2012

An adaptive framework to provide personalisation for mobile learners

Ahmed Al-Hmouz
University of Wollongong, ahmeda@uow.edu.au

University of Wollongong

AN ADAPTIVE FRAMEWORK TO PROVIDE PERSONALISATION FOR MOBILE LEARNERS

A Dissertation Submitted in Fulfilment of
the Requirements for the Award of the Degree of

Doctor of Philosophy

from

UNIVERSITY OF WOLLONGONG

by

Ahmed Al-Hmouz

Bachelor of Computer Science, Master of Internet Technology, Master of Information Systems

School of Information Systems & Technology (SISAT)
Faculty of Informatics

2012

CERTIFICATION

I, Ahmed Al-Hmouz, declare that this thesis, submitted in fulfilment of the requirements for the award of Doctor of Philosophy, in the School of Information Systems & Technology (SISAT), Faculty of Informatics, University of Wollongong, is wholly my own work unless otherwise referenced or acknowledged. The document has not been submitted for qualifications at any other academic institution.

(Signature Required)

Ahmed Al-Hmouz
14th March 2012

Dedicated to

My Parents

Table of Contents

List of Tables

List of Figures

An Adaptive Framework to Provide Personalisation for Mobile Learners

Ahmed Al-Hmouz

A Thesis for Doctor of Philosophy

School of Information Systems & Technology (SISAT)
University of Wollongong

ABSTRACT

The advancement of technologies in wireless and hand-held devices, combined with the ability to access learning content everywhere and anytime, has created significant interest in mobile learning (m-learning) in recent years. New smart phones are capable of exchanging voice, text, pictures and video. In addition, the new wireless network provides high-speed connections at a low cost to mobile users.

Mobile learning fulfils the promise of learning "on the move" by allowing learners to take control over the time and location of their learning. Learning through a mobile device makes learning truly personalised. Learners have the ability to choose learning content based on their interests, thus making learning learner-centric. In contrast to typical electronic learning (e-learning) products, this access to personalised information means each learner can access the resources they need in a timely manner while minimising wasted bandwidth. Providing immediate access to relevant and interesting information, based on the individual learner's requirements, encourages use and increases engagement because learners are able to access the information they want wherever they are. Mobile learning is still in its infancy and research indicates that few projects have produced any lasting outcomes. Other research to date has focused on the connectivity problem of using wireless networks.

The ultimate goal of this thesis is to present the design and implementation of a **Machine Learning Based Framework for Adaptive Mobile Learning**, to provide a logical structure for the process of adapting learning content to satisfy individual learner characteristics by taking into consideration the learner's needs. One main contribution of this thesis is a novel development framework in the field of mobile learning. The framework depicts the process of adapting learning content to satisfy individual learner characteristics by taking into consideration the learner's needs. The system architecture of the context adaptation based learner profile framework is fundamentally grounded on a number of logical layers. This framework provides a way to reduce the complexity of managing different mobile device settings to enhance learning environments.

Delivery options for mobile learning are increasing, however new technologies alone will not improve the experience of mobile learners. There are a number of factors that impact on a typical learning experience, and many more when that learning experience becomes 'mobile'. This thesis presents an **Adaptive Mobile Learning Content Framework** that describes the factors that play an important role in delivering learning content to mobile learners, and their relationship with each other. Once the necessary information is collected about a learner - either automatically (e.g. location,

device, previous usage) or through learner input (e.g. age) - learning content can be adapted to meet the unique and personal needs of that learner within their current context. It allows consideration of individual learning styles and scenarios, device and application capabilities, and material structure, leading to a customisation of the type and delivery format of learning information in response to the learner.

Based on the newly developed frameworks, another major contribution of the thesis is the establishment of an adaptive learner model. Generally speaking, a m-learning adaptive framework provide personalised services to learners in accordance with their current situation or assumptions about each interacting learner. Adaptive systems adapt their own behaviour to suit and find the optimal outcome for a specific learner's needs. An efficient adaptive system is capable of deciding autonomously what to deliver, how to do it and when to do it. It is essential for adaptive systems to gather information about the learner. Without such information about the learner, the adaptive system is not able to adapt itself to the learner's characteristics and preferences. The required information is stored and managed in the form of a learner profile and learner model. The construction of the learner model is another main contribution of this thesis. The vast amount of data involved in any successful adaptation process creates complexity and poses serious challenges. The **Enhanced Learner Model** focuses on how to model the learner and all possible contexts in an extensible way that can be used for personalisation in mobile learning.

The learner model is logically partitioned into smaller elements or classes in the form of a learner profile, which can represent the entire learning process. Learner profile contains learner's preferences, knowledge, goals, plans, place and possibly other relevant aspects that are used to provide personalised learning content. This thesis presents a Neuro-Fuzzy model for delivering adapted learning content to mobile learners. The adaptation of learning content is based on **Adaptive Neuro-Fuzzy Inference System (ANFIS)**. ANFIS has been recognized for its flexible and adaptive characteristics. ANFIS is a powerful approach to develop fuzzy systems that are capable of learning by providing IF-THEN fuzzy rules in linguistic form. The ANFIS approach is adopted to determine all possible conditions; these cannot be determined by using individual techniquea. The detailed simulation results demonstrated that ANFIS would help the adaptive system to determine a suitable learning content format.

KEYWORDS: Mobile Learning, Adaptation, Learner Model, Personlisation

Acknowledgements

I would never have been able to finish my thesis without the mercy of God (Allah S.W.T), guidance of my supervisors, help from friends, and support from my family.

First and foremost I am thankful to Almighty Allah who gave me knowledge and power to **ALLOW ME** to complete my thesis.

I would like to express my gratitude to my supervisors, Dr. Jun Shen and Dr. Jun Yan for their guidance, knowledge, help and patience.

I would like to express my thanks to all my friends in Australia for their support and encouragement during this journey. No words can express my gratitude to a close friend Dr. Alison Freeman and her beautiful daughter Zahra.

Very special appreciation and gratitude goes to the people I hold most dear, my parents for their love, care, support and prayers. I am also thankful to all my brothers and sisters for their support, encouragement, and patience during my study.

Chapter 1

Introduction

1.1 Motivations of the Thesis

This thesis deals with challenging issues in mobile learning. Fischer stated [52] ”the challenge in an information-rich world is not only to make information available to people at any time, at any place, and in any form, but specifically to say the right thing at the right time in the right way”.

In a traditional classroom, where the education process take place, teachers and learners meet face to face at the same time and in the same place. The learners typically receive the learning materials from the teacher in advance. As a result, the learning activities are limited to those prepared by the teacher; it is usually difficult to adapt the learning materials to each individual learner's learning needs.

In contrast, electronic learning (e-learning), does not require the teacher and learner to be present at the same location. Instead, they typically communicate using wired infrastructures. The popularity of e-learning is growing rapidly to support and assist learning participants (teachers and learners) through the use of learning tools, such as emails, learning content and Web-sites.

Most current learning content used in e-learning was designed for use with desktop computers and high-speed network connections. Learning content usually contains rich media data such as images, audio and video. Therefore, learning content may not be suitable for presentation on devices with limited capability and limited network bandwidth. Another widespread problem in e-learning environments is that learning content does not offer personalisation for the learner and is only able to present identical content to all the learners.

The use of mobile devices for e-learning creates additional challenges for the delivery of adapting learning content. The limited space on mobile device screens does not allow all the learning content available on a traditional Web-site to be presented. Moreover, the limited data transfer rate and processing power of mobile devices often

causes frustration for learner using a mobile device to access learning content, with internet connection speeds consistently lower than using a wired connection.

Mobile based education is already reaching a large number of learners and it offers valuable advantages over traditional teaching. It provides the possibility of adaptation based on individual learners, which is hard to achieve in the common teaching process. Mobile learning through the use of mobile technology allows learners to access learning content from anywhere and at anytime. As a result, learners have full control of the time and location of their learning; they are not limited by a class timetable or classroom location. Learners are also able to choose the content they want to learn; they do not have to learn what the teacher has prepared in advance and scheduled for that class. Learners can use their mobile device to access any learning content from the Internet. "On the Move" learning allows the learner to access and apply information immediately to their learning environment, rather than waiting to apply the information at a later time. Despite the numerous advantages of mobile learning, until recently the practicality of mobile learning had been limited due to the quality of the technologies supporting it (such as unpredicted weather conditions limiting connectivity).

Humans make use of their surrounding environment through interaction with other humans; they interpret their current context and react to it. For example, when a student interacts with a peer, he automatically observes his peer's tone of voice and responds in an appropriate manner. Devices such as computers and mobile phones are not able to interpret the surrounding context in the same way as humans during the interaction process. These types of devices cannot make use of available information in a transparent way; therefore this context information must be explicitly supplied to these devices to allow the devices to respond appropriately.

Particularly in mobile learning applications, contextual and personal characteristics

have a strong relationship with each other, i.e. personal characteristics determine a learner's behaviour and the behaviour determines the context. The concept of being context-aware is composed of two elements: personalisation related to the learner, and automatic customisation related to the adaptation process. Personalisation related the learner means that the system knows about the learner and changes its appearance or behaviour according to learner's needs. Automatic customisation or adaptation means that the system creates a model for the learner in an automatic (machine learning) way to suits the learner's situation and needs.

The term 'context-awareness system' refers to a system that can use, extract and interpret context information and adapt its services to the current context. While the concept is simple, the development of such a system is challenging due to the complexity of capturing, representing and processing contextual information. In addition to the difficulties associated with obtaining context information, context-aware systems must reason and make decisions about how to process the context information to deduce useful and meaningful services. Context reasoning is the most challenging issue in the area of adaptation and personalisation due to the nature of context (imprecise context).

Reasoning approaches seek to draw assumptions about individual learners based on their interactions with the system. Techniques that have been used to model learner contexts include *Neural Networks*, *k-Nearest Neighbors*, *Bayesian Network*, *Genetic* and *Fuzzy algorithms*; *Hybrid Systems*, which consist of combinations of different machine learning techniques, can also be used to model learner contexts.

An adaptive mobile learning system is a mechanism that performs adaptation based on a learner model and updates that learner model with new facts derived by the adaptation. A mobile learning application consists of interactive systems which have to deal with imprecise information; its interpretation is typically vague and uncertain.

Machine learning techniques are commonly used for learner modelling because of the complex nature of the relationships between learner contexts that are hard to be represented.

1.2 Outline of The Thesis

This thesis is divided into three sections:

- the theoretical aspects of the research, presented in Chapters 2, 3, 4 and 5
- the practical aspects of the research, presented in Chapters 6, 7 and 8 based on the knowledge and assumptions described in the first section of the thesis; and
- the conclusion of this thesis and suggestions for further research, presented in Chapter 9.

Chapter 2 introduces basic principles of this research, including terminology, and a literature study. Key topics include mobile learning and its technologies; the concepts of distance, electronic and mobile learning; classifications of mobile and wireless technologies; and the platforms of mobile devices. The chapter highlights the move from e-learning to m-learning, and describes the concept of adaptation, personalisation and learner model. Finally, the chapter gives an introduction into the fundamentals and related work of adaptive m-learning applications.

Adaptive m-learning systems require a logical structure for the process of adapting learning content. Chapter 3 presents a framework that depicts the process of adapting learning content to satisfy individual learner's needs. The framework is to provide personalisation and tackle adaptation using machine learning based algorithm for acquiring, representing, storing, reasoning and updating each learner acquired profile. Therefore, the content and the structure of adaptive m-learning framework is described.

There are a number of factors that impact on a typical learning experience, and many more when that learning experience becomes 'mobile'. Chapter 4 presents a framework to describe the factors that play an important role in delivering learning content to mobile learners, and their relationship with each other. The learning content framework allows consideration of individual learning styles and scenarios, device and application capabilities, and material structure, leading to a customisation of the type and delivery format of learning information in response to the learner. Ultimately, the personalised response to each learner (whether they are working independently or in communication with other learners) improves learner engagement and the overall learning experience, as well as improving efficiency.

Based on Chapters 3, 4 and 4 assumptions, personalisation and learner modelling are becoming more important in the area of mobile learning applications, taking into consideration each learner's interests, preferences and contextual information. The vast amount of data involved in any successful adaptation process creates complexity and poses serious challenges. In order to apply a learner model, it is necessary to construct, initialise and keep the learner model up-to-date. Chapter 5 focuses on how to model the learner and all possible contexts in an extensible way that can be used for personalisation in mobile learning. Challenges and current solutions related to learner modelling are discussed in this chapter, and the Enhanced Learner Model structure to be used in a mobile learning system is proposed.

Additional issues which are not covered in the theoretical section of this thesis, but are important for the proposed solution, are described in Chapter 6. This chapter provides a general overview of Fuzzy Logic (*Fuzzy Sets*, *Membership Functions*, *Fuzzy System Components* and *Inference Algorithm*) and Neural Networks (*Learning Method*, *Types* and *Inference Algorithm*). In Chapters 3, 4 and 5 the Adaptive Mobile Learning Framework, Learning Content Framework and Enhanced Learner Model

were presented for the purpose of constant support for mobile learners. Chapter 6 introduces the principles of Adaptive Neuro-Fuzzy Inference Systems (ANFIS). In this chapter, a general overview of *Neuro-Fuzzy Systems* types, *Fuzzy Inference System (FIS)* and the *ANFIS Hybrid Learning* algorithm are provided.

Chapter 7 presents the design and implementation of the modelling system which describes the learner model process of the proposed approach. It also introduces the Adaptive Neuro-Fuzzy Inference System (ANFIS) as a reasoning engine to deliver learning content for mobile learning application.

The methodology of the research employs a two-phased approach. The first phase is to build a theoretical framework that represents the mobile learning contexts and adaptation process (initiation, proposal, selection and production of adaptation) as explained in Chapters 3, 4 and 5. Chapter 7 and 8 introduce the second phase of this research which is the practical phase. This phase involves data processing, ANFIS model coding, simulation, assumptions, results, adaptation and training of the data set using MATLAB.

