http://research.microsoft.com/en-us/um/cambridge/projects/sensecam/review.htm [Accessed June 1, 2013].

Miller, E., 1998. An Introduction to the Resource Description Framework. *Bulletin of the American Society for Information Science and Technology*, 25(1), pp.15–19.

Mokaya, F. et al., 2013. MARS: A Muscle Activity Recognition System Enabling Self-configuring Musculoskeletal Sensor Networks. In *The 12th ACM/IEEE Conference on Information Processing in Sensor Networks (IPSN)*. pp. 191–202.

Morishita, S. & Sese, J., 2000. Traversing Itemset Lattices with Statistical Metric Pruning. In *Proceedings of the nineteenth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems - PODS ’00*. New York, New York, USA: ACM Press, pp. 226–236.

Nakajima, C. et al., 2000. People Recognition in Image Sequences by Supervised Learning. In *Report for Massachusetts Institute of Technology (MIT) Artificial Intelligence Laboratory and Center for Biological and Computational Learning a Department Of Brain and Cognitive Sciences*. pp. 1–12.

NHS, 2012. Pressure Ulcers - Causes. Available at: http://www.nhs.uk/Conditions/Pressure-ulcers/Pages/Causes.aspx [Accessed December 14, 2012].

Niazmand, K. et al., 2010. A New Washable Low-Cost Garment for Everyday Fall Detection. In *Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. pp. 6377–6380.

NPD Group, 2013. More than 400 Million Devices are Connected in U.S. Homes, According to The NPD Group. Available at: https://www.npd.com/wps/portal/npd/us/news/press-releases/more-than-400-million-devices-are-connected-in-us-homes-according-to-the-npd-group/ [Accessed January 8, 2013].

O’Hare, N. et al., 2006. MediAssist: Using Content-Based Analysis and Context to Manage Personal Photo Collections H. Sundaram et al., eds. *Image and Video Retrieval*, 4071, pp.529–532.

O’Hare, N. & Smeaton, A.F., 2009. Context-Aware Person Identification in Personal Photo Collections. *IEEE Transactions on Multimedia*, 11(2), pp.220–228.

Ofcom, 2011. Here are all the latest facts and figures about communications in the UK today. Available at: http://media.ofcom.org.uk/facts/ [Accessed November 1, 2011].

Ofcom, 2012. Facts & Figures. Available at: http://media.ofcom.org.uk/facts/ [Accessed August 17, 2012].

Office for National Statistics, 2010. Healthy Life Expectancy: Living Longer In Poor Health. Available at: http://www.statistics.gov.uk/cci/nugget.asp?id=2159 [Accessed June 1, 2013].

Olsson, T. et al., 2007. User-Centered Design of a Mobile Application for Sharing Life Memories. In *Proceedings of the 4th international conference on mobile technology, applications, and systems and the 1st international symposium on Computer human interaction in mobile technology - Mobility ’07*. New York, New York, USA: ACM Press, pp. 524–531.

Orozco, M. et al., 2006. Dissimilarity-Based Classification of Seismic Signals at Nevado del Ruiz Volcano. *Earth Sciences Research Journal*, 10(2), pp.57–66.

Owen, N. et al., 2011. Adults’ Sedentary Behavior: Determinants and Interventions. *American Journal of Preventive Medicine*, 41(2), pp.189–96.

Pacelli, M. et al., 2006. Sensing Fabrics for Monitoring Physiological and Biomechanical Variables: E-textile solutions. In *3rd IEEE-EMBS International Summer School and Symposium on Medical Devices and Biosensors*. IEEE, pp. 1–4.

Pal, S.K. & Mitra, S., 1992. Multilayer perceptron, fuzzy sets, and classification. *IEEE Transactions on Neural Networks*, 3(5), pp.683–697.

Palaniappan, S. & Awang, R., 2008. Intelligent Heart Disease Prediction System Using Data Mining Techniques. In *IEEE/ACS International Conference on Computer Systems and Applications (AICCSA)*. IEEE, pp. 108–115.

Palazzi, C.E., Bujari, A. & Cervi, E., 2009. P2P File Sharing on Mobile Phones: Design and Implementation of a Prototype. *2009 2nd IEEE International Conference on Computer Science and Information Technology*.

Pandian, P.S. et al., 2008. Smart Vest: Wearable Multi-Parameter Remote Physiological Monitoring System. *Medical engineering & physics*, 30(4), pp.466–77.

Pantelopoulos, A. & Bourbakis, N.G., 2010a. A Survey on Wearable Sensor-Based Systems for Health Monitoring and Prognosis. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 40(1), pp.1–12.

Pantelopoulos, A. & Bourbakis, N.G., 2010b. Prognosis - A Wearable Health-Monitoring System for People at Risk: Methodology and Modeling. *IEEE Transactions on Information Technology in Biomedicine*, 14(3), pp.613–21.

Paradiso, R., Loriga, G. & Taccini, N., 2005. A Wearable Health Care System Based on Knitted Integrated Sensors. *IEEE Transactions on Information Technology in Biomedicine*, 9(3), pp.337–344.

Parameswaran, M., Susarla, A. & Whinston, A.B., 2001. P2P Networking: An Information Sharing Alternative. *Computer*, 34(7), pp.31–38.

Park, H.-S. & Cho, S.-B., 2010. Building Mobile Social Network with Semantic Relation Using Bayesian Network-based Life-log Mining. In *2010 IEEE Second International Conference on Social Computing (SocialCom)*. IEEE, pp. 401–406.

Parzen, E., 1962. On Estimation of a Probability Density Function and Mode. *The Annals of Mathematical Statistics*, 33(3), pp.1065–1076.

Pauly-Takacs, K., Moulin, C.J.A. & Estlin, E.J., 2011. SenseCam as a rehabilitation tool in a child with anterograde amnesia. *Memory*, 19(7), pp.705–712.

Pfisterer, D. et al., 2011. SPITFIRE: Toward a Semantic Web of Things. *IEEE Communications Magazine*, 49(11), pp.40–48.

Phan, T., 2012. Generating Natural-Language Narratives from Activity Recognition with Spurious Classification Pruning. In *Proceedings of the Third International Workshop on Sensing Applications on Mobile Phones - PhoneSense '12*. New York, New York, USA: ACM Press, pp. 1–5.

Piasek, P., Irving, K. & Smeaton, A.F., 2011. SenseCam intervention based on Cognitive Stimulation Therapy framework for early-stage dementia. In *5th International Conference on Pervasive Computing Technologies for Healthcare (PervasiveHealth)*. pp. 522–525.

Picault, J., Ribière, M. & Senot, C., 2010. Beyond life streams : activities and intentions for managing personal digital memories. *Proceedings of the 1st International Workshop on Adaptation, Personalization and Recommendation in the Social-semantic Web (APRESW 2010)*, pp.25–32.

Plaisant, C. et al., 1996. LifeLines: Visualizing Personal Histories. In *Proceedings of the SIGCHI conference on Human factors in computing systems common ground - CHI '96*. New York, New York, USA, NY, USA: ACM Press, p. 221.

Platt, J.C., Czerwinski, M. & Field, B.A., 2003. PhotoTOC: automatic clustering for browsing personal photographs. In *Fourth International Conference on Information, Communications and Signal Processing, 2003 and the Fourth Pacific Rim Conference on Multimedia. Proceedings of the 2003 Joint*. IEEE, pp. 6–10.

Prull, M.W. et al., 2006. Recollection and familiarity in recognition memory: adult age differences and neuropsychological test correlates. *Psychology and Aging*, 21(1), pp.107–18.

Qiu, Z. et al., 2011. Mining User Activity as a Context Source for Search and Retrieval. In *2011 International Conference on Semantic Technology and Information Retrieval (STAIR)*. IEEE, pp. 162–166.

Quinlan, J.R., 1986. Induction of Decision Trees. *Machine Learning*, 1(1), pp.81–106.

Quinlan, J.R., 1996. Improved Use of Continuous Attributes in C4.5. *Journal of Artificial Intelligence Research*, 4, pp.77–90.

Ranpura, A., 2000. How We Remember, and Why We Forget. *BrainConnection.com*. Available at: http://brainconnection.positscience.com/how-we-remember-and-why-we-forget/.

Ravi, N. et al., 2005. Activity Recognition from Accelerometer Data. *Proceedings of the national conference on artificial intelligence*, 20(3), pp.1541–1546.

Reiss, A. & Stricker, D., 2012. Creating and Benchmarking a New Dataset for Physical Activity Monitoring. In *Proceedings of the 5th International Conference on Pervasive Technologies Related to Assistive Environments - PETRA '12*. New York, New York, USA: ACM Press.

Riva, O. & Kangasharju, J., 2008. Challenges and Lessons in Developing Middleware on Smart Phones. *Computer*, 41(10), pp.23–31.

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

Roy Chowdhury, A., Falchuk, B. & Misra, A., 2010. MediAlly: A Provenance-Aware Remote Health Monitoring Middleware. *2010 IEEE International Conference on Pervasive Computing and Communications (PerCom)*, pp.125–134.

Salkind, N.J., 2008. *Statistics for people who (think they) hate statistics* 3rd ed., Sage Publications.

Sandhya, J. et al., 2010. Classification and treatment of different stages of alzheimer's disease using various machine learning methods. *International Journal of Bioinformatics Research*, 2(1), pp.44–52.

Sellen, A. et al., 2007. Do Life-Logging Technologies Support Memory for the Past? An Experimental Study Using SenseCam. In *Conference on Human Factors in Computing Systems, CHI '07, Irvine, CA*. pp. 81–90.

Sellen, A.J. & Whittaker, S., 2010. Beyond Total Capture: A Constructive Critique of Lifelogging. *Communications of the ACM*, 53(5), pp.70–77.

Shorey, R. & Miller, B.A., 2000. The Bluetooth technology: merits and limitations. In *2000 IEEE International Conference on Personal Wireless Communications*. IEEE, pp. 80–84.

Siegemund, F., 2004. A Context-Aware Communication Platform for Smart Objects A. Ferscha & F. Mattern, eds. *Lecture Notes in Computer Science (LNCS)*, 3001, pp.69–86.

Sirlantzis, K., Fairhurst, M.C. & Guest, R.M., 2002. An Evolutionary Algorithm for Classifier and Combination Rule selection in Multiple Classifier Systems. In *Proceedings of the 16th Annual International Conference on Pattern Recognition*. IEEE Comput. Soc, pp. 771–774.

Smith, A., O'Hara, K. & Lewis, P., 2011. Visualising the Past: Annotating a Life with Linked Open Data. In *Web Science Conference 2011*. Koblenz, Germany.

Song, W. et al., 2012. Activity Recognition in Planetary Navigation Field Tests Using Classification Algorithms Applied to Accelerometer Data. In *2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. pp. 1586–1589.

Srivastava, P. & Wong, W.-C., 2012. Hierarchical Human Activity Recognition Using GMM. In *AMBIENT 2012: The Second International Conference on Ambient Computing, Applications, Services and Technologies*. pp. 32–37.

Ståhl, A. et al., 2009. Experiencing the Affective Diary. *Personal and Ubiquitous Computing*, 13(5), pp.365–378.

Stevens, M. M., Abowd, G. D., Truong, K. N., & Vollmer, F., 2003. Getting into the Living Memory Box: Family archives & holistic design. *Personal and Ubiquitous Computing*, 7(3-4), pp.210–216.

Stevens, M., Vollmer, F. & Abowd, G.D., 2002. The Living Memory Box: Function, Form and User Centered Design. In *CHII EA '02 - Extended Abstracts on Human Factors in Computing Systems*. New York, New York, USA: ACM Press, pp. 668–669.

Strayer, S.M. et al., 2011. Development and Evaluation of an Instrument For Assessing Brief Behavioral Change Interventions. *Patient education and counselling*, 83(1), pp.99–105.

Sugano, M. et al., 2006. Indoor Localization System using RSSI Measurement of Wireless Sensor Network based on ZigBee Standard. In *Proceedings of Wireless and Optical Communication MultiConference*.

Sun Microsystems Inc., 2003. Project JXTA v2.0: Java Programmer's Guide. , (May).

Sutton, J., 2010. Memory. *The Stanford Encyclopedia of Philosophy*.

Taraldsen, K. et al., 2011. Evaluation of a Body-Worn Sensor System to Measure Physical Activity in Older People With Impaired Function. *Physical Therapy: Journal of the American Physical Therapy Association*, 91(2), pp.277–285.

Teraoka, T., 2011. Aggregation and Exploration of Heterogeneous Data Collected from Diverse Information Sources. In *Proceedings of 1st international symposium on From digital footprints to social and community intelligence - SCI '11*. New York, New York, USA: ACM Press, pp. 31–36.

Tong, L.-I., Chang, Y.-C. & Lin, S.-H., 2011. Determining the optimal re-sampling strategy for a classification model with imbalanced data using design of experiments and response surface methodologies. *Expert Systems with Applications*, 38(4), pp.4222–4227.

Tranel, D. & Damasio, A.R., 2003. Chapter 2 Neurobiological Foundations of Human Memory. In *The Handbook of Memory Disorders*. John Wiley & Sons, p. 17.

Trelles, O. et al., 2011. Big data, but are we ready? *Nature Reviews Genetics*, 12(3), p.224.

Tsai, F.S. et al., 2009. Design and Development of a Mobile Peer-to-Peer Social Networking Application. *Expert Systems with Applications*, 36(8), pp.11077–11087.

Tulving, E., 1972. Organization of memory: Quo Vadis? , pp.839–847.

Tulving, E., 1984. Precis of elements of episodic memory. *Behavioral and Brain Sciences*, 7(2), pp.223–268.

Tulving, E., 1993. What Is Episodic Memory? *Current Directions in Psychological Science*, 2(3), pp.67 – 70.

Tulving, E., 2002. Episodic Memory: From Mind to Brain. *Annual Review of Psychology*, 53, pp.1–25.

Tummarello, G. et al., 2010. Sig.ma: Live views on the Web of Data. *Web Semantics: Science, Services and Agents on the World Wide Web*, 8(4), pp.355–364.

Tyrer, H.W. et al., 2011. Evolution of an Eldercare Technology System to Monitor Motion and Detect Falls. In *2011 IEEE 13th International Conference on e-Health Networking, Applications and Services (Healthcom)*. IEEE, pp. 157–161.

Van den Broek, E.L., Janssen, J.H. & Westerink, J.H.D.M., 2009. Guidelines for Affective Signal Processing (ASP): from lab to life. In *3rd International Conference on Affective Computing & Intelligent Interaction, ACII 2009, 10-12 Sept 2009, Amsterdam, The Netherlands.* IEEE Computer Society.

Van der Heijden, F. et al., 2004. *Classification, Parameter Estimation and State Estimation: An Engineering Approach using Matlab*, Wiley.

Vogt, J. et al., 2012. Putting Dementia Into Context M. Winckler, P. Forbrig, & R. Bernhaupt, eds. *Lecture Notes in Computer Science*, 7623, pp.181–198.

W3C, 2004a. Resource Description Framework (RDF). Available at: http://www.w3.org/RDF/ [Accessed August 12, 2013].

W3C, 2004b. Resource Description Framework (RDF): Concepts and Abstract Syntax. Available at: http://www.w3.org/TR/2004/REC-rdf-concepts-20040210/ [Accessed July 10, 2012].

W3C, 2008. SPARQL Query Language for RDF. Available at: http://www.w3.org/TR/rdf-sparql-query/ [Accessed August 10, 2013].

W3C, 2011a. Linking Open Data W3C SWEO Community Project. Available at: http://www.w3.org/wiki/SweoIG/TaskForces/CommunityProjects/LinkingOpenData [Accessed August 20, 2012].

W3C, 2011b. W3C Semantic Web Activity. Available at: http://www.w3.org/2001/sw/ [Accessed August 15, 2012].

Wagstaff, K. et al., 2001. Constrained K-means Clustering with Background Knowledge. In *Proceedings of the Eighteenth International Conference on Machine Learning (ICML)*. pp. 577–584.

Wakamiya, S., Lee, R. & Sumiya, K., 2011. Crowd-based Urban Characterization: Extracting Crowd Behavioral Patterns in Urban Areas from Twitter. In *Proceedings of the 3rd ACM SIGSPATIAL International Workshop on Location-Based Social Networks - LBSN '11*. New York, New York, USA: ACM Press, pp. 77–85.

Wang, A.I. & Motzfeldt, P.N., 2007. Peer2Schedule - An Experimental Peer-to-Peer Application to Support Present Collaboration. *2007 International Conference on Collaborative Computing: Networking, Applications and Worksharing (CollaborateCom 2007)*, pp.418–425.

Wang, J.-J. & Wang, S., 2010. Wireless sensor networks for Home Appliance Energy Management based on ZigBee technology. In *2010 International Conference on Machine Learning and Cybernetics*. IEEE, pp. 1041–1046.

Wang, P. & Smeaton, A.F., 2012. Semantics-based selection of everyday concepts in visual lifelogging. *International Journal of Multimedia Information Retrieval*, 1(2), pp.87–101.

Wang, P. & Smeaton, A.F., 2013. Using visual lifelogs to automatically characterize everyday activities. *Information Sciences*, 230, pp.147–161.

Wang, T.-Y. et al., 2011. The development of an intelligent monitoring and caution system for pressure ulcer prevention. In *2011 International Conference on Machine Learning and Cybernetics*. IEEE, pp. 566–571.

Weiser, M., 1999. The Computer for the 21st Century. *ACM SIGMOBILE Mobile Computing and Communications Review*, 3(3), pp.3–11.

West, D., Quigley, A. & Kay, J., 2007. MEMENTO: a digital-physical scrapbook for memory sharing. *Personal and Ubiquitous Computing*, 11(4), pp.313–328.

Whittaker, S., Bergman, O. & Clough, P., 2009. Easy On That Trigger Dad: A Study of Long Term Family Photo Retrieval. *Personal and Ubiquitous Computing*, 14(1), pp.31–43.

Whitwell, J.L. et al., 2012. Neuroimaging correlates of pathologically defined subtypes of Alzheimer's disease: a case-control study. *Lancet Neurology*, 11(10), pp.868–877.

Wilkin, G. a. & Huang, X., 2007. K-Means Clustering Algorithms: Implementation and Comparison. In *Second International Multi-Symposiums on Computer and Computational Sciences (IMSCCS 2007)*. Iowa City, IA, USA: Ieee, pp. 133–136.

Woon, W.L. et al., 2007. Techniques for early detection of Alzheimer's disease using spontaneous EEG recordings. *Physiological Measurement*, 28(4), pp.335–347.

Wu, X., Kumar, V., Ross Quinlan, J., Ghosh, J., Yang, Q., Motoda, H., … Steinberg, D., 2007. Top 10 algorithms in data mining. *Knowledge and Information Systems*, 14(1), pp.1–37.

Yip, M. et al., 2009. A flexible pressure monitoring system for pressure ulcer prevention. In *Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society*. pp. 1212–1215.

Yousefi, R. et al., 2011. A Smart Bed Platform for Monitoring & Ulcer Prevention. In *2011 4th International Conference on Biomedical Engineering and Informatics (BMEI)*. IEEE, pp. 1362–1366.

YouTube, 2012. YouTube Statistics. Available at: http://www.youtube.com/t/press_statistics [Accessed September 10, 2013].

Zhang, Y. et al., 2011. Continuous Functional Activity Monitoring Based on Wearable Tri-axial Accelerometer and Gyroscope. In *Proceedings of the 2011 5th International Conference on Pervasive Computing Technologies for Healthcare (PervasiveHealth)*. IEEE, pp. 370–373.

Appendices

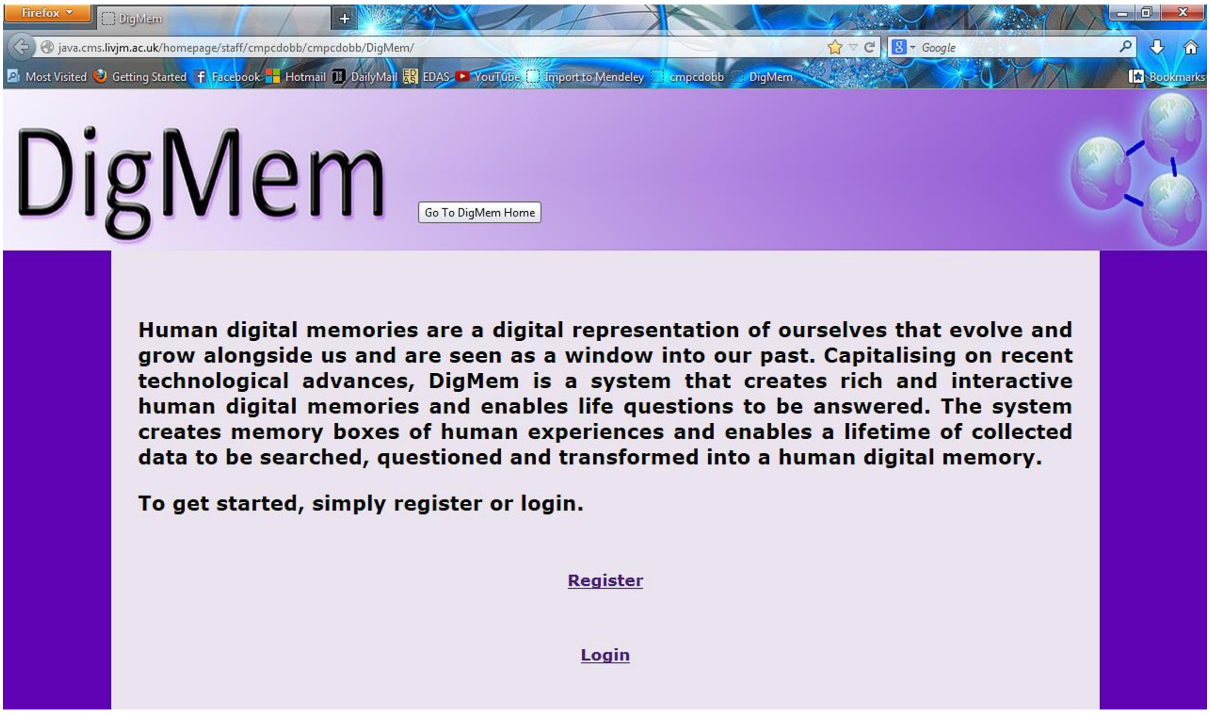
Appendix 1 – Abbreviation List

AD	Alzheimer's disease
AHRQ	Agency for Healthcare Research and Quality
AJAX	Asynchronous JavaScript and XML
AUC	Area Under the Curve
C4.5	Decision Tree Learning
CANFIS	Coactive Neuro-Fuzzy Inference System
CCTV	Closed-circuit television
CF	Cystic Fibrosis
CSP	Common Spatial Patterns Algorithm
CSV	Comma-separated values
DBN	Dynamic Bayesian Network
DBSCAN	Density-Based Spatial Clustering of Applications with Noise
DC	Direct Current
DCT	Discrete Cosine Transform Algorithm
DMEL	Data Mining By Evolutionary Learning Algorithm
ECG	Electrocardiogram
EDA	Electro-dermal Activity sensor
EE	Energy Expenditure
EEG	Electroencephalogram
EMG	Electromyogram
ENISA	European Network and Information Security Agency
EOG	Electrooculogram
FFT	Fast Fourier Transform
GABIL	Genetic Algorithm Based Learning
GPS	Global Positing System
HDM	Human Digital Memory
HDTV	High Definition Television
HTML	Hypertext Markup Language
HTTP	Hypertext Transfer Protocol
ICN	International Council of Nurses
ID	Identification Number

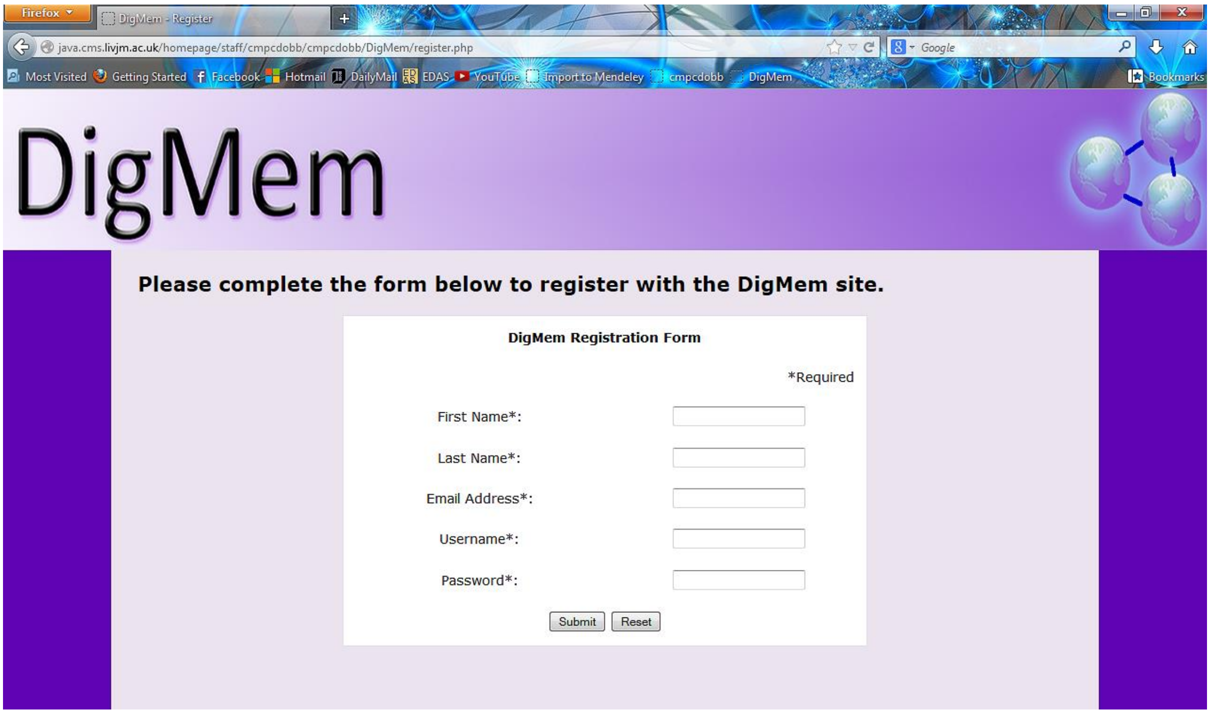
IoT	Internet of Things
J2ME	Java 2 Platform Micro Edition
JSON	JavaScript Object Notation
JXTA	Juxtapose
k-NN	k-nearest neighbour algorithm
KNNC	k-Nearest Neighbour
LDC	Linear Discriminant Classifier
LOGLC	Logistic Classifier
MAD	Multi-way Array Decomposition
MANET	Mobile Ad-Hoc Network
MLP	Multi-Layer Perceptrons
MoDM	Mobile DigMem
MoSoSo	Mobile Social Software
MSN	Mobile Social Networking
NAIVEBC	Naive Bayes classifier
OPTICS	Ordering Points to Identify the Clustering Structure
OVA	One Context Versus-All
P2P	Peer-to-Peer
PARZENC	Parzen Classifier
PCA	Principle Component Analysis
PDA	Personal Digital Assistant
PDF	Probabilistic Density Function
PHP	Hypertext Preprocessor
PLL	Personal Life Log
PML	Perfectly Matched Layer
PNN	Probabilistic Neural Networks
POLYC	Polynomial Classifier
PSD	Power Spectral Density
PSS	Physiological Sensor Suite
QDC	Quadratic Discriminant Classifier
RDF	Resource Description Framework
RFID	Radio-Frequency Identification
RMS	Root Mean Square
ROC	Receiver Operating Characteristics
SA	Silhouette Average
SCS	Simple Classifier System
SD	Sensor Device

SFIT	Smart Fabrics and Interactive Textiles
SMOTE	Synthetic Minority Over-Sampling Technique
SOM	Self-Organizing Map
SOTA	Self-Organising Tree Algorithm
SPARQL	SPARQL Protocol and RDF Query Language
SPI	Serial Parallel Independent
SQL	Structured Query Language
SVC	Support Vector Classifier
SVM	Support Vector Machine
SWA	Sense Wear Armband
TB	Terabyte
TREEC	Decision Tree
UD	User’s Mobile Device
UDC	Uncorrelated Normal Density Based Classifier
UML	Unified Modeling Language
URI	Uniform Resource Identifier
USB	Universal Serial Bus
W3C	World Wide Web Consortium
WHMS	Wearable Health-Monitoring Systems
WWW	World Wide Web
XML	Extensible Mark-up Language