Chapter 8 presents a series of simulations that were conducted to illustrate the potential effectiveness of ANFIS with hybrid learning, for the adaptation of learning content format for mobile learners. Various experiments were conducted and the sizes of the training and checking sets were determined by taking into consideration the classification accuracies. The performance of ANFIS was evaluated using standard error measurements which revealed the optimal setting for predictability. Chapter 8 shows that the feasibility of the Reasoning Layer (refer to Section 3.4.6) confirms assumptions made in Chapters 3, 4 and 5 of this thesis. These results were based on analysis of a number of inputs; the research results confirm that the m-learning application is functional.

Chapter 9 of the thesis describes the results of this work and suggestions are made for further research.

1.3 Contributions of the Thesis

The main contributions of this thesis are listed below:

- This thesis contributes to the field of educational technology, by proposing an adaptive framework to address the problem of mobile learning and its limitation (Chapters 3, 4, 5, 7 and 8).
- This thesis' main contribution is in attempting to overcome the limitations of using mobile learning by developing a framework to provide personalisation and tackle adaptation using machine learning technique according to an obtained learner profile for mobile learning purposes. The learner profile contains preferences, knowledge, goals, plans, place and possibly other relevant aspects that are used to provide personalised adaptations. The framework depicts the process of adapting learning content to satisfy individual learner characteristics by taking into consideration the learner's needs.

  The system architecture of our context adaptation based learner profile framework adaptation is fundamentally grounded on a number of logical layers. This framework will offer a way to reduce the complexity inherent from different mobile device settings and enhance learning environments. The framework is designed to adapt learning content in a way that matches the learner's preferences, supports the learning context (location, noise level, device type, availability of resources, network), and is compatible with the learning objectives (Chapter 3).
- This thesis contributes to the efficient use of portable devices (old and new) to accomplish routine learning activities in different contexts; there is limited

literature in this area. The literature mainly focuses on adaptation using new technologies. This thesis develops a framework to describe the factors that play an important role in delivering learning content to mobile learners, and their relationship to each other. It allows consideration of individual learning styles and scenarios, device and application capabilities, and material structure, leading to customisation of the type and delivery format of learning information in response to the learner (Chapter 4).

The concern of content adaptive learning is to develop the strategies and methods for creating learning content that always meet learners' needs. To effectively address the adaptation problem, the problem must be considered from both sides - the learning context and the learning content - by the integration of two frameworks. The first framework deals with the learner context and has been described in Chapter 3.

- Moreover, this thesis investigates the relationship between context, learning content and learning activities considering context as the combination of the physical as well as social features of learning activities. This thesis models the learner and all possible contexts related to his current situation in an extensible way that can be used for personalisation. The proposed learner model consists of four main components, namely: the representation of the *Learner Status* (Section 5.4.1), the *Situation Status* (Section 5.4.2), the *Knowledge and Shared Properties Status* (Section 5.4.3) and the *Educational Activity Status* (Section 5.4.4). It is essential for adaptive systems to gather information about the learner. Without such information about the learner, an adaptive system will not be able to adapt itself to the learner's characteristics and preferences. The required information is stored and managed in form of a learner profile and learner model (Chapter 5).

- This thesis develops a reasoning engine based on machine learning techniques,

the structure of the reasoning engine which adopts hybrid machine learning techniques in two stages (Fuzzy Logic and Neural Networks). This thesis presents a Neuro-Fuzzy model for delivering adapted learning content to mobile learners. The adaptation of learning content is based on the Adaptive Neuro-Fuzzy Inference System (ANFIS).

ANFIS has been recognised for its flexible and adaptive characteristics. ANFIS is a powerful approach to develop fuzzy systems that are capable of learning by providing IF-THEN fuzzy rules in linguistic form. ANFIS approach is adopted to determine all possible conditions which can not be determined through the use of individual techniques (Chapters 6, 7 and 8).

An Adaptive Neuro-Fuzzy Inference System (ANFIS) for a number of cases reflect the mobile learning scenarios has been designed, modeled and simulated (Chapter 7). Confirmation of the analysis results through extensive simulation to test all possible scenarios which could be encountered by learners while accomplishing a learning task (Chapter 8).

1.4 Publications of the Thesis

The following publications have resulted from this work.

- A. Al-Hmouz, J. Shen, J. Yan, "A Machine Learning based Framework for Adaptive Mobile Learning," The 8th International Conference on Web-based Learning (ICWL 2009), Aachen, Germany, published by Springer (LNCS 5686), Aug. pp. 34-43, 2009. The work of this paper is based on Chapters 3, 4, 5, 7 and 8.

- A. Al-Hmouz, A. Freeman, "Learning on Location: An Adaptive Mobile Learning Content Framework," IEEE International Symposium on Technology and

Society (ISTAS), Wollongong, Australia, pp. 450-456, 2010. The work of this paper is based on Chapter 4.

- A. Al-Hmouz, J. Shen, J. Yan, R. Al-Hmouz "Enhanced Learner Model for Adaptive Mobile Learning," The 12th International Conference on Information Integration and Web-based Applications & Services (iiWAS2010), Paris, France, ACM, pp. 781-784, 2010. The work of this paper is based on Chapters 5, 7 and 8.

- A. Al-Hmouz, J. Shen, J. Yan, R. Al-Hmouz "Modelling Mobile Learning System Using ANFIS," The 11th IEEE International Conference on Advanced Learning Technologies (ICALT 2011), Athens, Georgia, USA. The work of this paper is based on Chapter 7 and 8.

- A. Al-Hmouz, J. Shen, R. Al-Hmouz, J. Yan "Modelling and Simulation of an Adaptive Neuro-Fuzzy Inference System (ANFIS) for Mobile Learning," IEEE Transactions on Learning Technologies Journal (TLT), 2011. The work of this paper is based on Chapters 3, 4, 5, 7 and 8.

Chapter 2

Literature Review

2.1 Outline

This chapter presents a literature study on mobile learning and its technologies. The concept of distance, electronic and mobile learning is discussed. In this chapter, the classification of mobile and wireless technologies is presented and the platforms of mobile devices are described. This chapter highlights the move from electronic learning to mobile learning, and describes the concept of adaptation, personalisation and the learner model. The Chapter ends with a review of the literature that covers some of machine learning applications to a variety of educational institutions.

2.2 Introduction

Learning can occur in all situations and environments, however until recently the role of technology in this learning has been limited in many settings. For example, unpredicted weather conditions may affect a learner's ability to accomplish a learning task. In the traditional classroom where the education process takes place, teachers and learners meet face to face at the same time and in the same place. Learners typically receive the learning content from the teacher in advance. As a result, the learning activities are limited to those prepared in advance by the teacher, and it is difficult to adapt the learning materials to each individual learner's learning needs. Key advantages of traditional learning are direct contact between the teacher and students and immediate feedback from the student. However, it also has some disadvantages such as a student missing content if they are unable to attend a lesson.

Distance Learning (d-learning) and Electronic Learning (e-learning), on the other hand, continue to grow rapidly to support and assist learning participants (teachers and learners) through the use of learning tools (emails, learning content Web-site). Most d-learning and e-learning technologies do not require the teacher and learner to

be present at the same location at the same time, and involve wired infrastructures.

m-learning is the intersection of mobile computing and e-learning; m-learning refers to the ability to learn everywhere at anytime without physical connection to cable networks. Mobile computing refers to continuous accessibility to the learner, while wireless implies communicating without wires. Mobile and wireless systems cover two areas: mobility and computing. Mobile and wireless technologies have become popular in a range of areas such as travel, education, trading and the military.

Mobile learning through the use of wireless mobile technology allows anyone to access information and learning materials from anywhere and at anytime. As a result, learners have control over when and where they want to learn. Also, all learners have the ability to access learning content or information to improve their quality of life regardless of the time and location. Mobile learners do not have to wait for a certain time to learn or go to a certain place to learn; mobile learning allows learners to learn whenever and wherever they want. Also, learners do not have to learn what is prescribed to them. They can use the wireless mobile technology for formal and informal learning where they can access additional and personalised learning content from the Internet.

2.3 Distance and Electronic Learning

2.3.1 Distance Learning (d-learning)

The advancement of technologies in communication and transport has created significant interest in adopting new forms of education such as d-learning [76]. The concept of d-learning is based on the possibility of offering learning services at distance where teachers and students are separated by location and time. In order to overcome the barrier of location and time, communication technologies are used to close the gap and

support the learning process [62].

d-learning has three characteristics as identified in [76]:

- the separation of the teacher and the learner,
- the separation of the learner from the learning group, and
- the technology being used to communicate with an educational institution.

d-learning offers the learner the chance to work from home and browse the learning content whenever is convenient to them. By providing the teacher and learner with technology aid tools, it is possible to have an external element to the learning process such as experts who can join the class using video conferences from remote locations. Such technologies and tools enhance learner motivation and open other choices in terms of learning.

2.3.2 Electronic Learning (e-learning)

e-learning is defined as "the effective learning process by combining digitally delivered content with (learning) support and services" [14]. Urdan and Weggen [142] define e-learning as "the delivery of content via all electronic media, including the Internet, intranets, extranets, satellite broadcast, audio/video tape, interactive TV, and CD-ROM.".

Paulsen [113] listed some characteristics of online education which are similar to those of e-learning:

- the separation of teachers and learners which distinguish it from face to face learning,
- the influence of an educational organisation which distinguishes it from self study,
- the use of a computer network to present educational content, and

- the provision of two-way communication via a computer network.

Learners can study course materials from their computer screen and have no need to be physically present in a class room; no time and location restrictions apply to limit the learning experience.

2.4 Mobile Learning (m-learning)

e-learning depend much on the availability of wired network connection to get access to learning materials, this kind of services still considered as a limitation among learners who are constantly on the move. In order to overcome the continuing change of learner location, there is a great demand to make use of wireless networking technologies which can provide learning content at anytime and anywhere. Mellow [102] explains that m-learning is "a means to enhance the broader learning experience, [it] is not a primary method for delivering courses/distance learning".

The main difference between e-learning and m-learning is that the first takes place in front of a computer or in internet labs, while the second takes place at any location [123].

In [12], the author listed some of the benefits of m-learning:

- helps learners to improve their literacy,
- helps learners to identify the areas where they need assistance and support,
- helps to remove some of the formality from learning experience and engage reluctant learners, and
- helps learners to remain focused for long periods and helps to raise self-esteem and self-confidence.

The move from wired to wireless services has a big impact at learning activity and it is very evident that the use of mobile learning technologies and services has many potential implications on learning process such as *Accessibility*, *Context*, *Collaboration* and *Appeal*.

- *Accessibility*: Providing mobile learning services in an educational environment can help learners access their learning materials very easily regardless the location and time. This advantage is not limited to location and time but also the devices required to access mobile networks are relatively inexpensive compared to laptop or desktop computers [13].

- *Context*: Dey, Abowd and Salber [45] define context as "any information that can be used to characterize the situation of an entity. An entity is a person, place, or object that is considered relevant for the interaction between a user and an application, including the user and the application themselves". The awareness of learning context is important. A learning system should adapt the learning process in response to context change.

- *Collaboration*: m-learning also allows the collaboration to happen among learners regardless of their location because of the accessibility of mobile networks services. Learners can discuss some learning tasks by interacting with other learners using mobile device.

- *Appeal*: Taking control over learning activities using mobile devices sounds appealing to learners, the main interest here is not the device but in the ability to learn "anytime" and "anywhere". Savill-Smith found a correlation between the use of palmtop computers and the increase in students' motivation [120].

New advanced mobile telephones are capable of exchanging voice, text, pictures and video, facilitated by wireless networks that provide high-speed connections with low

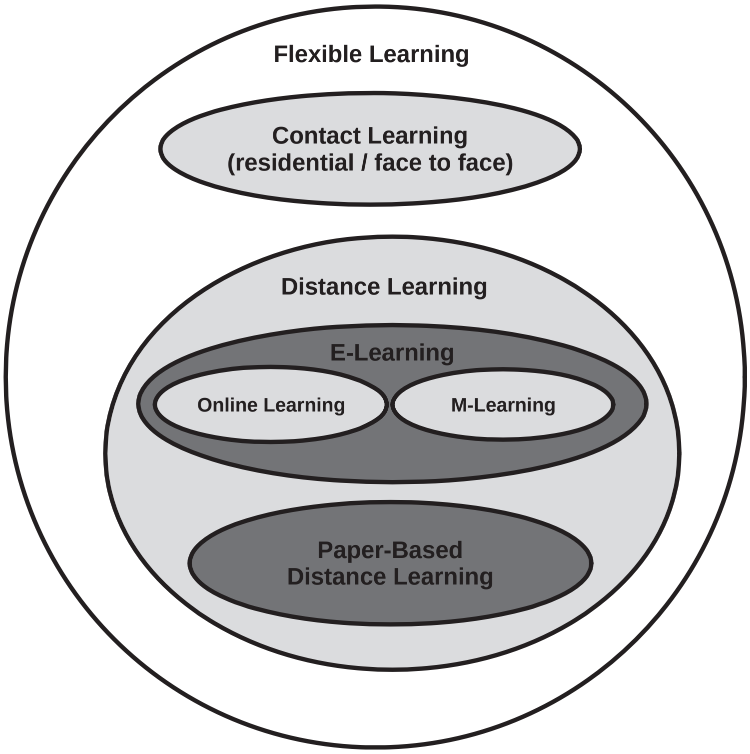

Figure 2.1: The Subsets of Flexible Learning [28].

costs to mobile users. In the presence of wireless technology, mobile learning can provide an alternative solution to the limitations of e-learning (see Figure 2.1).

Mobile learning is the ability to deliver learning content anytime, anywhere through the use of mobile devices. Mobile learning is also referred to as the intersection of mobile computing and e-learning to produce an anytime, anywhere learning experience. Mobile learning (m-learning) is defined as "any sort of learning that happens when the learner is not at a fixed, predetermined location, or learning that happens when the learner takes advantage of learning opportunities offered by mobile technologies" [144]. Lehner [90], explained that m-learning refer to "any service or facility that supplies a learner with general electronic information and educational content that aids in the acquisition of knowledge regardless of location and time".