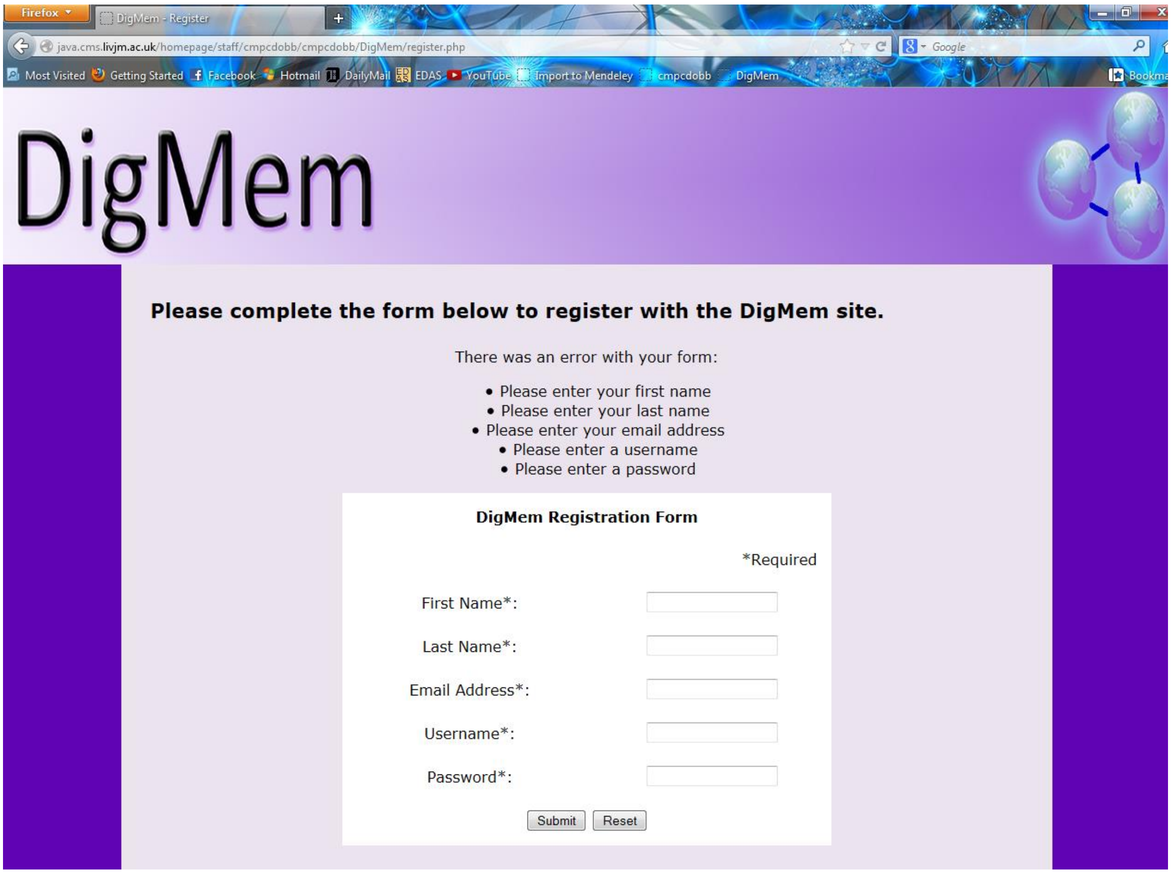
Appendix 2 –DigMem Web Pages



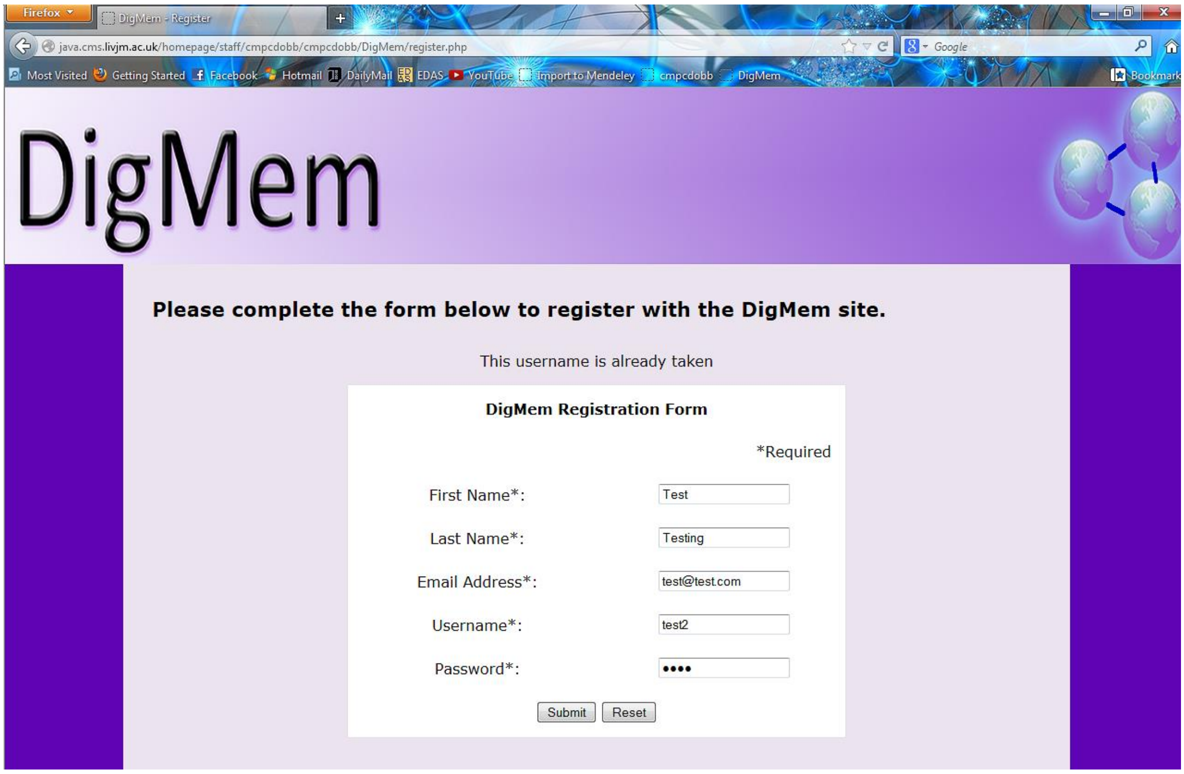
Home Page



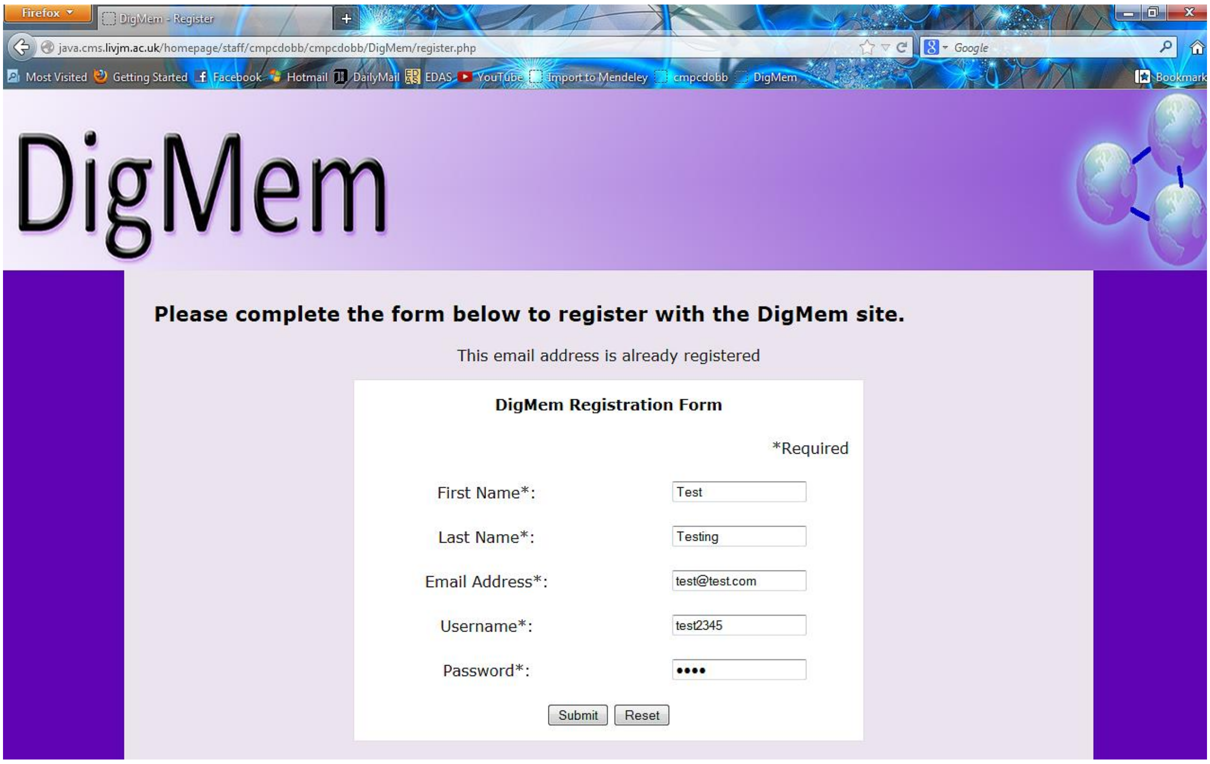
Registration Page



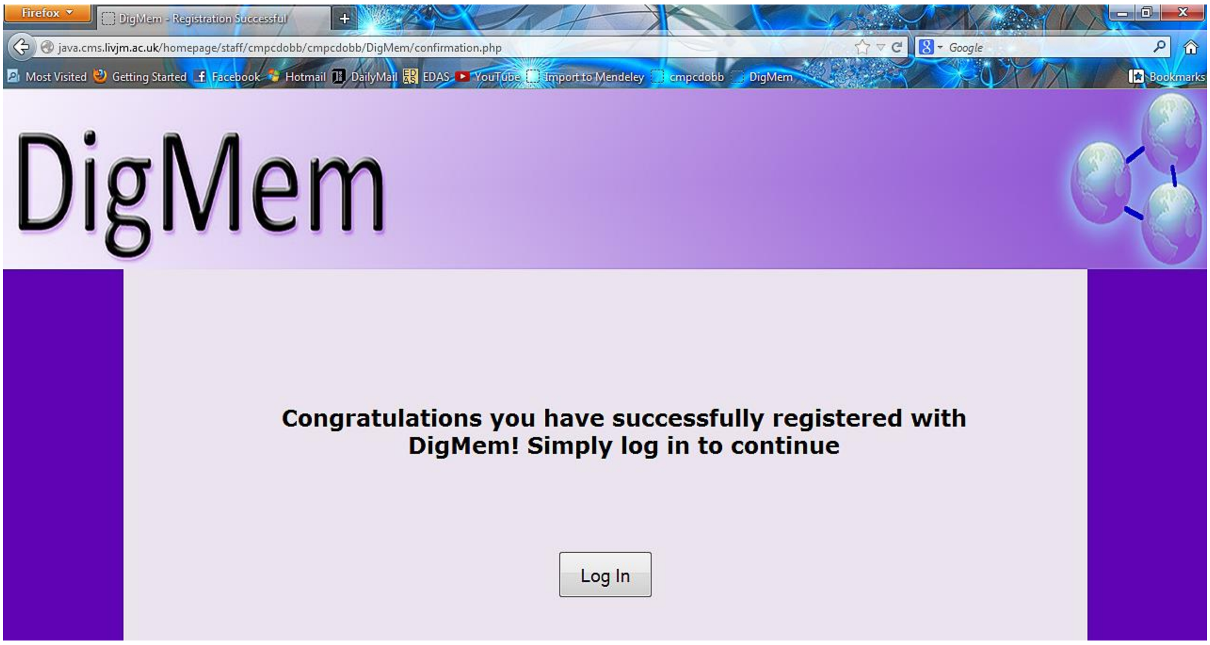
Registration Page Form Validation



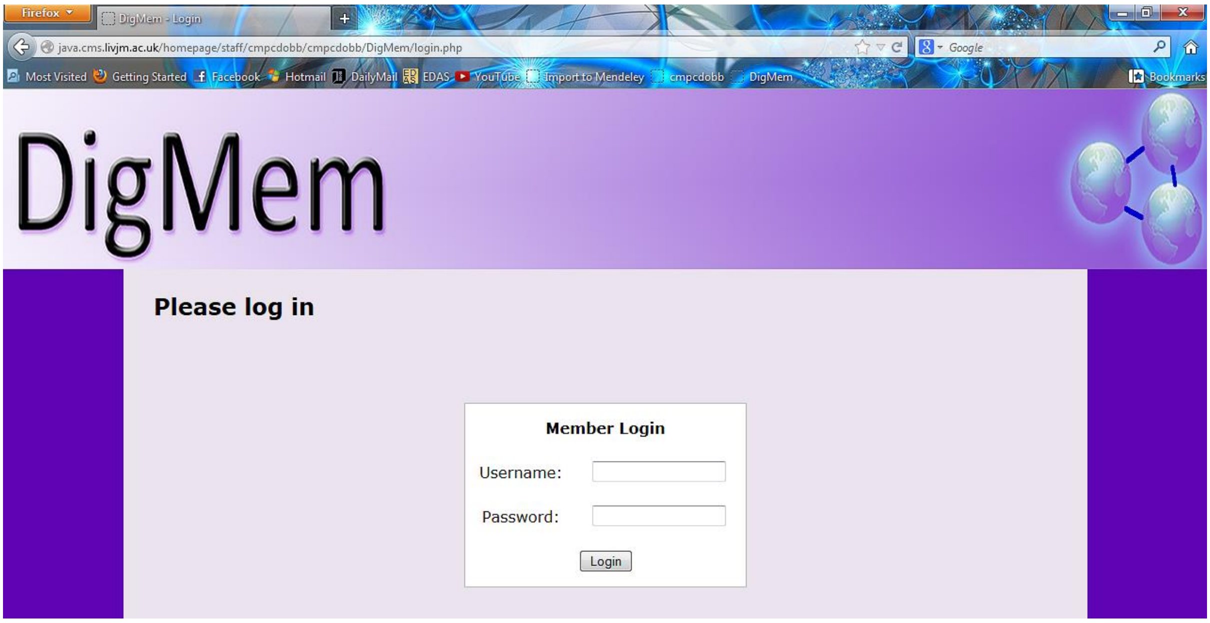
Registration Page Username Validation



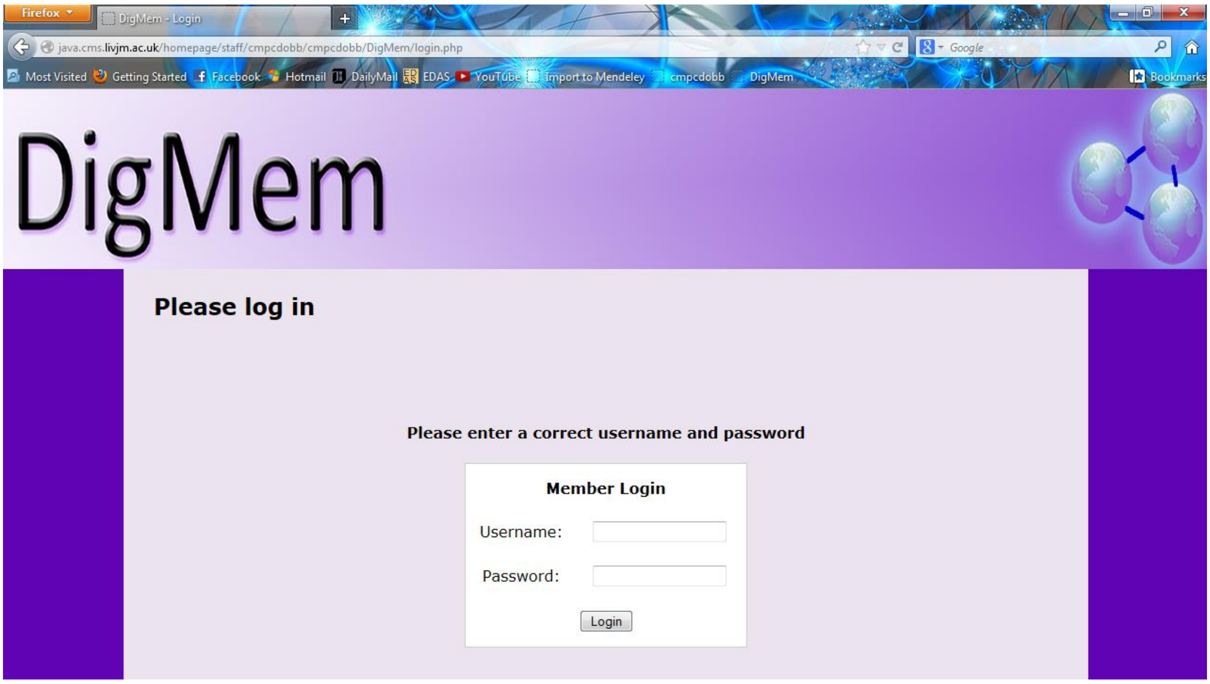
Registration Page Email Validation



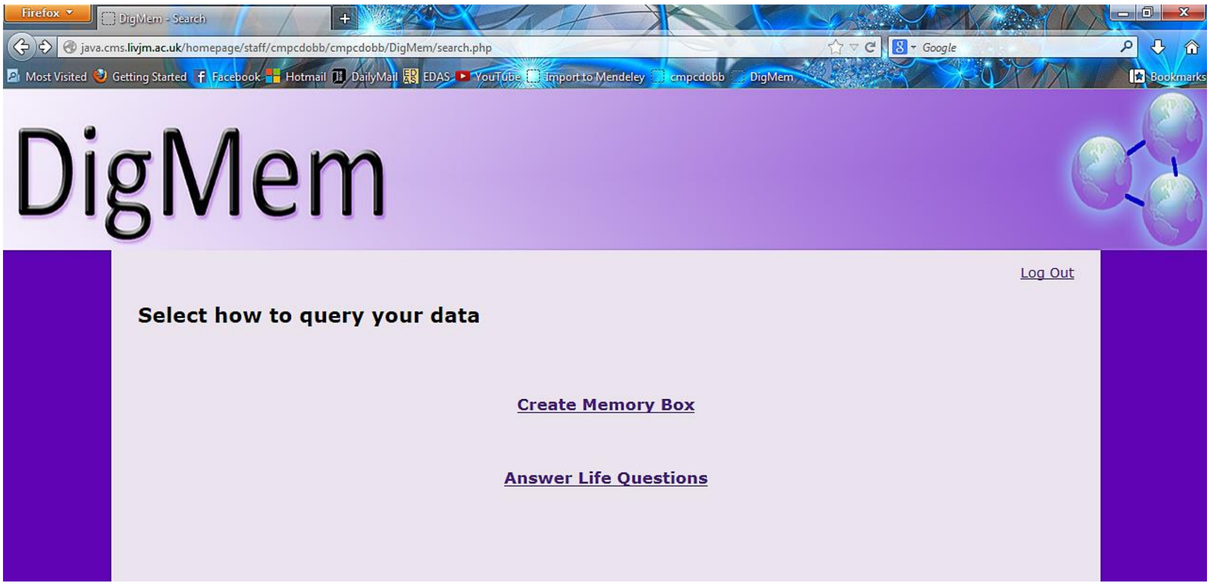