The concept of mobile learning is defined under four categories [124] : *Technocentric*, *Relationship to e-learning*, *Augmenting Formal Education*, and *Learner-centred*.

- *Technocentric*: In this perspective, m-learning is defined as learning that takes place via such wireless devices as mobile phones, personal digital assistants (PDAs), or laptop computers.

- *Relationship to e-learning*: In this perspective, m-learning is defined as subset or extension of e-learning.

- *Augmenting Formal Education*: In this perspective, m-learning is defined as a form of distance education; m-learning is not only in classrooms, but also in all forms of traditional learning.

- *Learner-centred*: In this perspective, m-learning is defined as the focuses on the mobility of the learner rather than the device, any sort of learning that happens when the learner is not a fixed, predetermined location [109].

In [40], the author listed some unique characteristics of m-learning that differentiate m-learning from other learning forms:

- *Urgency of Learning Need*: The use of mobile for an urgent matter of learning

- *Initiative of Knowledge Acquisition*: The learner can access the learning content on the move and demand using wireless network.

- *Mobility of Learning*: The "anywhere" and "anytime" learning due to the large coverage of the wireless networks.

- *Interactivity of the Learning Process*: Learner can easily communicate with his tutors, classmates, or other materials using the mobile devices.

- *Situating of Instructional Activity*: The learning activities can be adapted to fit the learner context.

- *Integration of Instructional Content*: Integrating many information resources to fit the learner context.

Researchers have suggested that m-learning can enhance the learning experience through collaborative learning; however, there are many obstacles to successfully implementing any significant m-learning applications. The e-learning concept is well designed, implemented and tested and there are many standard applications in use today. To transform e-learning application success to m-learning will not be completely applicable due to many limitations such as device and network. To tackle this issue, there is a great need to understand the way in which learner uses the mobile phone and make use of that to design an effective m-learning application.

Some of the major obstacles related to the use of m-learning are limited interface, memory, battery life and storage, different kind of mobile connectivity, cross-platform, the lack of integration of some mobile application to mobile technology environment and no standardisation in term of mobile technology development.

There are many similarities between m-learning and e-learning such as:

- Both are concerned with on-line learning content.
- Learners, authors, the administrator and tutors are the main participants in both m-learning and e-learning environments.
- Both are trying to provide a personalised learning experience for learners.
- A learning coordination tool is required in order to allow interaction between participants.

While there are similarities in relation to the technological environments of m-learning and e-learning, significant differences also exist. Specifically, m-learning carries a range of prerequisites related to devices, connectivity limitations, service providers and mobile content limitations.

Nowadays, there is a wide range of mobile phone devices available in the market like PDA, Smart Phones and multimedia players. All of these devices have some common constraints, such as limited input capabilities, small screen, limited memory and limited battery life.

The bandwidth offered by the mobile networks providers is low for an extensive use of internet and expensive at the same time compared to fixed line internet services. Each mobile network has different communication networks around the world (e.g. GSM, GPRS, 3G and CDMA).

Service provider is a vital part of any mobile phone due to its restrictions on the way the users can use their phones. For example, user can't access other network if the phone is subscribed to a specific network provider. Moreover, mobile phone manufactures are very strict in term of security policies. For example, some phone manufactures dose not allows third party software to run without approval.

A m-learning application needs to consider the above mentioned limitations while preparing the learning content. Creating unified learning content to fits all the above limitations is a very challenging task. Therefore the learning content author has to consider the target devices and their supported presentation programs to automatically adapt learning content based on the learner device. The learning content can be found in more forms like images and multimedia which are not normally supported by the mobile devices. The learning content design will be discussed in more details in Chapter 4.

The m-learning system need to maintain the learning content so that learners can access learning content per request over the Internet. In addition, the structure of the learning content should enable learners to enhance their learning skills and experience which depends on many factors like learners preference, behaviour and how learning content is structured, built and processed.

Objectives of an m-learning application are to have a broad "on the move" and easy access to learning content, provide an interactive and engaged learner experience. So that learning system must assemble learning content from scratch and deliver learning content to meet the learner's needs.

Mobile learning is the main focus of this thesis which consist of five main aspects that will be explored and discussed in the following chapters of this thesis:

- Learning Technology (Section 4.3.3),
- Learning Context (Chapter 3 and Section 4.3.1),
- Learning Content (Chapter 4),
- Learner Modelling (Chapter 5), and
- Adaptation (Chapter 6 and 7).

2.5 Learner Profile and Learner Model

The concepts of learner profile and learner model are often used as synonyms, and to be able to clearly differentiate between the two concepts the following definitions is dedicated to describe the difference.

Koch [85] illustrates a learner profile as a collection of personal information; this information is saved without adding any further description to the original information. Learner profile represents cognitive skills, intentions, learning styles, preferences and interactions with the system. These values may be constant or changeable over time according the learner's context. Learner profile represents the store which all information about the learner will be saved in, learner profile can be used to retrieve the needed information to create learner model.

Koch [85] describes a learner model as the representation of the system's beliefs about the user in the real world. The learner model is based on the information about the real learner. A learner model is the presentation of a mental status (such as knowledge, preference, background, and experience) related to a context in the real world [157]. The learner model stores the specific information on each individual, which enable the system to identify different users [35].

The main goals of learner modelling are [69]:

- To assist the user in locating information;
- To customise the information presented to the user;
- To modify an interface according to the user;
- To select appropriate instructional exercises or interventions;
- To provide feedback to the user on the level of their knowledge;
- To reinforce collaboration; and
- To predict future behaviour of users.

The entire learner model from building and updating process is called learner modelling. The main task for building learner model is to help the system predicting future behaviour of learners, there are many numerous user models which automatically identify user patterns [156]; [25]; [119]; [17]; [138].

Kofod-Petersen propose a model to represent the context and divided it into following categories [86]:

- *Environmental Context* (services, people, and information accessed by the user),
- *Personal Context* (mood, expertise and disabilities),

- *Social Context*,
- *Task Context* (users goals, tasks and activities), and
- *Spatio-temporal Context* (time and location).

In order to acquire, manage, analyse and predict profiles of the learner, m-learning applications need to collect general data such as observable information content, selective actions, and/or the temporal behaviour. This can be done using an *Overlay Model*, *Stereotype Model*, *Perturbation Model*, *Fuzzy Logic* or *Machine learning techniques*.

- An *Overlay Model*: The main idea is to consider the learners knowledge as a subset of the domain model. Domain model is consisting of sets of knowledge elements that represent expertises knowledge. Overlay models are based on correlation between the student's knowledge and the expert's knowledge (see Figure 2.2). Using a comparison between the learners knowledge and the experts knowledge, the system is able to derive the learners lack of knowledge. One of the main negative aspects of this approach is that the users misconceptions of knowledge cannot be modelled, which is an important aspect within learning environments [84] [20]. In [30], the author presented more advanced overlay models which indicate the learner knowledge status by grades (an integer or probability measure).
- *Stereotype Modelling*: The first researcher to introduce the idea of stereotypes of learners is Rich [115]. Every learner is classified into a certain class, and each class is assumed to have common characteristics. The stereotypes provide a detailed description about the learner which enables the system to classify users based on personality and preferences. People frequently make assumptions based on simple observations. In constructing a stereotype model, similar to daily life

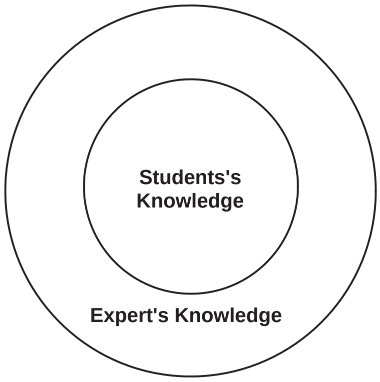

Figure 2.2: Overlay Model [20].

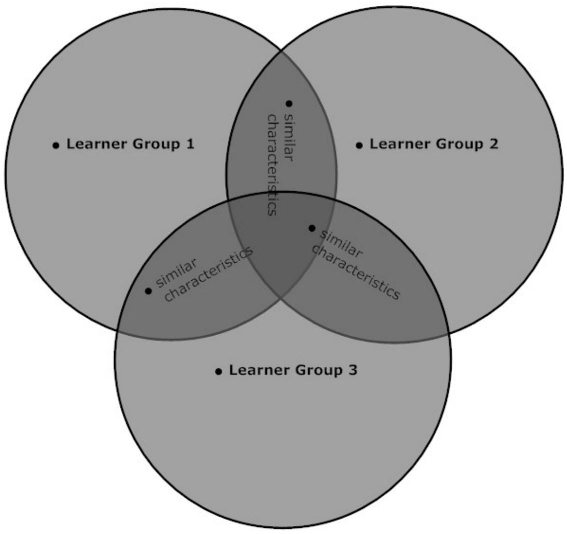

Figure 2.3: Stereotype Model [115].

assumptions, groups are formed based on certain assumptions as shown in Figure 2.3 [115].

- One of the overlay model disadvantage is the user misconceptions can not be modelled. For that reason, overlay models have been expanded to model and represent the learner misconceptions. Beck, Stern and Haugsjaa have constructed the *Perturbation Model* [20]. The perturbation model dose not perceives students knowledge as a simplification of expert knowledge, but more like perturbations over the expert knowledge [153]. The perturbation model may contain users'

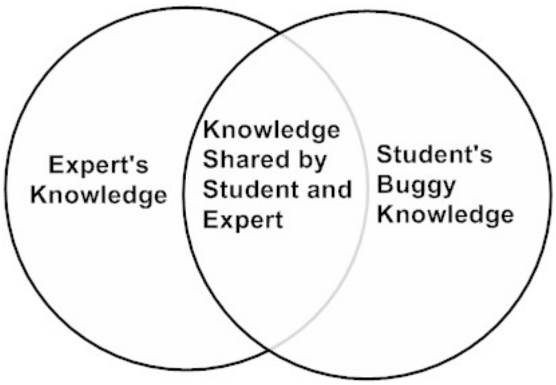

Figure 2.4: Perturbation Model [20].

errors or bugs that encounter the users (see Figure 2.4). Perturbation model used to better evaluate student errors and transform students' false information [20].

- *Fuzzy Logic* was introduced by Zadeh [164] as a new means to solve real world control problems by manipulating data that was not precise. This theory is a result of imprecise information in the real world, and one of humans abilities is to effectively process imprecise and "'fuzzy" information. Fuzzy Logic used in the expert systems and in the artificial intelligence application. Fuzzy logic allows for more qualitative definition and mimic human decision-making ([54].

- *Machine Learning Techniques* are commonly used for learner modelling because of the complex nature of relationships between learner contexts that are hard to be represented. The main idea of using machine learning is extracting meaningful information out of large data using numeric and statistical methods [147]. Machine learning directly observes user's interaction with the system and let the systems to make decisions based on these observation [11].

Numerous projects attempting to implement learner modelling in very specific contexts have been undertaken in recent years. These projects have had varied aims and levels of adaptation for individual users. There have also been many projects concerned

primarly with context awareness. A sample of such projects is discussed below.

- *The Active Badge Project* [[44], [150]] was one of the first indoor positioning systems. The system required users to wear badges that emitted signals to sensors that determined the location of the user. The *Active Bat* is an extension of the Active Badge, which is a 3D indoor position system. Small units called Bats are attached to objects (user, desk, chair and computer) in the space [4].

- In [73], Kassinen investigates the use of location to predict the context in well defined application such as route planning, guidance and location-based information. Location-aware messaging can add a new dimension to the way people communicate with each other, creating a new form of interaction.

- In [23], the authors developed a system called *LAMMS*. The system main idea is allow friends and strangers to communicate by sending and receiving SMS (short messaging service) to each other which could be tied to locations.

- *CAMS* is a Context based dynamic system developed [105]. CAMS use a combination of schedule and location information of SMS, voice and e-mail recipients to redirect communication to an appropriate device or address. The system requires users to register rules about which means of communication are appropriate in certain locations or at certain times.

- In [107], the authors developed a system called *LATTE*. *LATTE* is a location and time sensitive e-mail system, where e-mails are used to include location and time, to determine the appropriate recipients for messages. The main goal of this system is to filter the incoming messages so that their content is relevant to the users' current context.

- *Gate Reminder* is a context aware reminder, the main idea of this application is when the user is leaving their house as he/she approaching the front door,

he/she will be reminded by either SMS or a list of missing items that the user has registered them before [79].

- *ActiveCampus* Explorer is an application using location-based service; the application is a PDA based context aware application. The information provided to the user is adjusted based on the user's location [59].

- *ComMotion* is a context-aware system developed by MIT Media Laboratory [95] in 2000. The system utilises the location or GPS technology, and reminds the user of a to-do-list for selected locations by SMS or voice recording. The hardware of *ComMotion* includes a portable PC, a GPS receiver, a Cellular Digital Packet Data (CDPD) modem, and a Jabra earphone.

- The *Assisted Cognition Project* [47] uses ubiquitous computing environments to enhance human capabilities with cognitive limitations. The project uses various sensors, handheld and wireless devices to assist patient behaviour.

- The *MyPlace* Project [36] explores techniques for developing models of users, sensors and locations using accretion user modelling techniques and developing various applications to use these models.

2.6 Adaptation and Personalisation

According to Merrian Webster Dictionary [1], the verb "to adapt" means "to make fit for, or change to suit a new purpose", and according to the Oxford Dictionary [2], the term "adaptive" is defined as: "adaptive adj.: (technical) concerned with changing; able to change when necessary in order to deal with different situations". So adaptive is the ability to change, to suit different conditions. These definitions imply to the changes to meet specific requirements. In [1], adaptation is defined as "the act or

process of adapting or fitting" and "the state of being adapted or fitted".

Opperman [112] defines the system as adaptable in two ways:

- Adaptable system is a system which changes its parameters according to the user's needs. In other words, the user is able to modify the system in specified ways to fit his needs.