Registration Confirmation Page



Log In Page



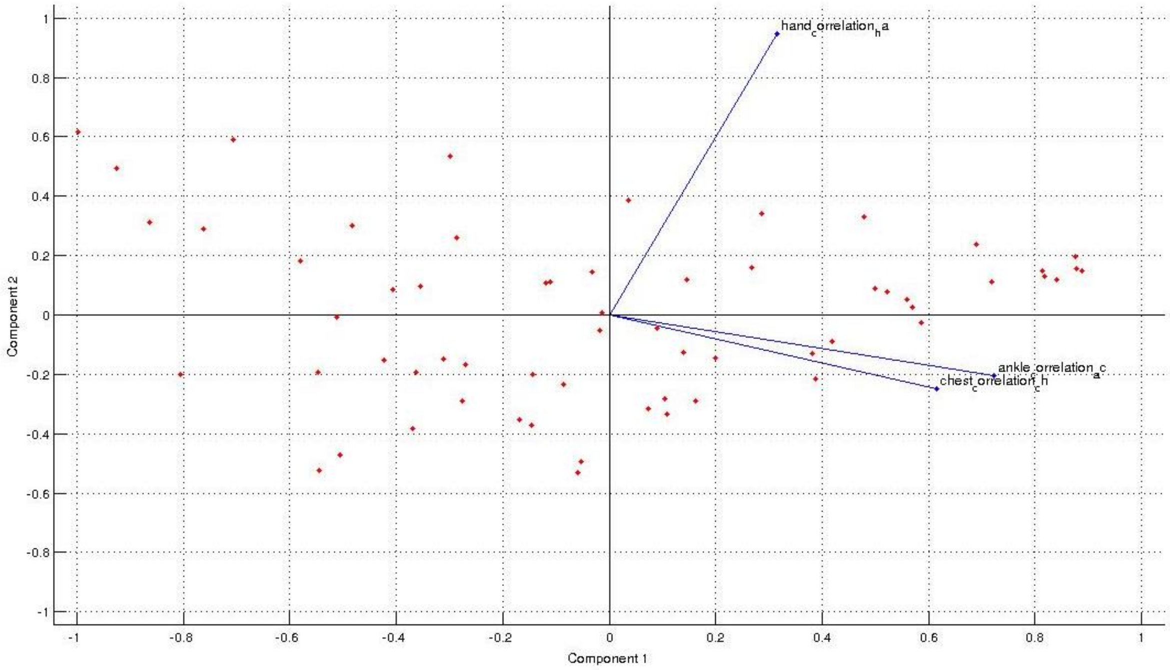
Log In Form Validation

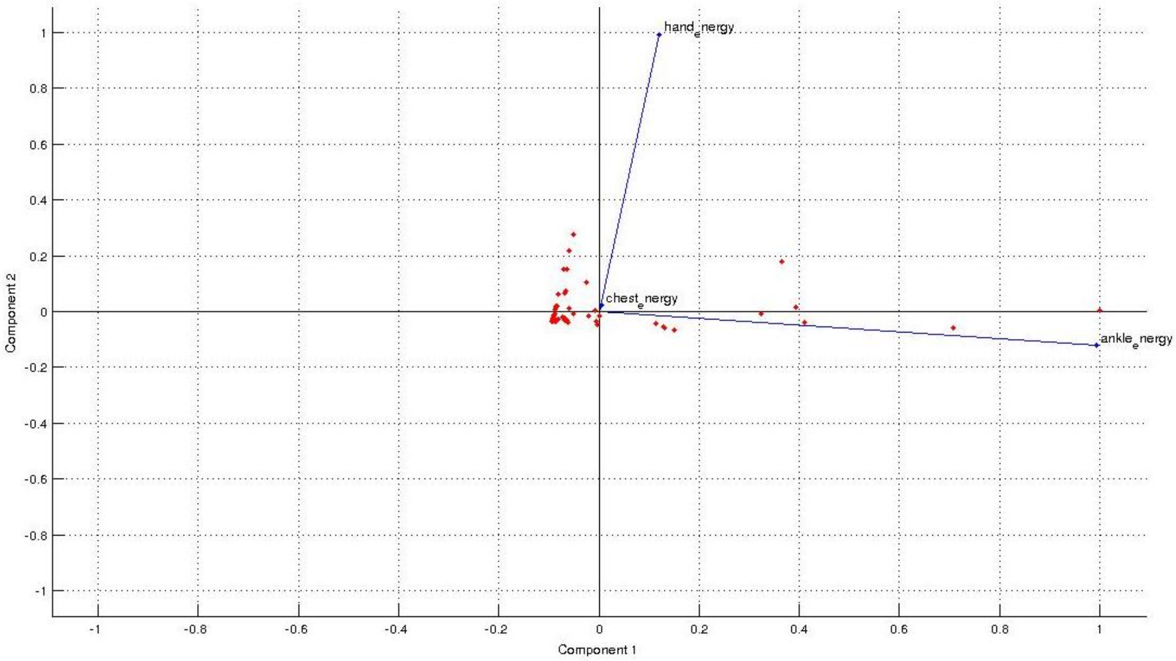


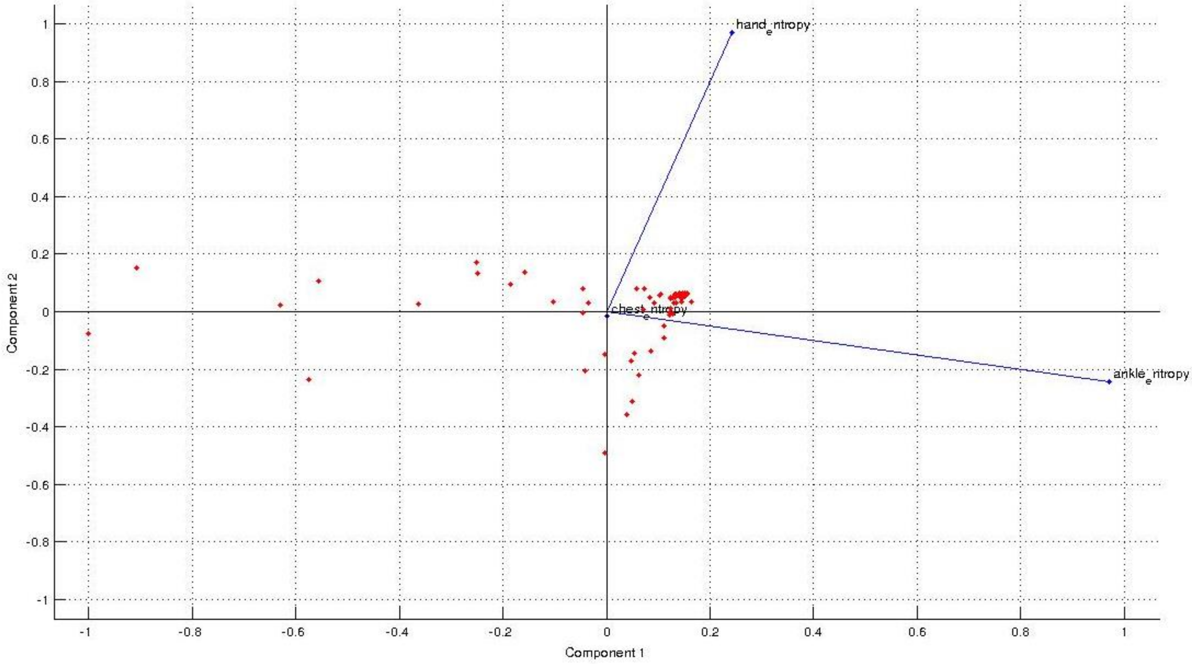
Options Page

Appendix 3 –Principle Component Analysis (PCA) Results

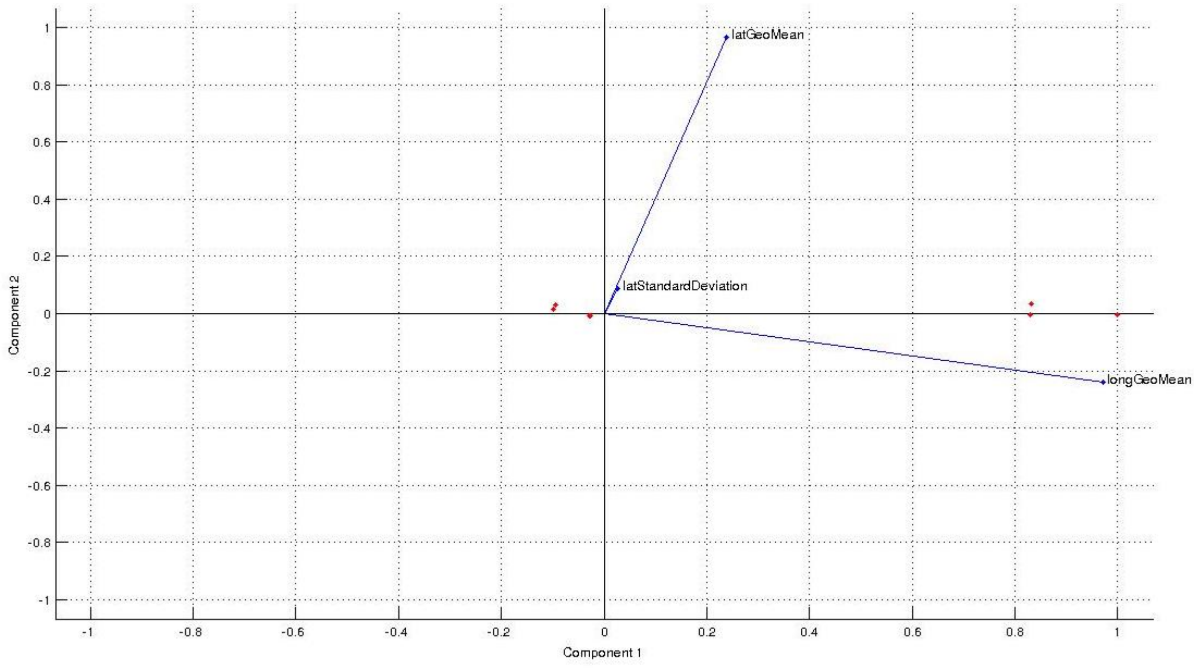
3.1 Round 1 of PCA



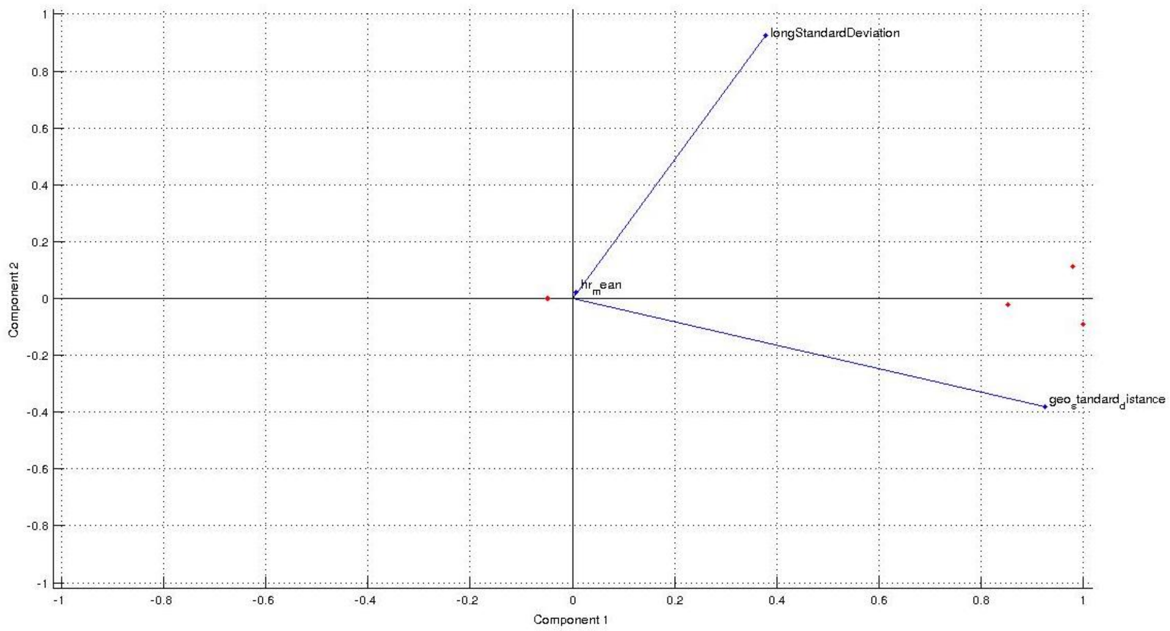


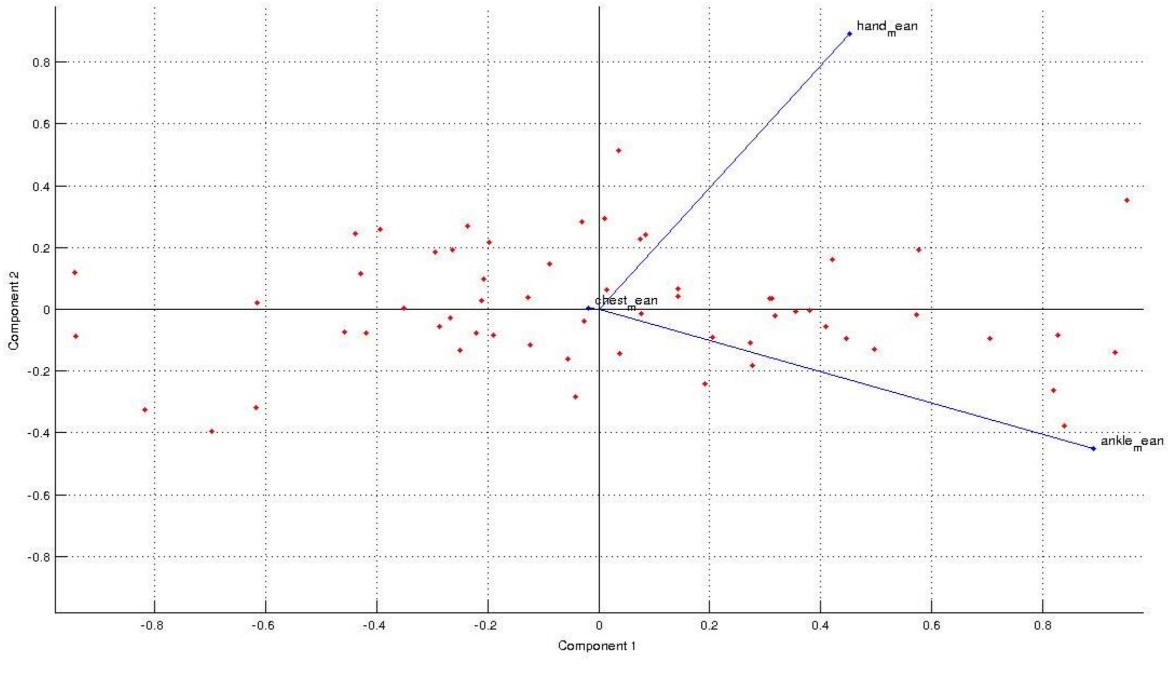


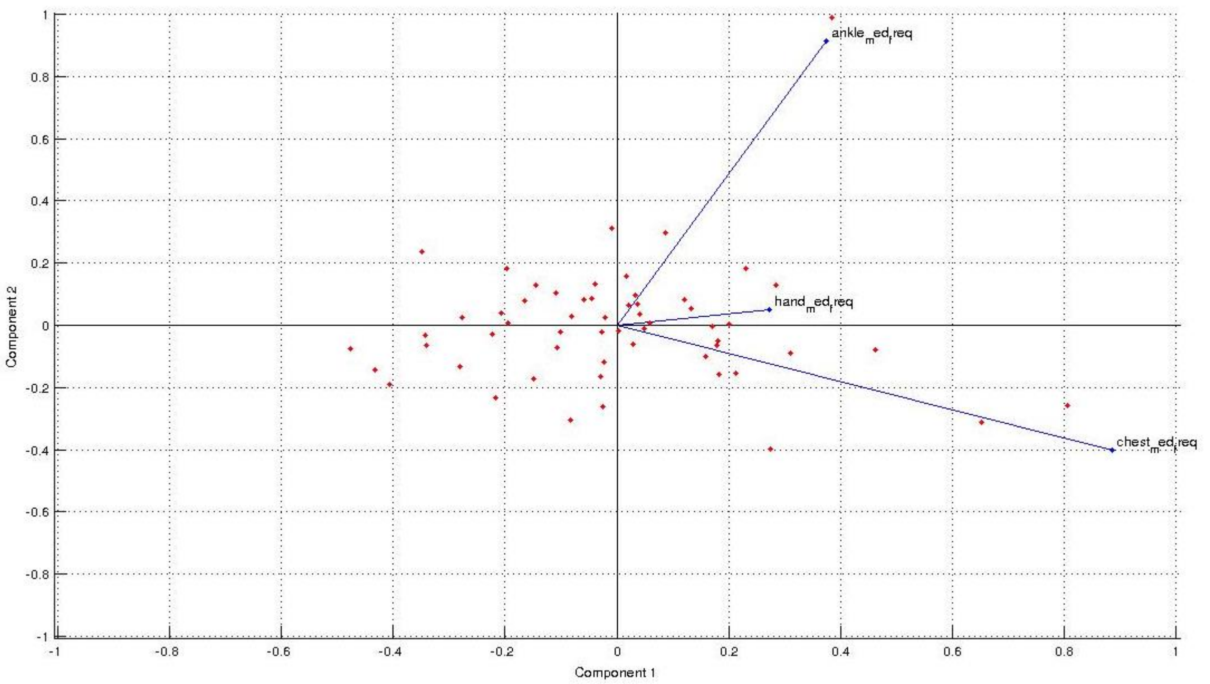
hand_entropy
chest_entropy