- An adaptive system refers to the automatic tailoring of the system, and its assumptions, to the needs of the user. The system changes its behaviour according to users' needs.

According to [155], the term personalisation represents both types of system (adaptivity and adaptability) as synonyms. Personalisation can be provided by tailoring the content or the visualisation of the system to the user's preferences.

According to [80], there are at least two distinct definitions of the term personalisation:

- Firstly, gathering and delivering the relevant information to individuals.

- Secondly, besides delivering relevant information to individuals, also providing recommendation to individuals based on the gathered information.

Considering the topic of m-learning it is necessary to deliver relevant information for the learner. Here, relevant information is the learning content.

An adaptive system is a system that adapts itself to different needs or circumstances. The process of adaptation uses a learners preferences, which are combined with other properties to make adaptation decisions. Both the learner preferences and other properties are stored in a learner model. The learner model consists of information about the learner such as knowledge and goals. A learner model is the ability to tailor its reaction depending on the model of the learner [32].

In the context of m-learning, adaptive systems are more specialised and focus on the adaptation of delivering learning content and the presentation of this content. The main focuses of adaptive system is how to make use of the knowledge learned by the learner during his/her learning activity [100].

2.7 Machine Learning Techniques for Learner Modelling Projects

Machine learning techniques are commonly used for learner modelling because of the complex nature of relationships between learner contexts that are hard to be represented.

Webb, Pazzani and Billsus [152] listed four main issues related to the use of machine learning techniques for modelling purposes:

- the need for large data sets,
- the need for labelled data,
- concept drift, and
- computational complexity.

Most personalisation approaches depend on machine learning techniques that require a large amount of labelled data in order to provide proper results. The continuous changes in learners' interests and profiles are what is called concept drift.

Reasoning approaches try to draw assumptions about individual learners based on their interaction with the system. *Neural Networks*, *k-Nearest Neighbors*, *Bayesian Network*, *Genetic* and *Fuzzy Logic* are some of the techniques that have been used to model the learners' contexts and also *Hybrid Systems* which consist of combinations of different machine learning techniques.

Numerous projects attempting to implement machine learning techniques for learner modelling have been undertaken in recent years. These projects have had varied aims and levels of adaptation for individual users. A variety of machine learning techniques have been employed in learner modelling systems in order to handle the imprecise information provided by learners, a comprehensive review of machine learning techniques can be found in [68]. A sample of such projects is discussed below.

The *K-Nearest Neighbor (K-NN)* model was proposed by Tsiriga and Virvou [140] for the initialisation of learner models in Web-based educational applications (Web-based Passive Tutor). This system is concerned with initialising and updating a learner model with a combination of stereotypes. The proposed system was implemented with 117 learners belonging to a range of different stereotypes. The results indicate that the use of K-NN framework facilitates the building of enriched learner models for adaptation.

A *Bayesian Network* model was proposed by [43] to model and represent the knowledge and characteristics of a learner in a probabilistic way; the main goal of this model lies in the representation of accurate probabilistic dependencies.

A fuzzy learner model for evaluating students during learning activities has been proposed [61]. The method is based on imitating a human tutor inside the classroom. *Fuzzy Logic* has been proposed to capture the interaction between the tutor and learners during the class, and using the capabilities of Fuzzy Logic to deal with imprecise information. This results in more accurate responses from learners to enhance the learning environment.

In the 'Forces' system, a *Fuzzy Logic* algorithm is applied to detect the cognitive status during learning. The main goal of this system is to model a learner's knowledge and learning skills in Newtonian Dynamics [10].

A similar approach has been proposed by Katz In Sherlock II [116] and in the

MDF tutor [19]. The uncertainty in students' performance was managed using *Fuzzy Inference System* and a set of rules for their formulation and update.

A tutoring system (BSS1) based on fuzzy rules has been proposed by [151] to implement a *Fuzzy Logic* inference engine that can manage student's learning activity. Another example of using Fuzzy Logic in a tutoring system has been proposed by [89]; this system monitors each learner's cognitive ability.

In [103], the proposed *Bayesian* learner model is integrated with an adaptive testing algorithm. Results from testing with simulated learners indicated that this Bayesian network model produces a highly accurate estimations of the learners' cognitive states.

Neural Networks are among the most effective learning methods currently known [104] in learner modelling. Neural Networks have been proposed in the literature mainly due to their ability to learn from noisy or incomplete patterns of users' behaviours and then generate knowledge to recognise unknown sequences [39].

A *Bayesian Network* model to support educational tutors in making decisions under uncertain conditions was proposed by Xenos [158]. The system was implemented with 800 learners who were studying an informatics course. The system was designed to model learner behaviour in order to make predictions about the success of tutors' decision making. Xenos determined that the proposed system can be a valuable tool in decision-making under conditions of uncertainty.

A computer aided language system based on *Neural Networks* has been proposed by Yeh and Lo [163]. The learner model accepts the learner's behaviour as the input using a multi-layer feed forward Neural Network. The proposed system was implemented with 46 college students studying a freshman English course. The study indicated that fast execution of Neural Networks makes it possible to assess a learner's meta-cognitive level, with promising results in the development of adaptive educational systems.

A *Bayesian Network* model for detecting the learning styles defined by Felder

and Silverman [51] was proposed by Garcia et al [56]. The proposed system was implemented with over 27 computer science engineering learners who were enrolled in an artificial intelligence course. The results of this approach was compared with the Index of Learning Styles questionnaire, proving that Bayesian networks are effective in predicting the learning styles of the learners with high accuracy.

In [146], a *Feed Forward Neural Network* model has been used to identify learners' learning styles, as defined by Felder and Silverman [51]. An artificial data set was generated for experimentation by simulating the actions of learners.

An intelligent tutoring system, proposed by Curilem et al. [118], used *Neural Networks* to model the learner preferences to be used in an adaptive interface mechanism. The main focus of the application is interface configuration.

Neural Networks and other Artificial Intelligence (AI) methods have been criticised for their inability to support learning interactions because they only allow for implicit learning. However, several attempts have been made to incorporate the powerful learning abilities of Neural Networks into existing student modelling systems, taking advantage of other AI methods capabilities.

An intelligent learning diagnosis system that supports Web-based learning was proposed by Huang et al [66]. The system processes a learner's profile and guides the learner in improving study behaviour, as well as helping the tutors with grading online class participation. *Vector Machines*, *Nave Bayesian* and *K-nearest Neighbor* algorithms process the data in the learner profile database to update the learner assessment database. The fuzzy expert system works on the learner profile to update both the learner assessment database and learner diagnosis database. The experimental results indicate that the proposed learning diagnosis system can efficiently help learners on theme-based learning model.

A *Neuro-Fuzzy* learner model system is proposed by Stathacopoulou et al. [128]

to diagnose the errors of high-school learners by collecting data using simulation tools related to a course, namely vectors in physics and mathematics. The system is tested with simulated learner data with different knowledge level categories. The behaviours of these learners corresponds to fuzzy values. A feed-forward Neural Networks model was also trained for error classification purposes.

Most of the projects referred to above show that machine learning techniques offer a set of powerful techniques either for learner modelling or supporting decision making tasks.

2.8 Summary

This chapter introduces the concept of d-learning, e-learning and m-learning services which include the differences between m-learning and e-learning. In addition, classification of mobile and wireless technologies is described. To be able to respond to the learner actions, a system needs to have complete details about the learner's preferences. To provide the system with this information, a user profile is created to gather this information.

To adapt to the learner needs, more reasoning is needed to answer all questions related to the learner, all data stored within a learner profile is not enough. A learner model stores enriched information about the learner, for example, a learner model provides information such as learning styles and history of interactions with the system. This allows adaptive m-learning systems to adapt the learning content to a specific learner. Every personalised system including adaptive m-learning systems base their adaptation processes on learner models. In Chapter 3, 4 and 5, adaptation components and learner models are depicted in more details.

Chapter 3

Adaptive Mobile Learning Framework

3.1 Outline

Advances in wireless technology and hand-held devices have created significant interest in mobile learning (m-learning) in recent years. Students nowadays are able to learn anywhere and at anytime. Mobile learning environments must also cater for different user preferences and various devices with limited capability, where not all of the information is relevant or critical to each learning environment. To address this issue, this chapter presents a framework that depicts the process of adapting learning content to satisfy individual learner characteristics by taking into consideration each learner's learning context. A machine learning based algorithm is used for acquiring, representing, storing, reasoning and updating each obtained learner profile. The main objective of this framework is to provide personalisation and tackle adaptation using a machine learning technique according to obtained learner profiles. These learner profiles contain learners' preferences, knowledge, goals, plans, place and possibly other relevant aspects that are used to provide personalised adaptations. Section 3.2 of this chapter discusses related works. Section 3.3 presents the framework foundation. Sections 3.4 - 3.6 presents the structure and implementation of the proposed framework.

3.2 Introduction

Electronic learning (e-learning) continues to grow rapidly but most e-learning technology involves wired infrastructures. It is believed that the emerging wireless and mobile networks will provide new applications in mobile learning [57]. With the rapid evolution of mobile devices such as PDAs, Table PCs and smart phones, pervasive (or ubiquitous) systems are becoming increasingly popular.

Mobile learning (m-learning) is "any sort of learning that happens when the learner is not at a fixed, predetermined location, or learning that happens when the learner

takes advantage of learning opportunities offered by mobile technologies" [144]. Given the rapid use of mobile technologies for facilitating the learning process anywhere and anytime, learners are able to use idle time, for example, when waiting for public transport, in between lectures, and traveling to and from university. This time can therefore be used more efficiently in terms of learning [98].

The awareness of learning context is important. A learning system should adapt the learning process in response to context change. The main goal for context-aware mobile learning applications is to sense the mobile learner's situation (environment) and respond to it [37]. Shilit [122] divided context into three categories: computing context, learner context, and physical context. Chen and Kotz [38] extended this list by adding a time context. The study in [45] identified four categories: identity, location, status, and time. Context has four dimensions [160]: situation, network, device, and expertise.

Most current learning content were designed for use with desktop computers and high-speed network connections. They usually contain rich media data such as image, audio and video. Learning content may not be suitable for presentation on devices with limited capability and limited network bandwidth. Moreover, the widespread problem in e-learning environments is that they cannot offer personalisation for the learner and that they can only present identical content to all learners. Mobile based education is already reaching a large number of learners and it offers a valuable advantage over traditional teaching with the possibility to adapt to individual learners, which is hard to achieve in the common teaching process.

It is possible for learning activities to occur everywhere: educational institutes, within homes, on buses and trains and in parks and restaurants. Unpredicted weather conditions may affect the learner's ability to accomplish a learning task [34]. Mobile learning is still in its infancy and most of the research projects are focusing on the

connectivity problem of using wireless networks or the problem of accessing course content using mobile devices [15].

Numerous projects attempting to use context to change the behaviour of an application. These projects have had varied aims and levels of adaptation for different individual users. A sample of such projects is discussed below.

Martin [97] designed a system for recommending activities for learners; this process is dependent on the learner's personal attributes, actions and the current context (location, time, available devices). The system can be used individually or collaboratively.

Ogata and Yano [110] designed a computer supported ubiquitous learning environment for language learning. This learning environment composed of two systems, the first system is a context-aware language learning support system for learning Japanese polite expressions and the second system is TANGO which can detects the objects around the learner using RFID tags.

The MOBIlearn project is an interactive model in which data is collected from sensors, and translated to appropriate services. An adaptive learner interface system has also been developed within this project [130].

In [40], a mobile scaffolding-aid-based bird-watching learning system, an outdoor learning system is proposed, meaning that a learner with higher learning efficiency will gain less support from the system.

In [78], Ketamo have implemented an m-learning environment (xTask) that adapts to different user devices (PC, PDA and WAP devices). xTask also implements a library for managing learning objects in different formats.

Few of the m-learning researchers have tackled the problems of adaptation of learning tasks and personalisation of course content based on students' models, learning styles and strategies [29]. These issues have been explored within the traditional Web-

based systems in numerous well-known projects:

- *ELM-ART* is an intelligent learning Web-site environment that supports example-based programming, intelligent analysis of problem solutions, and advanced testing and debugging facilities. ELM-ART provides all learning material online in the form of an adaptive interactive textbook. ELM-ART provides adaptive navigation support, course sequencing, individualised diagnosis of student solutions, and example-based problem-solving support [154].

- *InterBook* [31] is a tool for authoring and delivering adaptive electronic textbooks. Museum tour guide is another research [42] used mobile devices personal museum expert which is mainly concerned with user location.

- The *ENLACE* project [145], referred to as SEO/Birdlife, is a research project implemented in conjunction with teachers in a secondary school.

3.3 Framework Foundation

Learner adaptive systems aim to adapt learning content, location and presentation to each individual learner's characteristics or behaviour in order to improve the interaction between learners and the system (see Figure 3.1).

In this respect, modelling learner behaviour is a fundamental mechanism for providing personalisation [58]. Depending on the application, learner profiles are generated to store information about learner's preferences, interests, goals, usage data and interactive behaviour. Learner's preference is an important concept to predict learner behaviours and make appropriate adaptation actions. Preferences can be explicitly supplied by the learner [82].