ankle_entropy
Component 1
Component 2



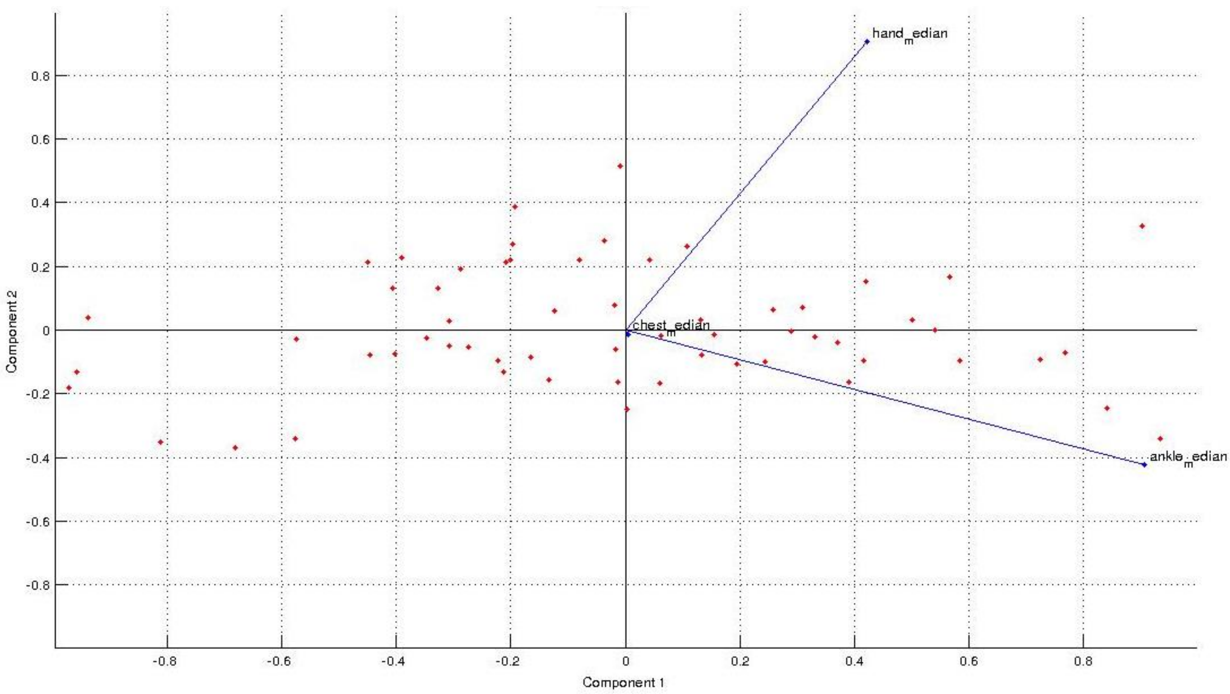
latGeoMean
latStandardDeviation
longGeoMean
Component 1
Component 2


longStandardDeviation
hr_mean
geo_standard_distance
Component 2
Component 1

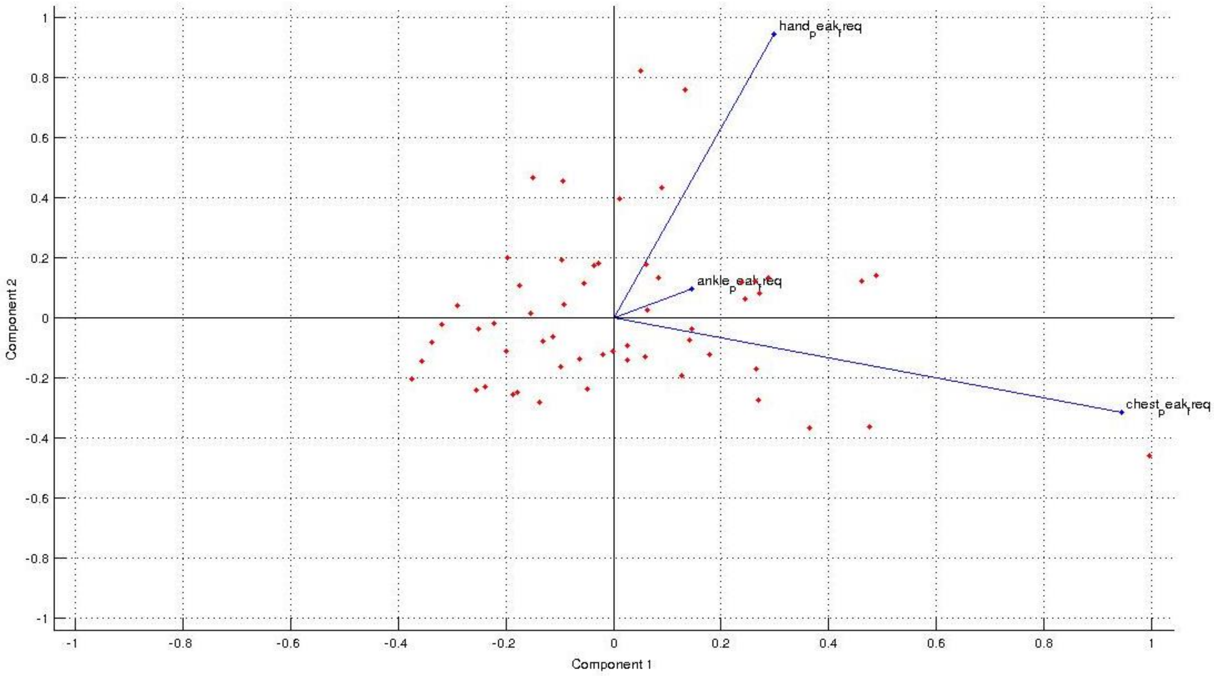

hand_mean
chest_mean
ankle_mean
Component 2
Component 1

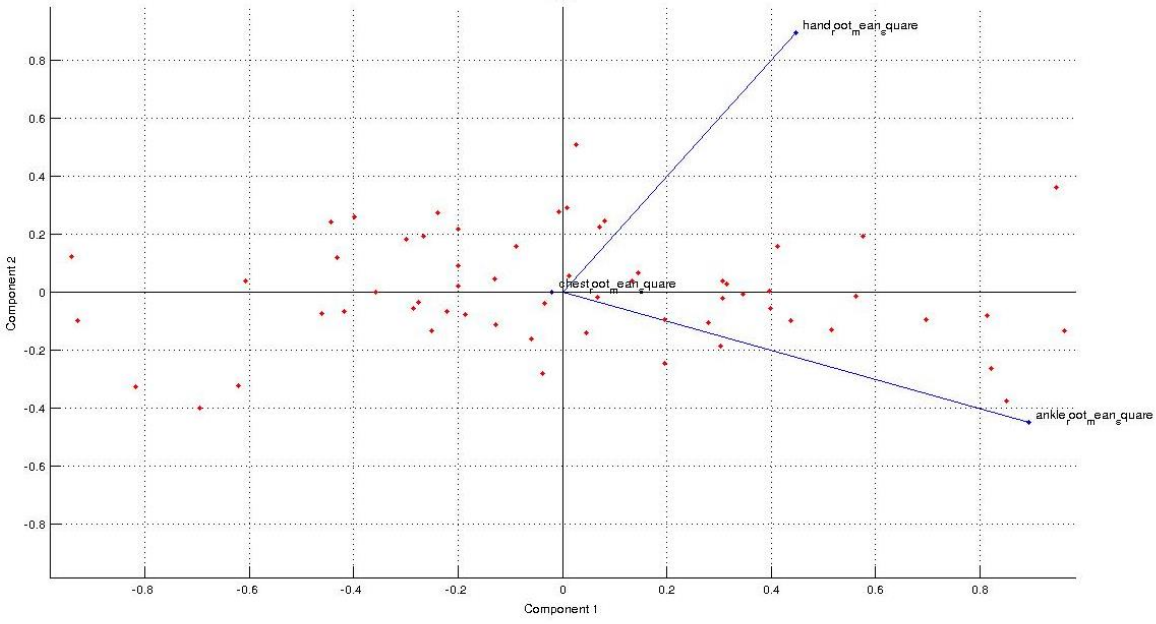


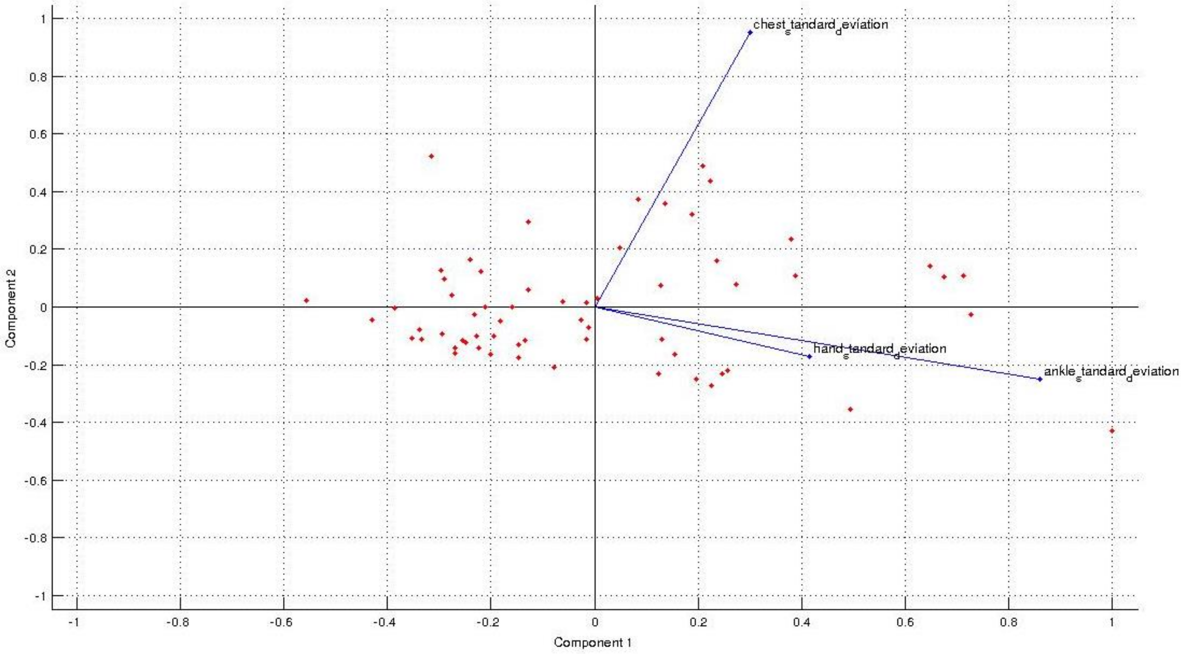
ankle_med_freq
hand_med_freq
chest_med_freq
Component 2
Component 1

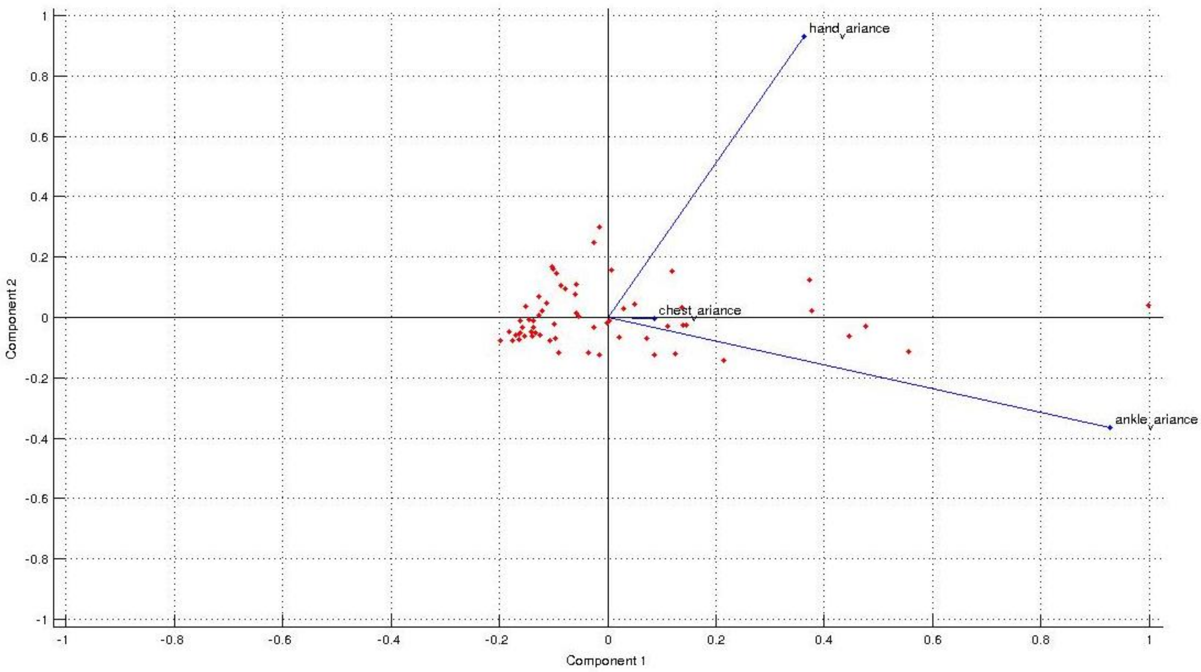


hand_median
chest_median
ankle_median
Component 2
Component 1

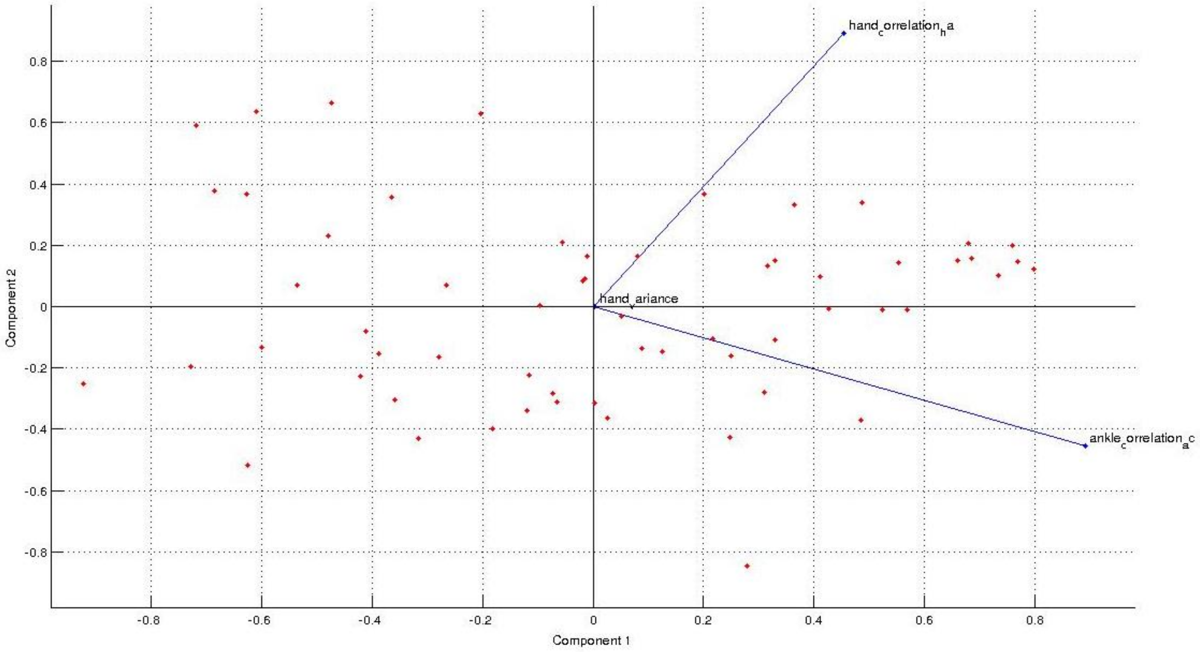

hand_peak_freq
ankle_peak_freq
chest_peak_freq
Component 1
Component 2

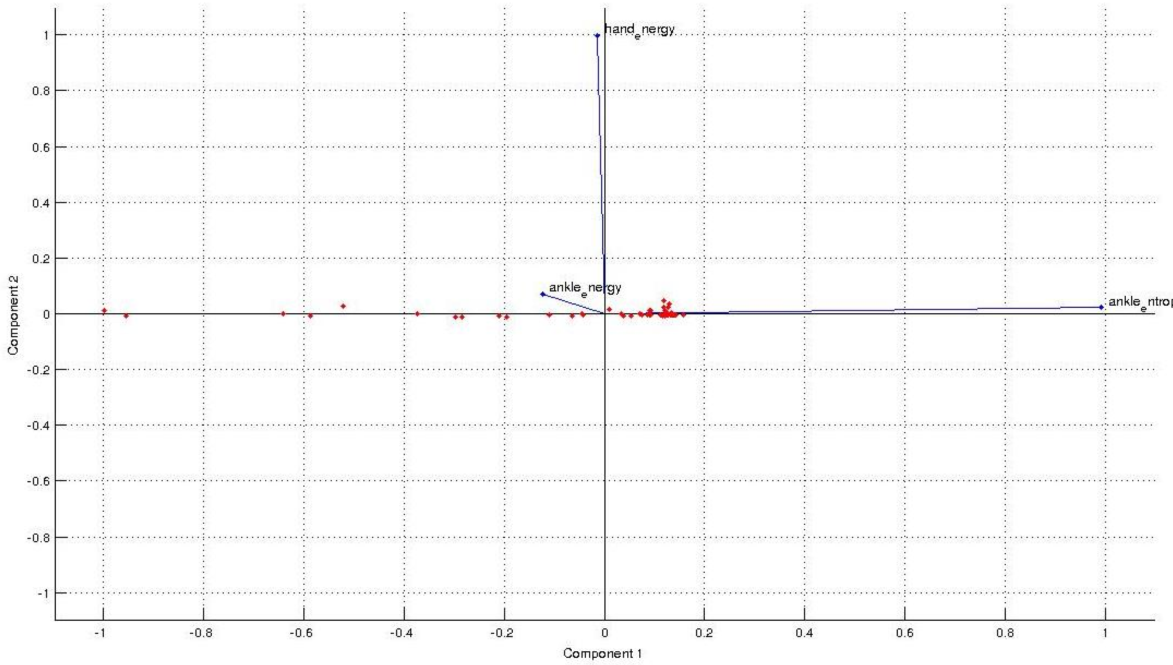

hand_root_mean_square
chest_root_mean_square
ankle_root_mean_square
Component 1
Component 2

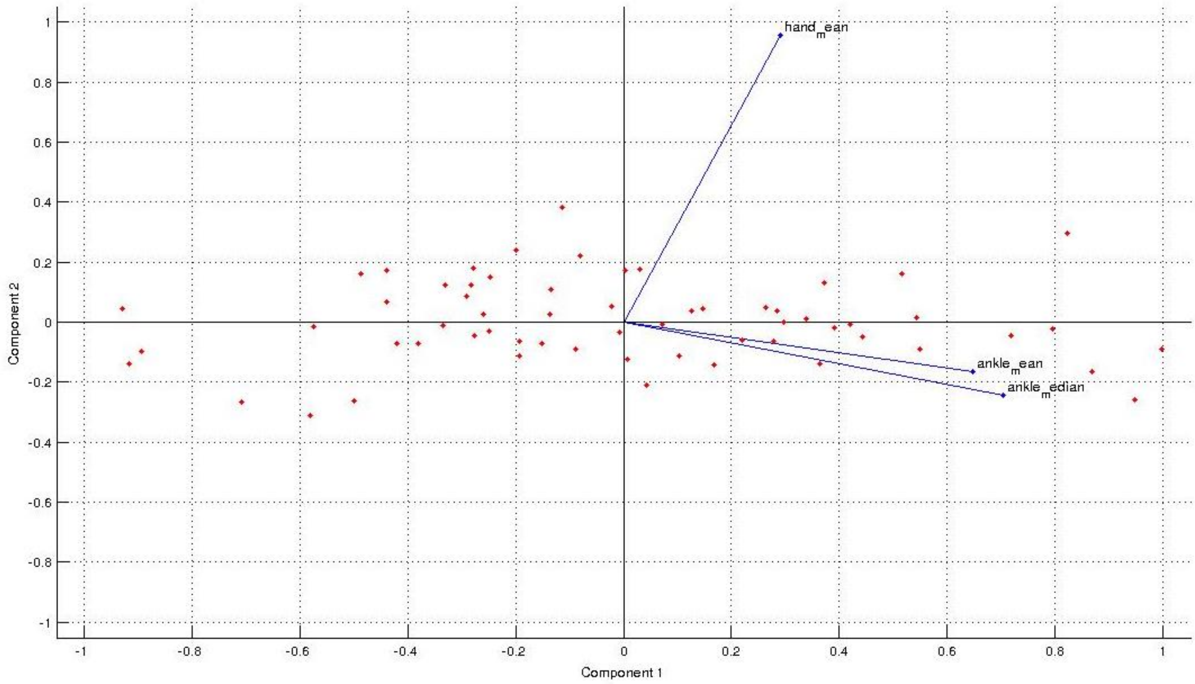

chest standard deviation
hand standard deviation
ankle standard deviation
Component 2
Component 1