In user-adaptive systems, the user model can be used for various tasks depending on the application [54], for example, to predict user's needs on the basis of past user

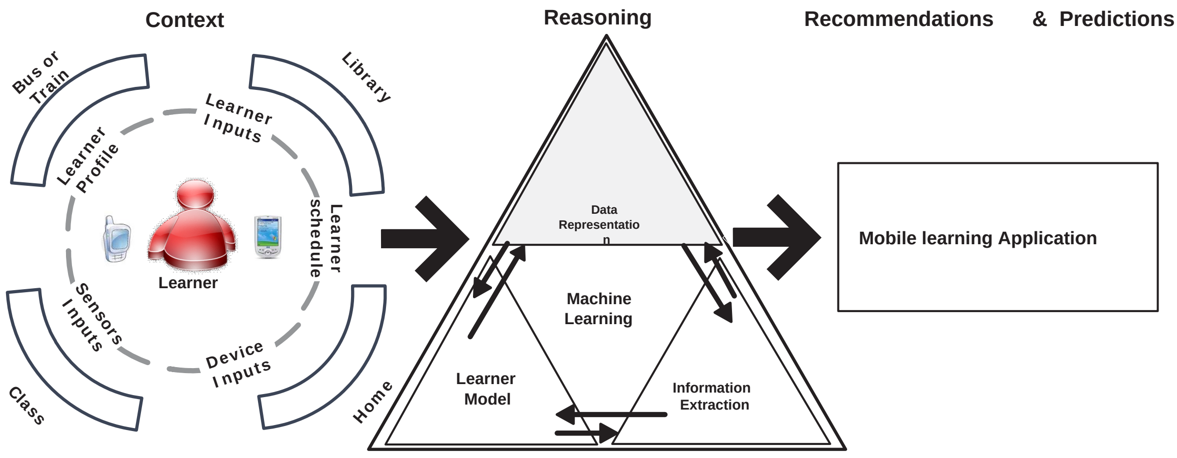

Figure 3.1: Basic Elements of a User Adaptive System.

behaviour, and to recommend interesting elements to a user based on preferences or previous user behaviour.

A personalisation engine as shown in Figure 3.1 is usually employed to infer adaptation actions on the basis of identified learner characteristics, for retrieving or filtering appropriate content and adapting the content presentation, and to match the navigation support and the interface attributes to the learner's needs. The framework presented in this work is principally designed to achieve content adaptation and personalisation based on individual learners, taking into consideration specific learning styles and subject matter learning motivation [48].

The concept of being context-aware is composed of two elements: *Personalisation* related to the learner, and *Automatic Customisation* related to the adaptation process.

- *Personalisation* for the learner means that the system knows about the learner and changes its appearance or behaviour according to learner needs.

- *Automatic Customisation* or adaptation means that the system creates a model for the learner in an automatic (machine learning) way to suits the learner's situation and needs.

It is believed that most people prefer some particular method of interacting with

information. These methods of interaction are referred to as learning styles. A learning style is the method of learning particular to an individual that is presumed to allow that individual to learn best [48]. The framework foundation is based on storing and exploiting information about the learner. However, learners differ in traits such as skills, aptitudes and preferences for processing information, constructing meaning from information, and applying it to real-world situations. Therefore, modelling each learner is a fundamental mechanism for providing personalisation [83].

It is commonly noted that interactive systems are becoming more complex. As a result, intelligent, friendly learner interfaces and adaptive systems are needed to improve learner interaction with these systems. Also, the exponential growth of the Internet for mobile users makes it difficult for these learners to cope with the huge amount of available information. The challenge that information providers and system engineers face is the creation of adaptive applications.

A learner adaptive system uses the knowledge given by learner model to implement the following tasks [53]:

- *Recommendation*: the capability of suggesting interesting scenarios to a learner based on some information; and
- *Classification*: building a learner model that classifies related data into one of several predefined classes.

3.4 Layer Components in the Framework

Within the framework, we first consider the application that supplies the events that have occurred in a specific context. In case of mobile learning adaptation, several dimensions of adaptation need to be considered such as [58]:

- the content dimension,

- the learner model dimension,
- the device dimension,
- the connectivity dimension, and
- the coordination dimension.

Within these dimensions, there exist sub dimensions.

In order to provide personalisation for individual learners, the system must first be able to identify the learner in order to collect the information required to perform the personalisation (Log-in). This provides better accuracy and consistency. Learner data is gathered to construct the learner profile. The data collected includes (see Figure 3.2):

- data regarding the interaction between the learner and system;
- personal information;
- data regarding the environment of the learner when interacting with system; and
- direct feedback given by the learner.

3.4.1 Context Acquisition Layer

The context acquisition layer is used to gather the information required for adaptation. It relies on both *Implicit* and *Explicit* information collection.

- *Implicit Information* is gathered by monitoring the learner's interactions with a system and making assumptions as to their motivations and needs. Typical methods for gathering implicit information include determining learner position using GPS, and sensing noise or time using a built-in microphone or clock.

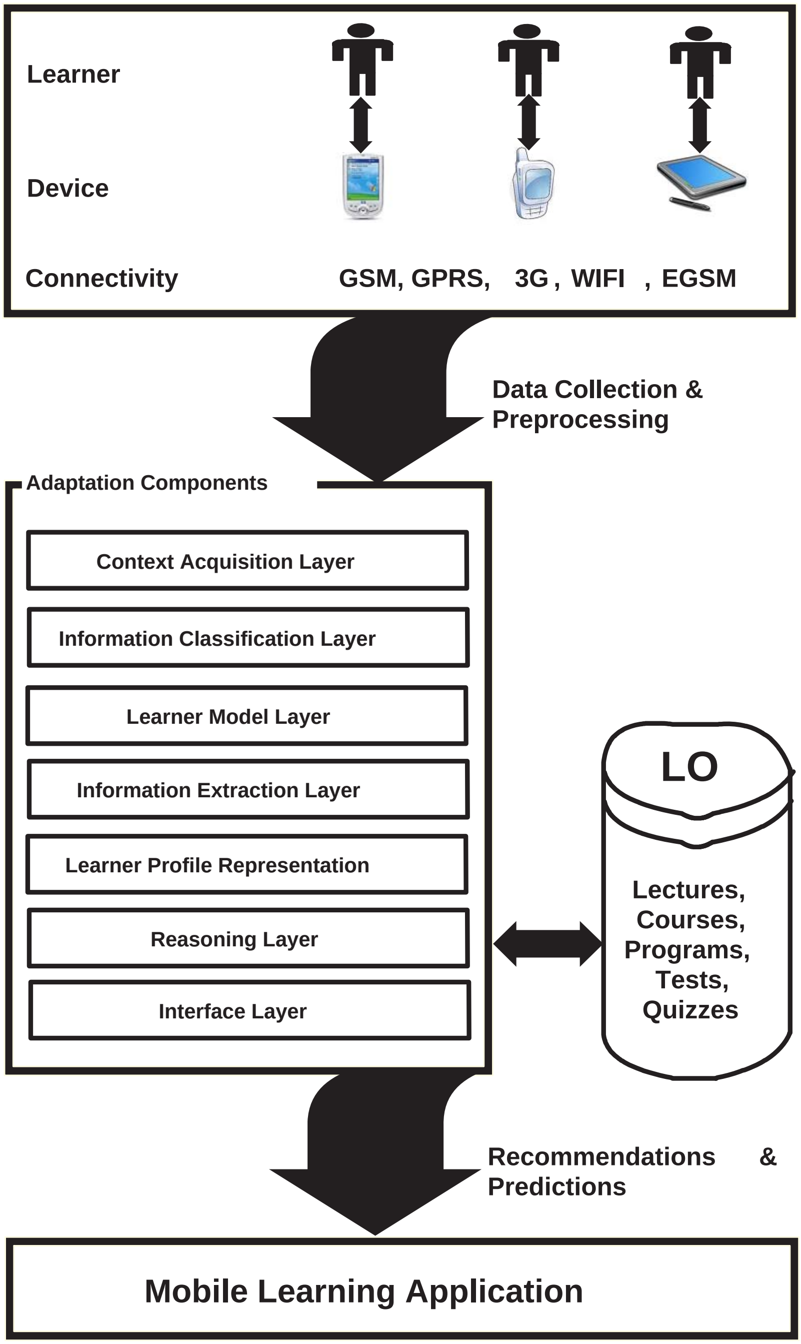

Figure 3.2: Adaptation Components.

- *Explicit Information* is information provided by the learner, usually through the use of forms with text boxes and check boxes. It often contains demographic information such as birth date, interests, marital status, job and personal characteristics. Explicit information forms will be completed only once by each learner: when a learner uses a system for the first time. Any such form should not require significant extra thinking, typing or memory retrieval. Requests for such information cost learners time, require willingness to participate and require that the system assumes that all the information provided by the learner is correct and valid. As well as the system relying on learners to attempt to supply correct information, it is possible that different learners may interpret the questions differently. To avoid this kind of problem, the questions asked should be straight-forward for learners to process.

A learner will not notice the importance of supplying this information unless he uses the adaptation system frequently and is aware of the difference in time saving and accuracy of results.

3.4.2 Information Classification Layer

This layer deals with all data obtained from the previous layer of the framework by categorising the data into several class types as shown in Figure 3.3. This layer consists of two categories:

- Personal context - all relevant attributes to the learner throughout his/her use of the system; and

- Shared Context - attributes relevant to all learners when using the system.

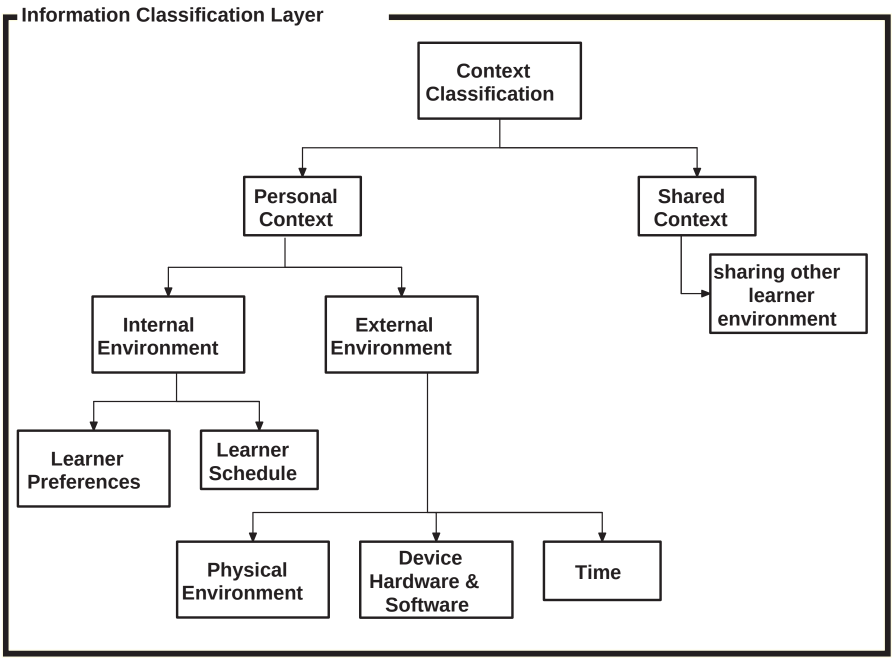

Figure 3.3: Information Classification Layer Structure.

3.4.3 Learner Model Layer

Learner modelling aims to make information systems learner-friendly by adapting the behaviour of the system to the needs of the individual, as shown in Figure 3.4. A learner model should capture the behaviour (preferences, location, interesting topics, etc.) of a learner when interacting with the system.

A learner model is defined as a set of information structures designed to represent one or more of the following elements [82]: goals, plans and preferences; representation of relevant common characteristics of learners stereotypes; classification of learner stereotypes; learner behaviour; assumptions about the learner based on the interaction history; and/or interaction histories of many learners into groups. Learner model structure details are presented and discussed in more details in Chapter 5.

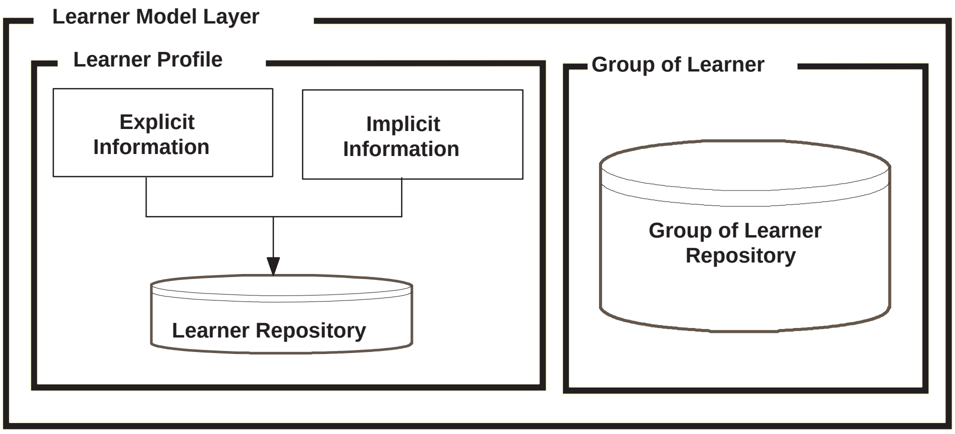

Figure 3.4: Learner Model Layer Structure.

3.4.4 Information Extraction Layer

The process of adaptation is based on storing and exploiting information about the learner. However, learners differ in traits such as skills, aptitudes and preferences. In a mobile learning environment, differences in learner preferences, types and amounts of content are relevant and critical to the learning process. This layer assesses, analyses, verifies and filters the data based on the current learner situation.

3.4.5 Learner Profile Representation Layer

In order to achieve personalised services, we must be able to specify learner interests. This can be done using a machine learning algorithm, which takes a learner's information for inputs, then compare and analyse the learner's need, interest and environment. Such profile can be further enriched with more specific information such as location and time. However, in this context, the focus is on learner's interests and characteristics. Enriched learner profile enable the system to select between a number of topics and interests to match with the best learning style for that user. Creating learner profile allows for increased accuracy of results, given a sufficiently expressive keyword.

A learner can indicate his interest in a specific domain or even single concept by specifying a value in a predefined range. Precise information allows the system to more accurately support learner decision-making. Learner Profile Representation is also discussed in more details in Section 5.5 and Section 5.6.

3.4.6 Reasoning Layer

Machine learning techniques have been applied to learner modelling problems for obtaining models of individual learners and grouping them into categories. This process is very important to inform an accurate and useful system that can modify its behaviour over time. Machine learning techniques are applied to the data obtained in the earlier layers, in order to capture learner behaviour patterns.

The output of this layer is a set of structural descriptions of what has been learned about learner behaviour and interests. Any machine learning techniques based on adaptation should consider the following conditions to provide a wide range of possibilities on m-learning [110]:

- the amount of effort required to provide the system with necessary background knowledge,
- the amount of time (computational time) required,
- the amount of input data required to be able to make useful decisions, and
- the appropriate handling of noise, uncertainty, and validity.

The detailed structure of the reasoning engine will be presented in Section 5.6 and Chapters 6 and 7.

3.4.7 Interface Layer

The interface layer is formed by the events that are processed by the adaptation system as well as the questions about the learner that it can answer. Furthermore, the interface forms a description of the way the application interacts with the adaptation system. The learner model contains the information about the learner that has been collected in previous layers.

One of the main issues related to the interface layer is whether the adaptation system frequently changes the learner's interface each time the learner uses the system, or gives the learner the freedom to determine their preferred way of representing the information when they first use the system. These changes include the arrangement of icons and items on the learner's device screen.

The system developed in this research will employ a default interface layout with only limited available changes to the interface, for example font colour and size. By using a default interface layout with limited changes to the device screen, problems arising from frequent changing interfaces will be minimised. Such potential problems include:

- learners could be prevented from engaging in an automatic learning process with respect to some aspects of the interface and not focusing on the learning procedures;
- learners spend more time looking for particular interface elements because every time they use the system the interface will change;
- frequent changes in the interface may prevent learners from acquiring skill and speed in using the interface;
- unexpected changes may be generally confusing and distracting for the learner;
- learners may not be aware or experienced enough to change the interface; and

- learners may not want to spend time adapting to the interface.

As a result, a suitable default interface layout chosen by the system may be able to choose the appropriate adaptations more accurately than a learner could.

3.5 Learning Objects Repository

A learning objects repository is any written digital material source (see Learning Objects Repository in Figure 3.2). Separating the educational content into small segments allows ease of use of the content. Digital resources are usually described with additional metadata attached to them and later arranged into more meaningful content such as lectures, courses, programs, tests, videos, images and quizzes. Metadata repositories help in categorising and searching for learning content.

3.6 Mobile Learning Application

It is important to specify how much control or involvement the learner will have over the adaptation system, from the outset of system development. To make an informed decision, the implications of giving the learner full (or no) control over the system will be considered.

Possible options for determining the learners' level of control include:

- the system submitting all recommended adaptations to the learner, who is required to give their approval before implementation;
- implementation of automated adaptations with learner ability to undo the adaptations if they do not like them; and
- even allowing learners to disable the entire adaptations system. Learners expect benefits from using the adaptation system such as saving of time and effort.

Incorrect actions by the system may have serious consequences which will discourage the learner from continuing to use the system. After consideration of the implications of level of learner control, the system in the framework presented will maintain full control over the adaptations process, with only one option for the learner to disable the entire process. This decision is made for two reasons:

- to give the learner full control over the system assumes that the learner has a certain level of knowledge about the system (which not everyone has); and
- to decrease the computational time.

3.7 Summary

This chapter has presented a new framework that depicts the process of adapting learning content to satisfy individual learner characteristics by taking into consideration each learner's learning context. The adaptive framework is designed to overcome the limitations of using mobile learning to provide personalisation and tackle adaptation using a machine learning technique according to an obtained learner profile for mobile learning purposes. We have described the system architecture of the presented context adaptation based learner profile framework and learning style adaptation which is fundamentally grounded on a number of logical layers: *Context Acquisition*, *Information Classification*, *Learner Model*, *Information Extraction*, *Learner Profile Representation*, *Reasoning* and *Interface*. It is a generic framework for selecting the most appropriate learning content format for learners based on their learner preferences and contextual features. The ultimate goal of the framework is to provide a logical structure for the process of adapting learning content to satisfy individual learner's needs.

Chapter 4

Mobile Learning Content

4.1 Outline

Delivery options for mobile learning are increasing, however new technologies alone will not improve the experience of mobile learners. There are a number of factors that impact on a typical learning experience, and many more when that learning experience becomes 'mobile'. This Chapter presents a framework to describe the factors that play an important role in delivering learning content to mobile learners, and their relationship to each other. Once the necessary information is collected about a learner - either automatically (e.g. location, device, previous usage) or through learner input (e.g. age) - learning content can be adapted to meet the unique and personal needs of that learner within their current context.

This chapter presents an m-learning content framework that uses a new approach to m-learning content adaption. The main objective of this framework is to show that there are a number of factors that impact on a typical learning experience, and that the number of factors in a mobile learning experience. The framework describes the factors that play an important role in delivering learning content to mobile learners. It allows consideration of individual learning styles and scenarios, device and application capabilities, and learning content structure, leading to customisation of the type and delivery format of learning information in response to the user. Section 4.3 of this chapter provides an overview of key factors in the success of the m-learning application, and their relationship to each other. Section 4.4 presents the structure of the proposed m-learning content framework. Finally, section 4.5 provides comment about the impact of the learning content framework on the learning experience.

4.2 Introduction

The advancement of technologies in wireless and hand-held devices has created significant interest in mobile learning (m-learning) in recent years. New advanced phones are capable of exchanging voice, text, pictures and video. In addition, wireless networks provide high-speed connections with low costs to mobile users.

Mobile learning (m-learning) is "any sort of learning that happens when the learner is not at a fixed, predetermined location, or learning that happens when the learner takes advantage of learning opportunities offered by mobile technologies" [144]. Most current learning content was designed for use with desktop computers and high-speed network connections. They usually contain rich media data such as image, audio, and video. Learning content may not be suitable for presentation on devices with limited capability and limited network bandwidth. Moreover, the widespread problem in e-learning environments is that they cannot offer personalisation for the learner and that they can only present identical content to all the learners. m-learning is already reaching a large number of learners and it offers a valuable advantage over traditional teaching with the possibility to adapt to individual learners, which is hard to achieve in the common teaching process.

Learning can occur in all situations and environments, however until recently the role of technology in this learning has been limited in many settings. With the advances in m-learning, it is becoming possible for learning activities facilitated by technology to occur everywhere: for example, educational institutes, within homes, on buses and trains and in parks and restaurants. However, it must be noted that m-learning is still in its infancy and most current research projects are focusing on the connectivity problem of using wireless networks [15]. For example, unpredicted weather conditions may affect the learner's ability to accomplish a learning task [34].

Numerous projects attempting to implement m-learning in very specific contexts

have been undertaken in recent years. These projects have had varied aims and levels of adaptation for individual users. There have also been many projects concerned primarily with context awareness and its impact on content delivery. A sample of these projects is discussed below.

- In the Electronic Guidebook [65], mobile Web content was specifically created for the Exploratorium museum in San Francisco.
- Tampere University of Technology presented content in the form of a game. The electronic device was used to measure each student's knowledge level, and then adapted the speed of presenting new learning material [77].
- The Kidsroom project [24] at MIT Media laboratory provided a room that guides children through an interactive storytelling game. Materials were adapted according to the activity context.
- A mobile device is used along with a robot to educate visitors in a museum [108]. Spatial context and spatial interface are used to present different multimedia learning content at different demonstration places.
- Another work focussed on the challenges of developing multi-channel e-learning services on the Internet, in which the content is located at the same node [135]. Multi-channel is defined as the technology which is using a framework on how the content should be distributed such as Web, wap, phone and fax. Prototype tests have shown that older devices and browsers do not support content control standards; this creates problems in realising multi-channel services that use these content control methods. Content control can also assist content editors to improve the learning outcome and utilise the service to increase the user's learning process.

- A general architecture to support mobility in the learning scenario is presented, which considers m-learning from two points of view (technical and educational side) [137]. A Mobile Learning Management System (mLMS) sits on top of the usual Electronic Learning Management System (eLMS). Three main functionalities for a Mobile Learning Management System (mLMS) are considered - 'Context Discovery', 'Mobile Content Management and Adaptation' and 'Packaging and Synchronisation'. Interaction between these modules will allow the automatic selection of services which are properly constructed for devices' capabilities, user preferences and needs, and will permit the usage both online and offline.

- The MOBIlearn project is an interactive model in which data is collected from sensors and translated to appropriate services. An adaptive learner interface system has also been developed within this project [130].

- Ketamo have implemented an m-learning environment (xTask) that adapts to different user devices (PC, PDA and WAP devices) [78].

- In [121], authors have explored the possibilities existed for informal learning and all settings involved

- In [49], authors have developed a system to support informal mobile language learning. This type of "designed" informal learning takes into consideration the situations where mobile devices are used for learning and available device features and software for use to response to learners needs.

Despite research into both m-learning and adaptation, few of the m-learning researchers have tackled the problems of adaptation of learning content based on learner models, learning styles and strategies [29]. The main purpose of mobile learning applications is to deliver learning content based on each user's current situation. There

are proposed frameworks and systems that achieve this goal [7] [140] [158] [103] [143] [162] [96] [149] [129] [94], all of which have an initial focus on technology and use the learner's context, style, preferences and knowledge.

Most of the projects referred to above have been concerned with the impact of context in relation to mobile learning. They did not address the mobile learner content and the importance of learning content in terms of adaptation techniques. These concerns are addressed in this chapter.

4.3 Key Factors in m-learning Application Success

Particularly in mobile learning applications, contextual and personal characteristics have a strong relationship with each other, i.e. personal characteristics determine a human behaviour and the behaviour determines the context. A study of a user's behaviour provides clues to determine the context in which the user is interacting with the system. For example, knowing the style of display preferred and access speed provides information that can be used to indicate the type of device being used and Internet access availability. Figure 4.1 [6] presents the key factors for a system to evaluate in order to provide personalisation to mobile users. User adaptive systems aim to adapt learning content, location and presentation to each individual learner's characteristics or behaviour in order to improve the interaction between users and the system. The process is based on storing and exploiting information about the learner.

4.3.1 Context

A learning activity is not an isolated process that occurs in a vacuum; many factors play an important role in this process [27]. The learning context is defined as any information that can be used to characterise the situation of an entity in a learning

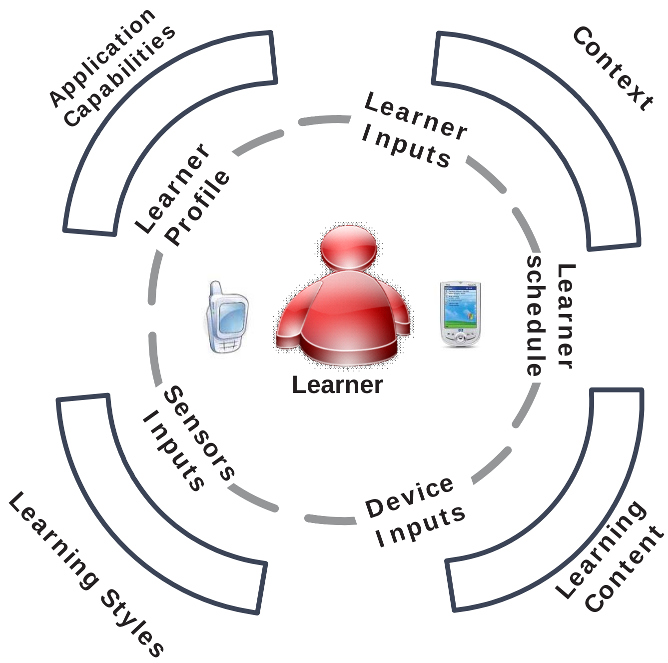

Figure 4.1: Key Factors in m-learning Application.

activity [46]. Dey, Abowd and Salber [45] define context as "any information that can be used to characterise the situation of an entity. An entity is a person, place, or object that is considered relevant for the interaction between a user and an application, including the user and the application themselves". Adaptive system developers can use context information in the following ways [94]:

- resolving references,
- tailoring lists of options,
- triggering automatic behaviour, and
- tagging information for retrieval.

Learning contexts are influenced by the learner's location, needs and environment. These learning contexts can constantly change over time for mobile learners which makes prediction based on learning context extremely difficult. Researchers have suggested many categories for context in a mobile learning environment.

In [149], the author proposed the following dimensions for all possible contexts that can occur in mobile learning:

- *Identity*: unique identifier of each learner.
- *Spatio-Temporal*: this includes the learner location and the time of learning activity.
- *Facility*: mobile device capabilities and limitations.
- *Activity*: the type of the appropriate set of activities that must be in the learning process.
- *Learner*: the psychological properties and characteristics of a learner which helps to describe the situation.
- *Community*: the use of social context of the learner.

In [133], the author proposed three classifications of context categories:

- *Environment*: describes the physical environment.
- *Participants*: describes the environment and their personal characteristics.
- *Activities*: includes the participant and the environmental activities.

The awareness of learning context is important. A learning system should adapt the learning process in response to context change. The main goal for context-aware m-learning applications is to sense the mobile learner's situation (environment) and respond to it [37].

Learner context inputs can contain various demographic properties, as well as the learners' skills and capabilities, interest and preferences. Also, the current situation can be used as a factor for adapting the service. The context acquisition techniques

used to gather the information required for adaptation rely on both *Explicit* and *Implicit* information collection (refer to Section 3.4.1).

Typical methods for gathering implicit information include determining user position using GPS, and sensing noise or time using a built-in microphone or clock. Extra parameters to the learner inputs, or a direct request to define a learner's current situation, may be required to successfully complete the adaptation process.

4.3.2 Learning Styles

It is believed that most people prefer some particular method of interacting with information. These methods of interaction are referred to as learning styles. Keefe [75] defined learning styles as characteristics which are "cognitive, affective and psychological behaviour that serve as relatively stable indicators of how learners perceive, interact with, and respond to the learning environment". A learning style is the method of learning particular to an individual that is presumed to allow that individual to learn best [48].

Each learner has his own learning style and preferences that can help him to improve and accelerate his learning process. In [51], Felder and Silverman learning style, the authors classify learning styles using eight dimensions: *Active*, *Reflective*, *Sensing*, *Intuitive*, *Visual*, *Verbal*, *Sequential* and *Global*.