hand variance
chest variance
ankle variance
Component 2
Component 1

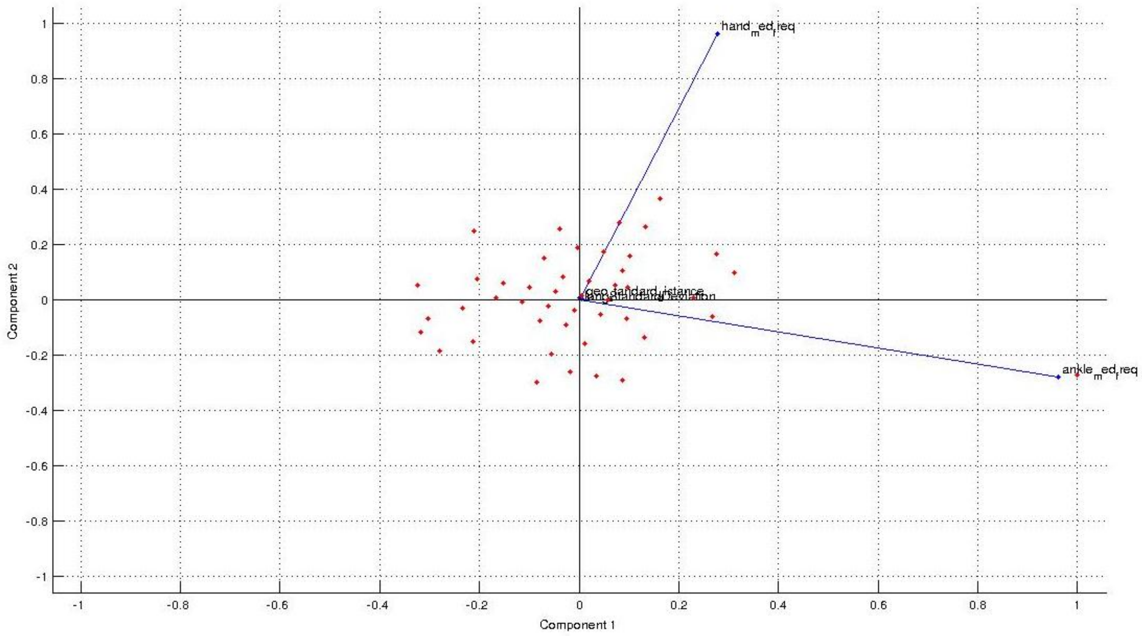
3.2 Round 2 of PCA



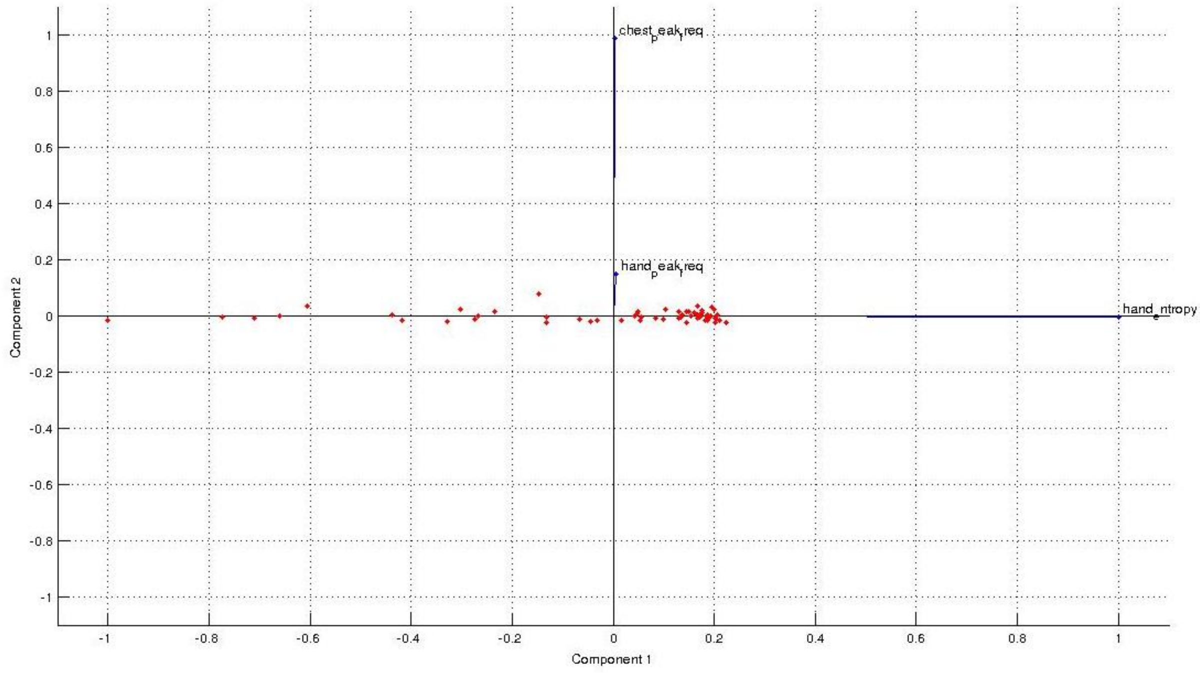


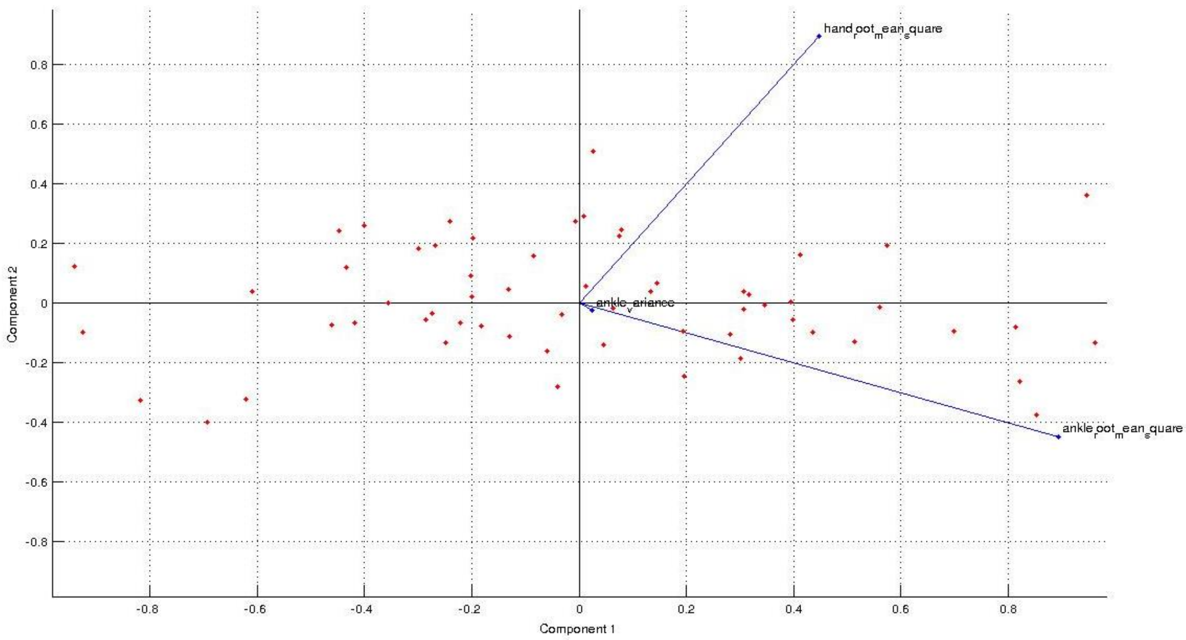


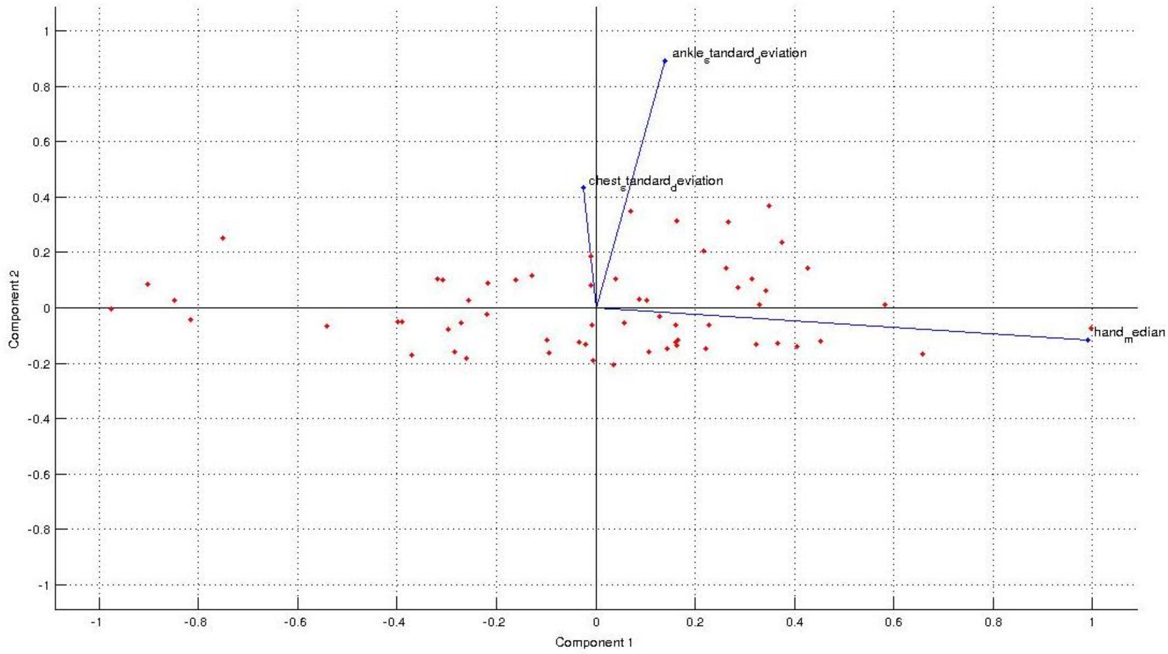
hand_mean
ankle_mean
ankle_median
Component 1
Component 2



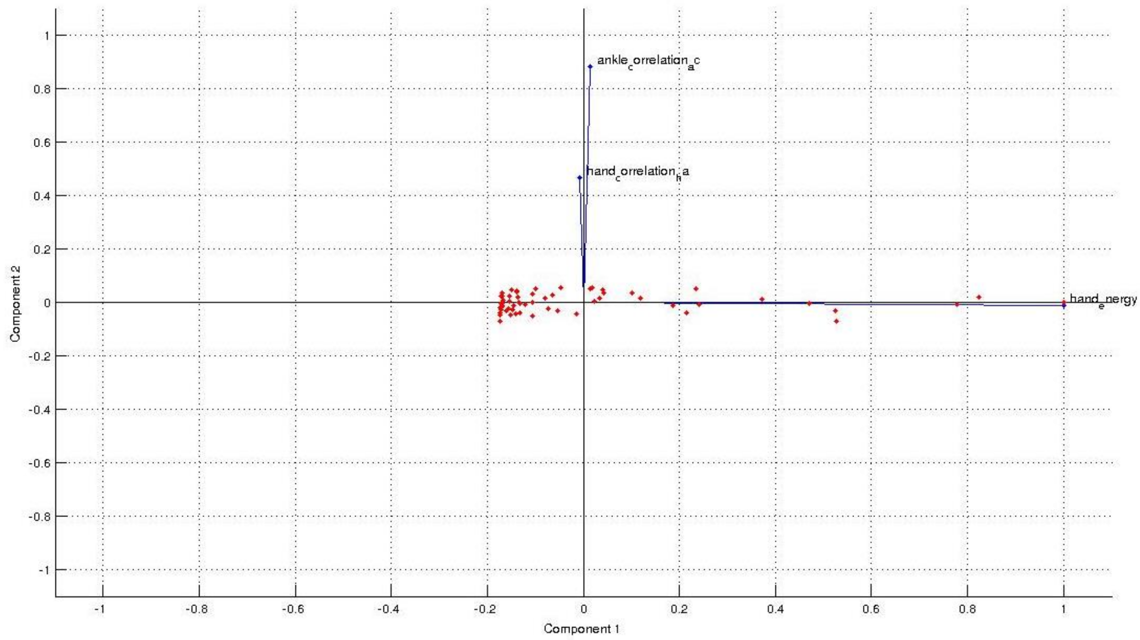
hand_med_freq
ankle_med_freq
Component 1
Component 2

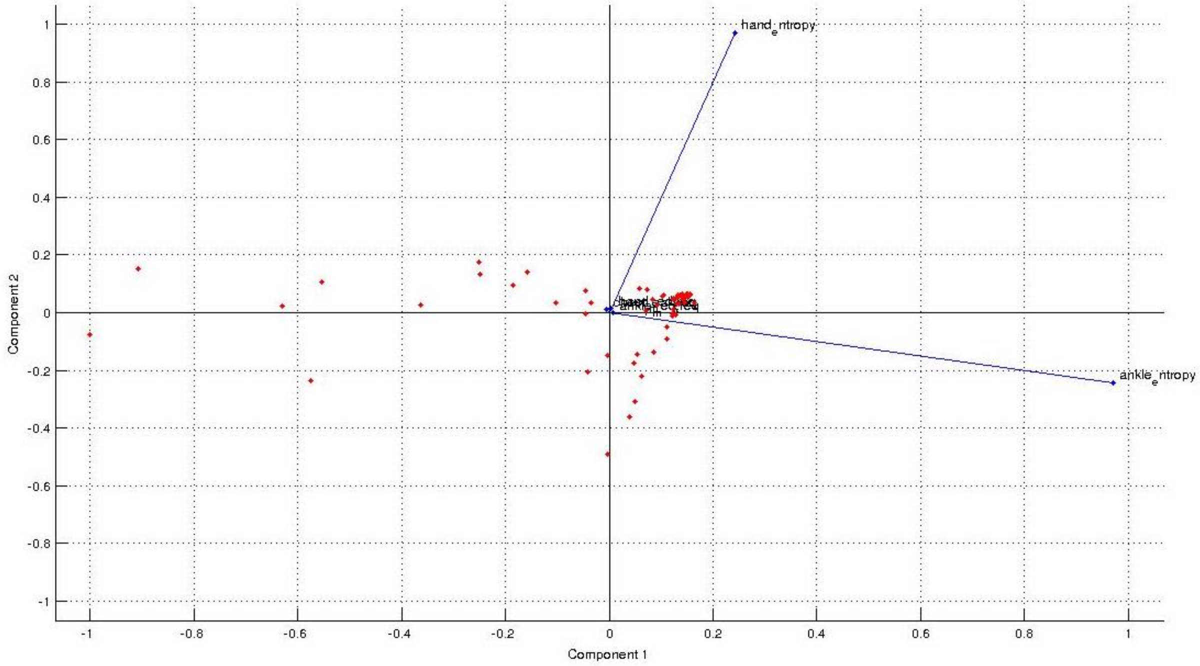

chest_peak_freq
hand_peak_freq
hand_entropy
Component 2
Component 1

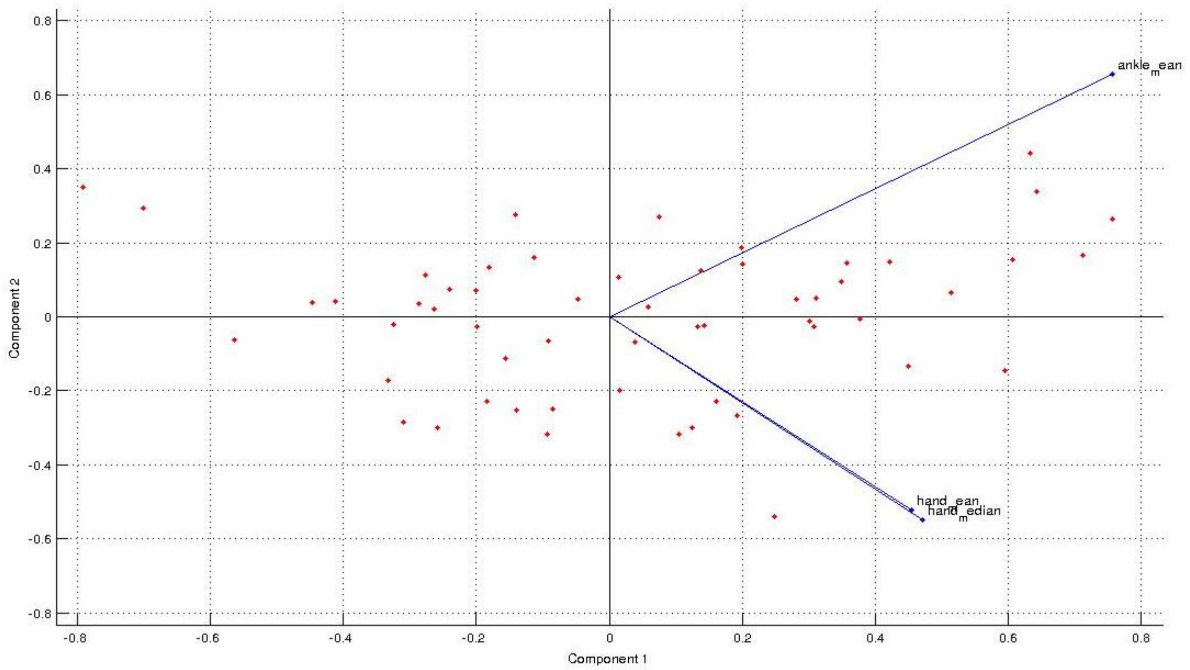

hand_root_mean_square
ankle_variance
ankle_root_mean_square
Component 2
Component 1

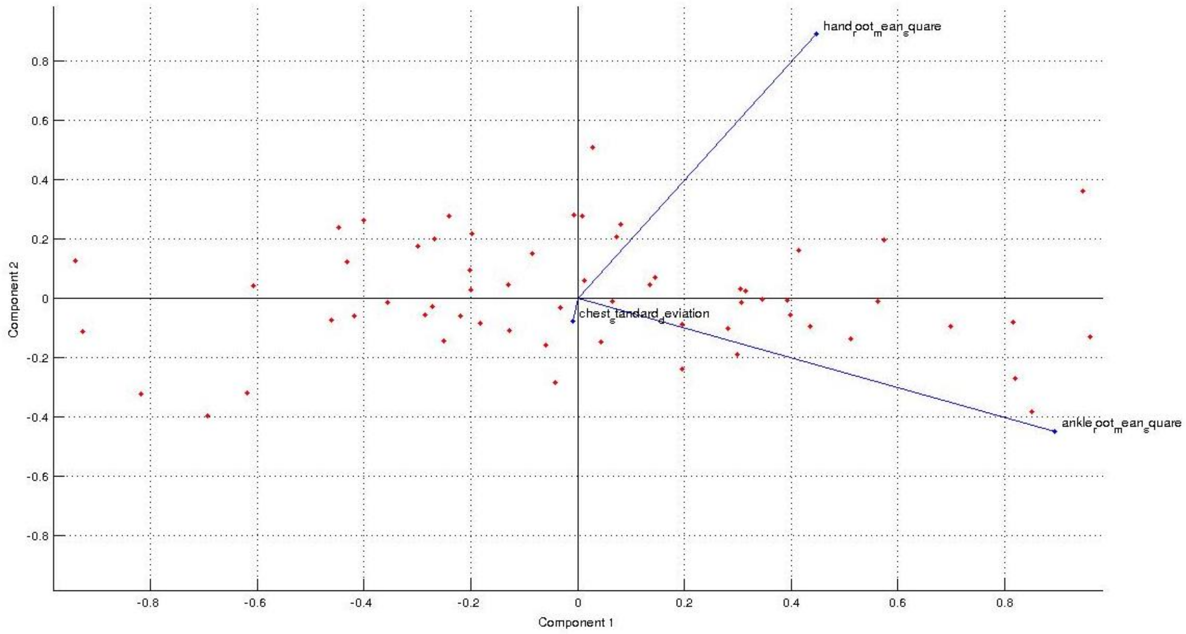


3.3 Round 3 of PCA




hand_entropy
ankle_entropy
Component 1
Component 2


ankle_mean
hand_mean
hand_median
Component 1
Component 2



hand_root_mean_square
chest_standard_deviation
ankle_root_mean_square
Component 1
Component 2

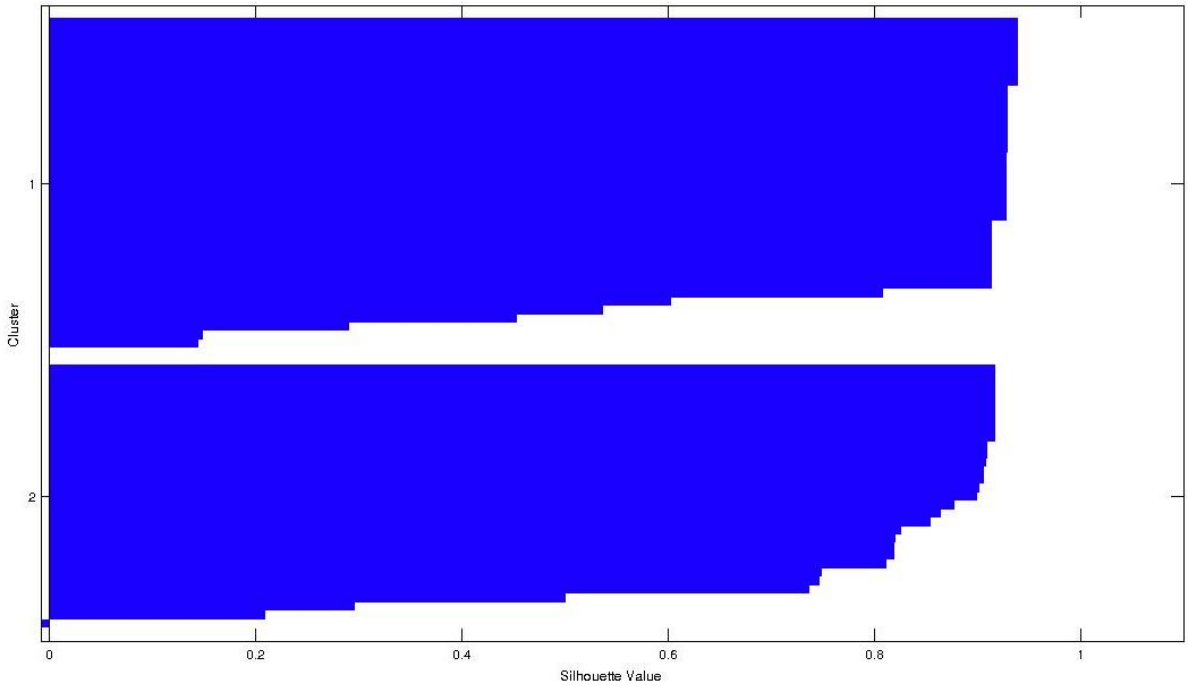
Appendix 4 – Silhouette Averages and Plots For High And Low Energy Activities – Oversampled using SMOTE

4.1 *CorrFeature_Oversampled1* dataset

4.1.1 High Energy

Cluster	Silhouette Average (SA)
2	0.816518155
3	0.757490357
4	0.743033053
5	0.622674459

CorrFeature_Oversampled1 dataset – Silhouette Averages for *k* Clusters of high-energy activities

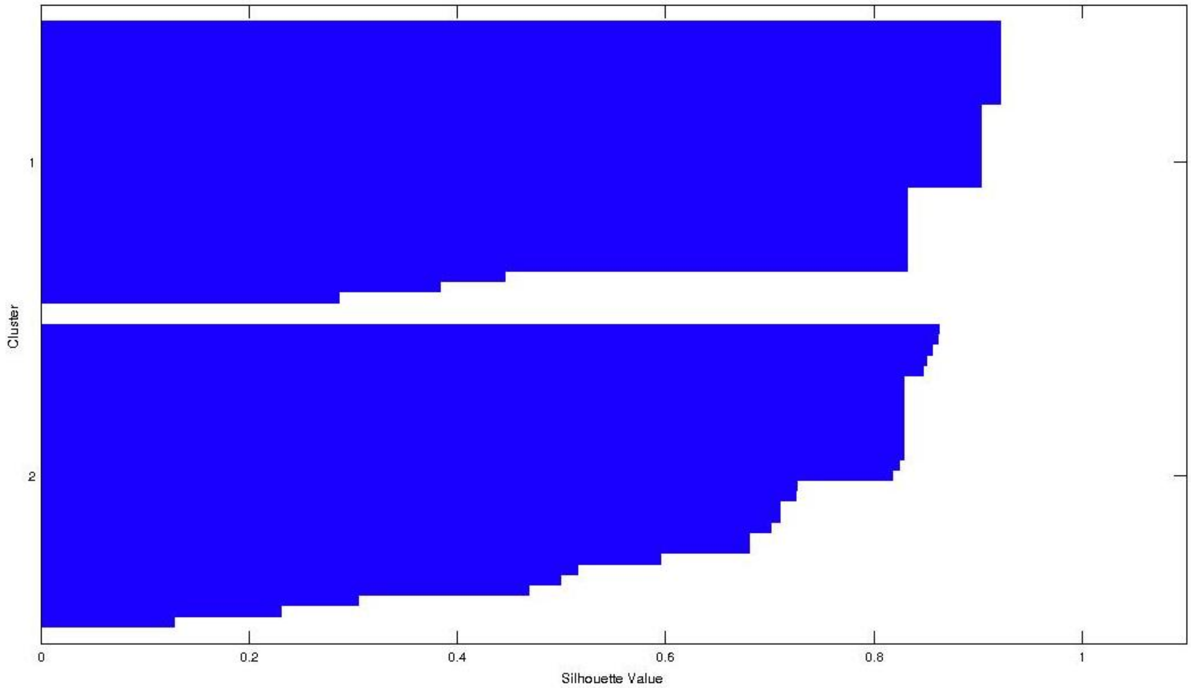


CorrFeature_Oversampled1 dataset – Silhouette Plot of high-energy activities

4.1.2 Low Energy

Cluster	Silhouette Average (SA)
2	0.760590375
3	0.631320152
4	0.592326878
5	0.632392758

CorrFeature_Oversampled1 dataset – Silhouette Averages for *k* Clusters of low-energy activities



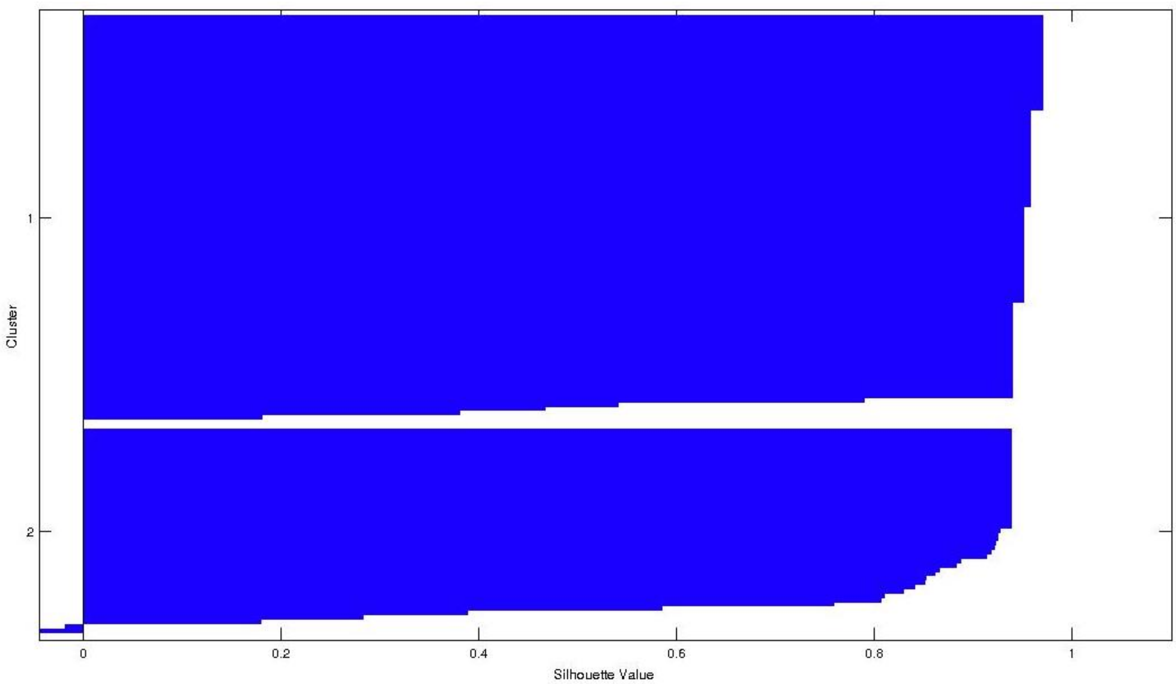
CorrFeature_Oversampled1 dataset – Silhouette Plot of low-energy activities

4.2 *CorrFeature_Oversampled2* dataset

4.2.1 High Energy

Cluster	Silhouette Average (SA)
2	0.892569735
3	0.834498032
4	0.800044318
5	0.78754961

CorrFeature_Oversampled2 dataset – Silhouette Averages for *k* Clusters of high-energy activities

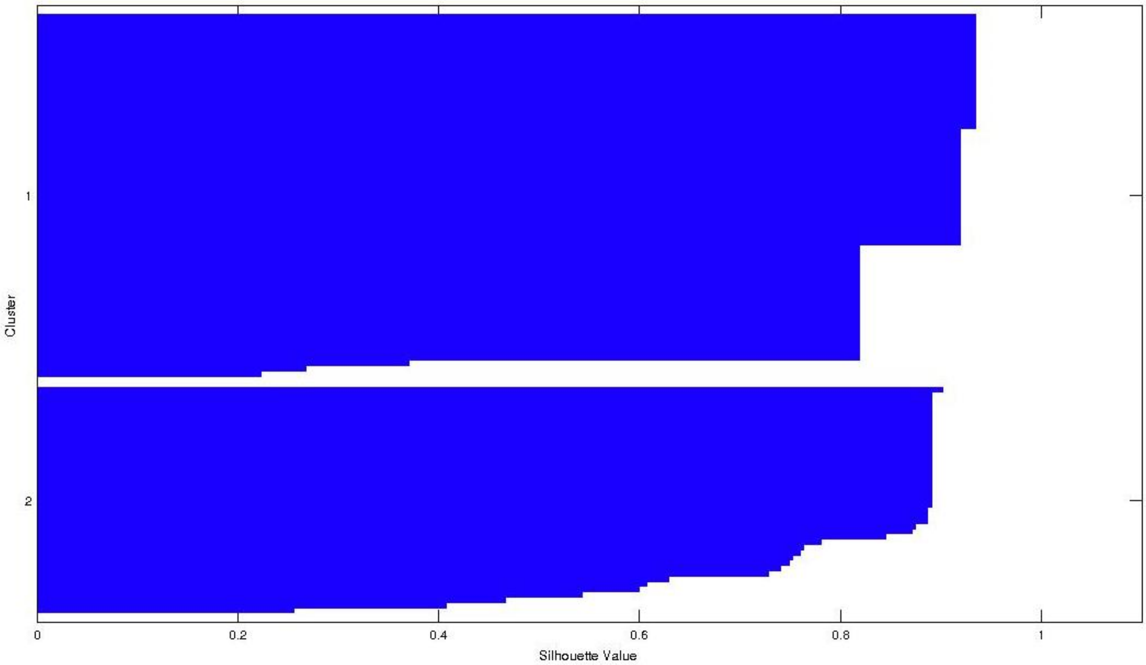


CorrFeature_Oversampled2 dataset – Silhouette Plot of high-energy activities

4.2.2 Low Energy

Cluster	Silhouette Average (SA)
2	0.84106459
3	0.76157969
4	0.760805618
5	0.83445121

CorrFeature_Oversampled2 dataset – Silhouette Averages for *k* Clusters of low-energy activities



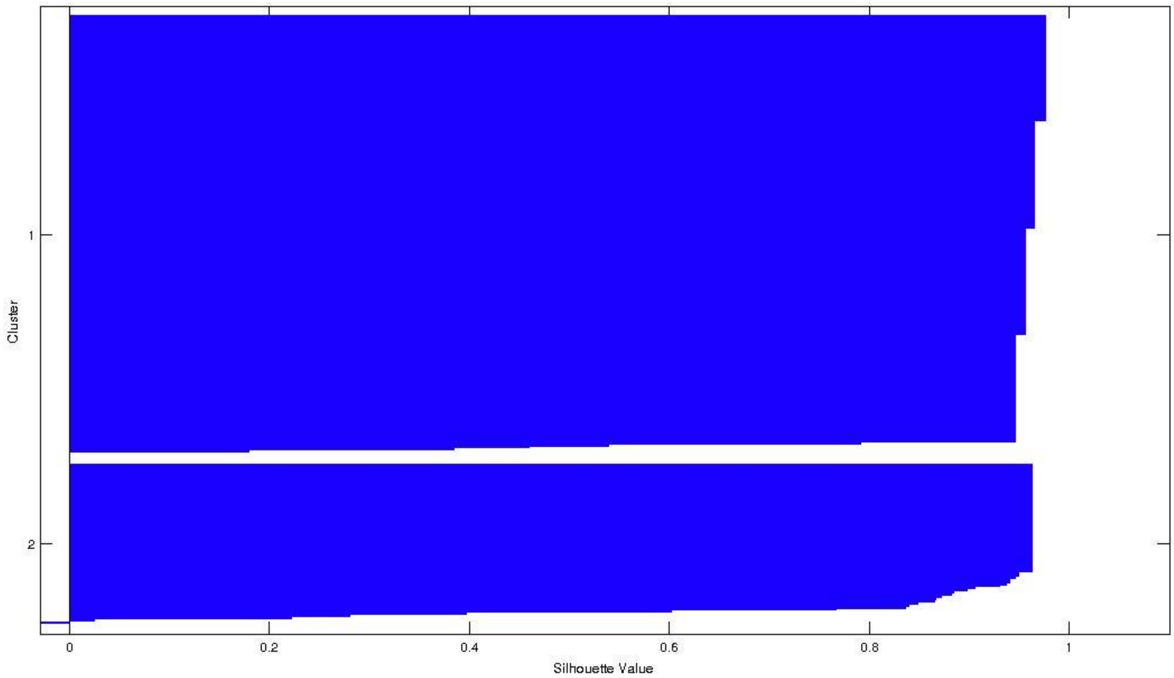
CorrFeature_Oversampled2 dataset – Silhouette Plot of low-energy activities

4.3 *CorrFeature_Oversampled3* dataset

4.3.1 High Energy

Cluster	Silhouette Average (SA)
2	0.932473851
3	0.824121039
4	0.682506776
5	0.643222595

CorrFeature_Oversampled3 dataset – Silhouette Averages for *k* Clusters of high-energy activities

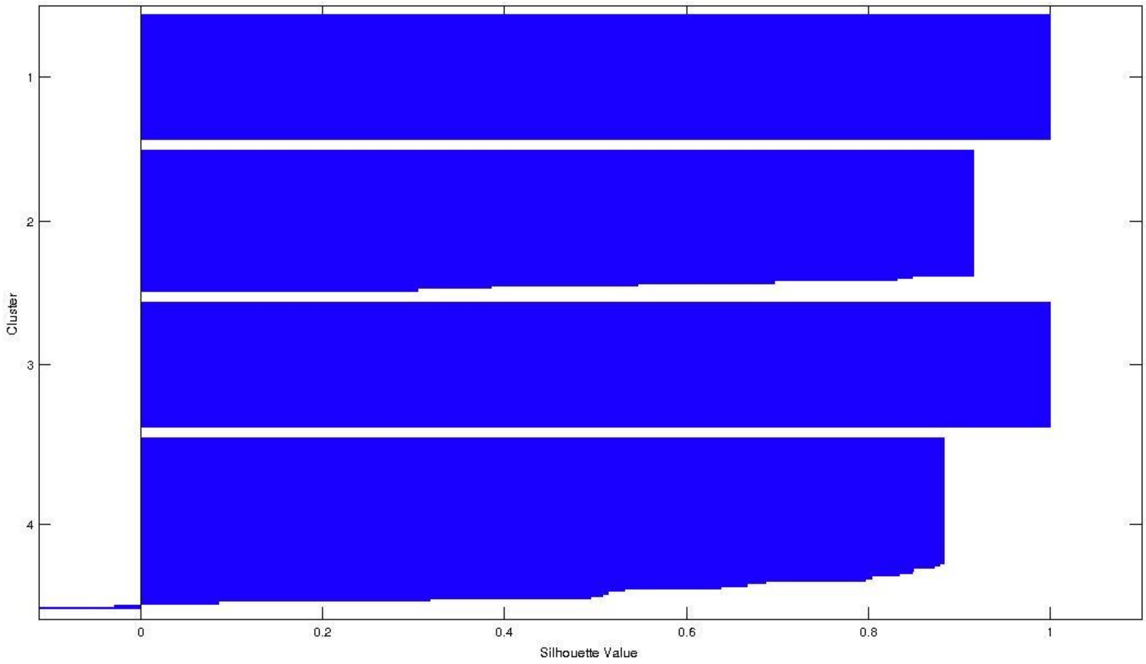


CorrFeature_Oversampled3 dataset – Silhouette Plot of high-energy activities

4.3.2 Low Energy

Cluster	Silhouette Average (SA)
2	0.877788159
3	0.82303278
4	0.909655531
5	0.89629171

CorrFeature_Oversampled3 dataset – Silhouette Averages for *k* Clusters of low-energy activities



CorrFeature_Oversampled3 dataset – Silhouette Plot of low-energy activities