- *Active*: the learner prefers to actively process the information in the learning content.

- *Reflective*: the learner has a preference to read and think about the learning content.

- *Sensing*: the learner prefers to read concrete materials for example facts, data.

- *Intuitive*: the learner prefers to read abstract material such as theories and concepts.
- *Visual*: the learner learns easily using pictures and images.
- *Verbal*: the learner prefers to learn by reading texts or listing to a speech.
- *Sequential (Bottom-up style)*: the learner learns better when he focuses on the narrow details first and brings them together to develop the large picture.
- *Global (Top-down style)*: the learner finds it easier to learn by having the overall picture and slowly getting into details.

For example, learning styles identified by Dunn and Dunn [48] place learners into categories such as: *Environmental*, *Emotional*, *Physical*, *Sociology* and *Personality*. Learner preferences may be auditory (audio lectures), visual (diagrams, graphs), linguistic (Word, PowerPoint), or a combination of these.

4.3.3 Application Capabilities

An m-learning application is a mobile software application that is used by the learner to accomplish learning activities, taking all contextual elements into consideration. The learning application capabilities for m-learning are concerned with:

- the different ways of interactions between learners,
- the amount of control over the adaptation process,
- different options that are given to the mobile learner in a specific learning activity,
- the amount of help and support provided to a learner when the learning activity begins,

- the privacy data (identification, learner profile) provided by the learning tool, and
- most importantly, the interface capabilities of the learning tool.

All of these capabilities will allow the content creators to tailor various alternatives of learning content to meet the learner and application requirements. Some significant technologies should be utilised to manage the learning experience for mobile learners:

- *Mobile Devices*, A different number of mobile devices are being used in many educational institutes. Mobile learning is impossible without the use of the mobile devices. These devices vary significantly in features, ability, size and price. The common features in all mobile devices are their mobility and ability to make wireless connections. The main types of mobile devices used in the education institutes are:
  - *Notebook Computers* or *Wireless Laptops*. These are the most commonly used computers in educational institutes, because the added feature of a wireless card enables data communications.
  - *Tablet PC*. A wireless, portable personal computer with a touch screen interface is typically smaller than a notebook computer but larger than a smart phone.
  - *Personal Digital Assistant (PDA)*. Combining the functionalities of computers, telephones, internet and network, they are small and have significant processor power.
  - *Cellular phones*, also called mobile, cellular telephone or cell phone. An electronic device used to make mobile telephone calls across a wide geographic area, it is mainly used for voice communication and the sending

  and receiving of text messages (SMS). Disadvantages include low memory capacity and low data transfer rate.

  - *Smart Phones.* Hybrid devices which combine the abilities of cellular phones and PDA, they are smaller than PDAs and bigger than cellular phones. They use Symbian, Windows Mobile or another operating system, have Internet browsers and support different multimedia format.

- *Wireless Technologies*, Wireless networks are used in most educational institutes. Mobile learning is impossible without the use of wireless networks. The main types of wireless technologies used in the education institutes are:

  - *Global system for Mobile Communications (GSM).* One of the leading digital cellular systems, it was originally a European standard for digital mobile telephony. One of the reasons for developing GSM was the need for high capacity network to provide better quality, more services and cheaper infrastructure. It uses narrow band Time Division Multiple Access (TDMA).

  - *Wireless Application Protocol (WAP).* A free, unlicensed protocol for wireless communications, it facilitates the creation of advanced communications services and access to Internet pages from a cellular phone.

  - *General Packet Radio Service (GPRS).* A 2.5G protocol, it is a packet linked technology that enables high speed wireless internet and other data communications. GPRS is more suitable for sending SMS and Web page browsing. GPRS provides about four times greater speed than conventional GSM systems.

  - *Bluetooth Wireless Technology.* A short range radio technology, Bluetooth makes it possible to transmit signals over short distances between telephones, computers and other devices.

- *IEE 802.11.* A type of radio technology used for wireless local area networks (WLANs), this standard was developed by the Institute of Electrical and Electronic Engineers (IEEE).
- *Infrared.* The Infrared Data Association (IrDA) defined a suite of protocols for infrared (IR) exchange of data between two devices, up to 1 or 2 meters apart.
- *Short Message System (SMS).* Used to send and receive small amounts of alphanumeric messages that can include numbers or symbols from one mobile device to another (e.g. exams dates, assignment deadline etc).
- *Multimedia Message System (MMS).* A descendant of SMS, it allows users to send and receive multimedia messages (photos, video clips and sounds) through mobile devices.

4.3.4 Learning Content

The learning content is the central element in the success of any mobile learning application because it determines whether the learner fully engages in the learning experience. 'Learning Content Repository' refers to any written digital material sources (refer to Section 3.5). Separating the educational content into small segments increases ease of use of the content. Digital resources are usually described with additional metadata attached to them and later arranged into more meaningful content, such as lectures, courses, programs, tests, videos, images and quizzes.

Metadata repositories assist in categorising and searching for learning objects. The m-learning application must be able to support all relevant file formats and content types so that learners can access all required content seamlessly. The application's ability to deliver all content allows the learner to interact in the most natural way possible with the application, thereby allowing their true preferences and learning

style to be identified by the system. This, in turn, helps the system adapt more accurately and efficiently to each learner's preferred style.

The main three formats of m-learning content are:

- *Basic learning materials.* Usually text-based learning materials with complementary diagrams, charts or graphics.
- *Multimedia learning materials.* Typically audio, animation and video content.
- *Collaborative learning materials.* Tools to enable learners to share and send information to other learners.

The learning content format/type varies for different learning scenarios, contexts and interfaces, and each format provides distinct support options for learners. The learning content delivered to the learner may be delivered in any format/type, such as text, animation, audio, video, picture or slideshow. The determination of format/type of the learning content is made by sets of constraints and requirements such as display screen size, processing power, memory and location.

The learning content of e-learning and m-learning is very similar when considering digital content and comparing with the traditional learning process. Digital content can be presented in different format (audio, video, animation) to fit learners' preferences. The main difference in learning content between e-learning and m-learning is instructional content designed for both applications. e-learning content is mainly designed for desktop computers and laptops with a large screen and high speed processor. In contrast, m-learning content is designed to fit small screens, low quality, small keypad and other limitations of mobile devices [141].

4.4 Framework Stages

The concept of content adaptive learning is to develop the strategies and methods for creating learning content that always meets learners' needs. Because the combination of mobile devices and wireless networks enables learners to learn anytime anywhere, designing adaptive learning content based on learner profiles creates many contextual and learning content challenges. Contextual challenges relate to the methodologies and techniques for building practical adaptive mobile learning environments for a specific learning task. Learning content challenges relate to the techniques to model the content so it is more feasible for use in mobile learning applications.

To effectively address the adaptation problem we must consider the problem from both sides - the learning context and the learning content - by the integration of two frameworks. The first framework deals with the learner context and has been described in Chapter 3; the second framework is discussed below.

In this section, the framework for modelling the learning content of a mobile learning application is presented. The proposed framework is a system with three stages. First, the necessary information is collected about a learner, either automatically (e.g. location, device, previous usage) or through user input (e.g. age) as explained in Chapter 3. Learning content can then be adapted to meet the unique and personal needs of that learner within their current context. The learning content model allows consideration of individual learning styles, learner context, application capabilities, and material structure, leading to a customisation of the type and delivery format of learning information in response to the user. The learning content modelling is a very important step and cannot be considered as a minor task in the entire modelling process. The generic process of learning content creation is depicted in Figure 4.2 [6].

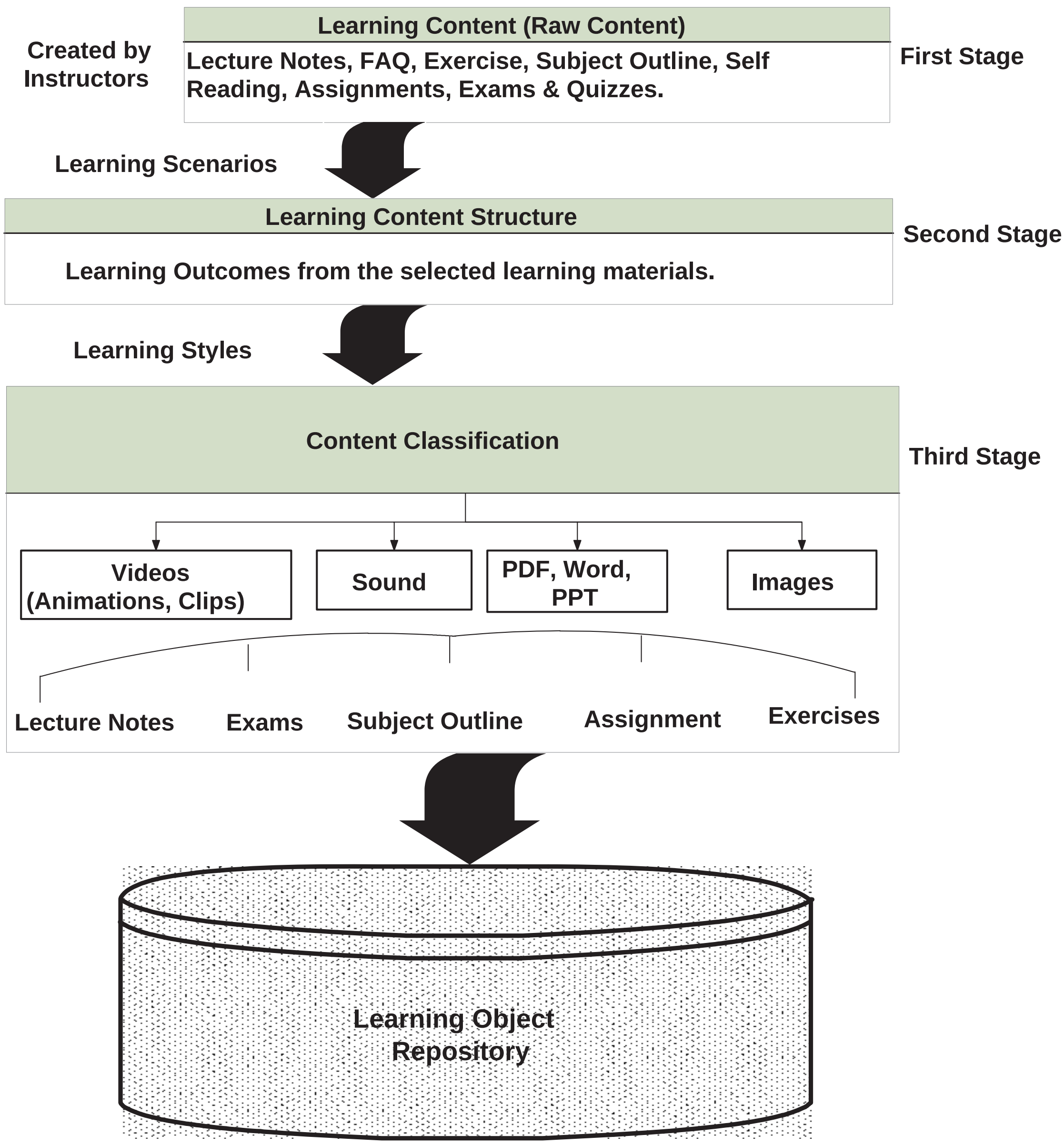

Figure 4.2: Modelling Process for m-learning Content.

4.4.1 First Stage

The stage begins with an instructor using authoring tools to design learning content from scratch. The basic form of learning content is a set of raw learning content such as lecture notes, syllabus, a subject outline, and self-reading tutorials. These elements of raw learning content can be used in one or more course structures, each of which are viewed as instructional designs within the m-learning application. To enable the creator to combine or reuse learning content to suit the varied needs of learners, and to ensure that learners access information in a useful format, the learning materials must be modified into a more appropriate form.

The design of mobile learning content is not a simple task; it requires significant attention and incorporation of all learning participants (tutors, educators and designers). In addition, the integration of learning content and learner context make it complex and impossible to separate both content and context when the learning activity takes place in mobile learning environments [141].

Mobile learning content must be adapted in a way that is suitable to the learning context of each learner and it must comply with learning process and structure. Adapting content means that it is adjusted with response to the presentation format, device limitations, order and course structure of the learning content. For example, the learning content will be delivered according to the learner preferences and context that meets each learner's needs.

The learner content designer must be well aware of the factors that affect the learning activity, such as the importance of designing small size learning content that can be transferred and delivered through simple and advanced technologies (SMS and Video) [101]. Small size learning content allows increased flexibility in the delivery of learning content in a way that accommodate learners' needs and context.

4.4.2 Second Stage

Once the content is organised into an appropriate format, the instructor must consider whether the learning content fits into the course structure that is being adopted by the instructor for the learner or group of learners. The course structure provides details about the learning content, such as the activities the learner has to complete, the classes the learner is required to attend, assignment details, and examinations. All these course activities have their own formats, which the instructor has to design and fit into mobile device technology.

Each piece of learning content must be associated and comply with course objectives, outcomes and structure so the system will know when the learner is required to complete the learning activity (based on the specified course structure). All the designed learning content is stored in a learning objects repository; this will help learners and tutors to easily access, add or edit the learning content.

Reducing the learning content size means reducing resolution and quality to fit mobile limitations and meet different learners and context, however reducing the resolution of learning content may impact negatively on learning experience. Due to the continuous change in location and limited amount of time to deliver the content to mobile learners, it is necessary to reduce the quality to fit with the requirements of a mobile screen, memory and battery.

The design process will consider the different forms of learning content that have been created by the instructor. m-learning content must be designed with the overall learning scenario in mind. In many learning scenarios it is necessary for tasks to be completed in a specified sequence; this structure can be built into the system by the instructor. Each learning activity that is conducted on the learner's mobile device and results in task completion can be considered as one element in a larger learning scenario.

m-learning requires set of technologies to successfully create, manage and deliver learning content for mobile learners. Similar to e-learning, authoring tools are very important tools that are used to assemble and create learning course from scratch. Examples of authoring tools include:

- *Web Authoring Tools.* A type of Web-site or Web page that can be used as an authoring tool to create m-learning content.
- *Course Authoring Tools.* Advanced tools with the design flexibility to build learning content via forms and templates.
- *Content Converters.* Tools that automatically transform any traditional content format (such as Word documents, PowerPoint presentations and graphics) into m-learning courses.

m-learning applications manage content using the same tools that are used in e-learning. These include:

- *Content Management System (CMS).* An integrated system used to create, manage and deliver the learning content [131].
- *Learning Management System (LMS).* Offers functionalities designed to administrate the learning process by managing students' information, tracking students' progress and delivering the learning content [131].
- *Learning Content Management System (LCMS).* Used to manage learning content by enabling the tutor to create a different format of the learning content, store and search [131].

Delivering learning content to mobile learners requires a set of technologies that can effectively understand the learning content, learner context and the limitations associated with the use of mobile devices. These technologies include:

- *Collaboration Tools.* Used to provide a medium for sharing and transferring information between learners either synchronously or asynchrony.
- *Chat.* A real-time delivery method that is used to send and receive comments for the tutor or between the learners.
- *Audio and Video Conferencing.* Used to support m-learning.
- *Shared Whiteboard.* A technology that presents and edits documents.
- *Electronic Mail (e-mail).* Usually brief text-based messages sent from one person to another.

4.4.3 Third Stage

The learning content modelling process involves categorising learners based on learning styles (refer to 4.3.2 Learning Styles). Each category has its own learning content representation that facilitates effective learning to match the style of that learner. Using the model in Figure 4.2, the application of learner styles and preferences can be used. Instructors can consciously choose to provide content in varied learning styles, as well as learners (or the adaptive system on behalf of the learner) selecting content that meets personalised preferences.

In this work, four main learning content formats are considered: These are:

1. *Text-based learning content.* The content in this format must be small in size (e.g. 10 kilobytes), direct and meaningful. It must include a small set of information, a summary of learning content and multiple choice questions and answers to allow learners to evaluate their understanding level at the completion of the learning task. A small text font is required to ensure compatibility with learner device properties.

2. PDF-based learning content. The content in this format must be medium in size (e.g. 70 kilobytes), high quality, direct and meaningful. It must also include a small set of information, a summary of learning content and multiple choice questions and answers to allow learners to evaluate their understanding level at the completion of learning task. A normal sized text font is recommended. Mobile devices must have the required software installed to use PDF-based learning content.

3. Multimedia-based learning content - Audio. The design of audio learning content is based on text-based learning content. For each text-based content item, there will be an audio-based version that provides spoken content to match the text-based content. Each audio-based content item requires a reasonable size (e.g. 40 kilobytes) that can be accessed over varied network bandwidths.

4. Multimedia-based learning content - Video. The design of video learning content is based on the size, quality and software capabilities of learner device. Each video-based content item must not be excessively large, and must be of a quality that is acceptable to successfully deliver information to a mobile learner, taking into consideration device battery, network bandwidth and other device limitations. Video learning content is required when a text-based explanation of a course will not be enough for the learner to understand. As a result, a combination of text-based and graphical explanation can be used to enhance the learning experience.

Learning through a mobile device allows learning to become truly personalised. Learners have the option to choose learning content based on their interests, thus making learning learner-centric. In contrast to typical electronic learning products, this access to personalised information means each learner gets the resources they need in a timely manner while minimising wasted bandwidth. The flexibility to access specific

information using mobile devices also helps to increase the productivity of an individual. Providing immediate access to relevant and interesting information, based on the individual user's requirements, encourages use and increases engagement because learners are able to access the information they want wherever they are.

4.5 Summary

Personalisation in m-learning applications via the integration of two adaptive frameworks has been discussed. The first framework, described in Chapter 3, in concerned with adapting to the context of the learner. The second framework, described in this chapter, is concerned with adapting the learning content.

The new framework presented in this chapter depicts the process of modelling learning content to satisfy individual learner characteristics by taking into consideration the learner's context, learning style, application capabilities, learning content. In response to this information, the adaptive learning content framework provides learners with learning materials, activities, and experiences that fit their individual needs and requirements. The proposed design includes three stages which can be regarded as subsystems of the adaptation framework (refer to Section 4.4). The instructor designs learning content (raw) from scratch using authoring tools. The learning content structure provides details about the learning content, such as the activities the learner has to complete. Learning content classification involves categorising learners based on learning styles. Learning content representation is established based on this classification, to facilitate effective learning that matches the style of each learner. The ultimate goal of the framework is to provide a logical structure for the process of adapting learning content to meet the needs of mobile users.

Chapter 5

Enhanced Learner Model for Adaptive Mobile Learning

5.1 Outline

Personalisation and learner modelling are becoming more important in the area of mobile learning applications, taking into consideration learners' interests, preferences and contextual information. Students nowadays are able to learn anywhere and at any time. Mobile learning application content is one of several factors within various contexts that play an important role in the success of the adaptation process. The vast amount of data involved in any successful adaptation process creates complexity and poses serious challenges. This chapter focuses on how to model the learner and all possible contexts in an extensible way that can be used for personalisation in mobile learning. Challenges and current solutions related to learner modelling are discussed in this chapter, and the Enhanced Learner Model structure to be used in a mobile learning system is proposed. The proposed structure provides personalisation by adopting a hybrid approach combining two machine learning techniques.

The main objective of the Enhanced Learner Model is to model the learner and all possible contexts related to his/her current situation in an extensible way so that they can be used for personalisation. The Enhanced Learner Model consists of four main representational components: *Learner Status* (Section 5.4.1), *Situation Status* (Section 5.4.2), *Knowledge and Shared Properties Status* (Section 5.4.3) and *Educational Activity Status* (Section 5.4.4). Sections 5.3 provides the workflow of the personalisation system, and also presents the learner model acquisition techniques. The structure of the proposed learner model is presented in Section 5.4. The structure of the Learner Profile Representation layer is presented in Section 5.5. Section 5.6 discusses the structure of the reasoning engine, which adopts hybrid machine learning techniques in two stages. Finally, in Section 5.7 we conclude with some key points about this chapter.

5.2 Introduction

In recent years, the use of mobile phones and wireless networks in educational institutes has become widespread, and hence, mobile learning has become widespread. Most students already possess handheld devices, such as personal digital assistants (PDAs) and mobile phones, and these are being used by a variety of learners in a variety of different environments. Several educational institutes are using wireless technology to deliver educational materials to these devices, however to date this has been largely limited to mobile phones. The diversity of learners' characteristics as well as mobile devices and networks requires personalisation for different needs. Context awareness is important because it allows the environment to be used in a way that supports the learner.

The reason for modelling context is to get a clear understanding of a learner's activity, thus allowing for the delivery of more appropriate learning content and services. A significant issue of adaptive and adaptable systems is the transformation from the learner input to the system output [83].

A system is considered to be adaptable if the user is in control of the complete adaptation process (*initiation*, *proposal*, *selection* and *production* of adaptation) [112]. Systems are considered as adaptive if these adaptation processes are done without the contribution of the user. The purpose of adaptable systems is to provide extra support for a variety of learners, given that they may have different skills and motivations in terms of learning methods.

Adaptive system developers can use context information for a variety of purposes, including resolving references, tailoring lists of options, triggering automatic behaviours and tagging information for retrieval [94]. The design of an adaptable system must focus on reducing all barriers that are encountered during the learning process by analysing or filtering what can be known about a learner and how to

respond to it.

To facilitate personalised learner services, the system requires a clear understanding about learners and their interaction with the system. Learner context inputs can contain various demographic properties, as well as the learner's skills and capabilities, interest and preferences. Also, the current situation can be used as a factor for adapting the service. Extra parameters to the learner inputs, or a direct request to define the learner's current situation, may be required to achieve this successfully.

The learners have the option to choose learning content based on their interest. The flexibility to access specific information using mobile devices also helps to increase the productivity of an individual. Providing immediate access to relevant and interesting information, based on the individual learners requirements, encourages use and increases engagement because learners are able to access the information they want wherever they are.

Robert Kass [74] defines learner models as "systems that tailor their behaviour to individual learners needs and often have an explicit representation structure that contains information about their learners". A learner model is also defined as a set of information structures designed to represent one or more of the following elements [82]:

- goals, plans and preferences;
- relevant common characteristics of learners' stereotypes;
- classification of learner stereotypes;
- learner behaviour;
- assumptions about the learner based on the interaction history; and/or
- interaction histories of many learners into groups.

More advanced learner models can contain information related to psychic, emotional, and physical states.

Learner models are developed to establish understanding of three main areas [152]:

- the cognitive processes underpinning a user's actions;
- the differences between the skills of a specific user and an expert; and
- the behavioural patterns and characteristics.

Desirable characteristics for an ideal learning system that is able to adapt the course to learner characteristics have been identified as [165]: *knowledge*, *objectives* and *learning goals*. The following characteristics were considered:

- a pedagogical approach for the adaptation,
- pedagogical rules for update,
- levels of sequencing of learning content,
- learner progress consideration,
- re-routing (re-plan), and
- use of standards to control the adaptation process.

The learner model contains all the information that the system knows about the learner. It is generally initialised either with default values or by querying the learner. The reasoning engine combines the learner profile with other models of the system to derive new facts about the learner. The reasoning engine can update the learner profile with the derived facts or initiate an action in the application (such as interrupting the learner with a suggestion).

In order to provide personalisation services, systems need to be able to make inferences about their learner, and this can be done by making assumptions about learners

based on their interaction with system. To achieve this, we apply techniques such as machine learning algorithms. Once a learner model is created, reasoning from that model can be made possible by drawing several assumptions about the learner's current situation.

Machine learning techniques are commonly used for learner modelling because of the complex nature of relationships between learner contexts, which are difficult to represent. The process is based on storing and exploiting information about the learner. However, learners differ in traits such as skills, aptitudes and preferences for processing information, constructing meaning from information, and applying it to real-world situations [51]. This is an acknowledged problem for Web-based education, with Aiken and Epstein [5] stating that learner models should acknowledge that learners might have different learning styles and skills.

Neural Networks [146] [163], k-Nearest Neighbors [140], Bayesian Network [158] [103] [143] and Fuzzy [10] algorithms are some of the techniques that have been used to model learner contexts. There is a great need to identify the interests/characteristics of the learner on the delivered learning content by modelling and understanding the learners' actions and needs. Most researchers agree that learner modelling is the most important part of any adaptable learning system with many challenges. Some approaches based on machine learning techniques are provided Section 2.7.

5.3 Workflow of Personalisation System

In this section, we describe the workflow of the personalisation system (see Figure 5.1), which can be summarised in the following steps:

Data Collection:

Initiation of a learning activity begins when the learner carries out the learning activity by interacting with the m-learning application through the selection of some

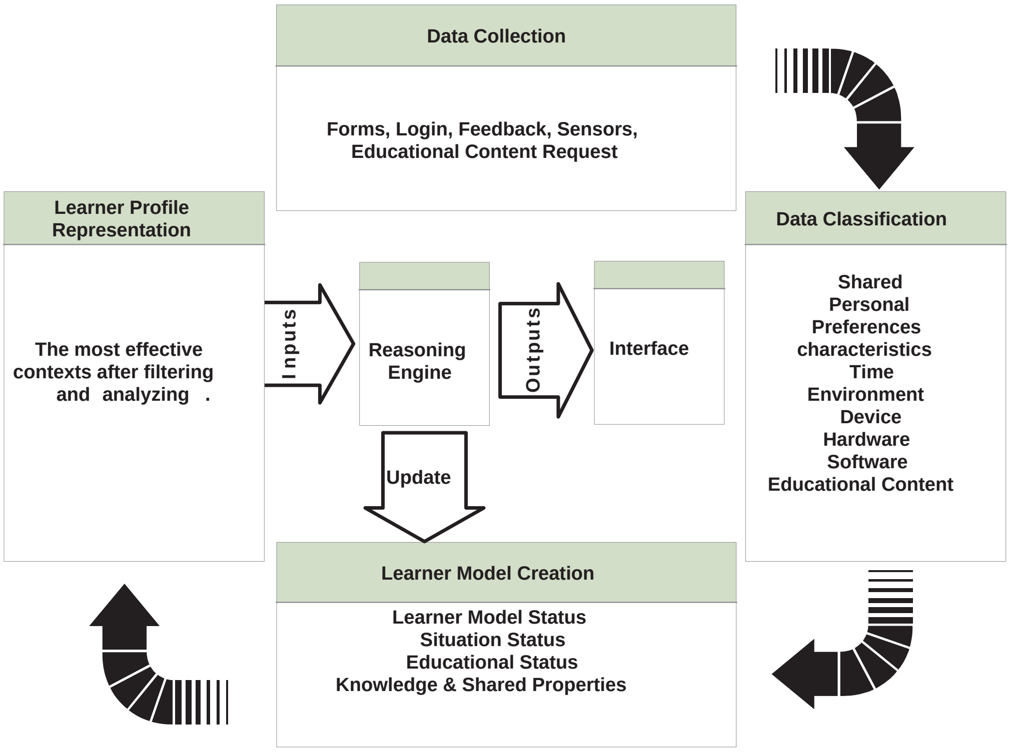

Figure 5.1: Workflow of the Personalisation System.

actions presented on the learner's device interface. If using the application for the first time, the learner will be asked to complete some forms to record personal information (e.g. name, age and job). Other information will be gathered by alternate means (e.g. sensors). Identity verification is essential; the system must first be able to identify the learner in order to collect the required information to perform the adaptation based on the learner request. This will result in improved accuracy and consistency.

There are numerous available techniques to gather all related information about learners and the methods/algorithms used to process such information to create learner profiles and provide adapted content [63]. To construct a learner model, all related information about the learner such as (hardware, software, capabilities, preferences of the learner, etc.) should be gathered and transferred to the learner model. Both *Implicit* and *Explicit* information is collected and recorded in the learner model. The methods of gathering learner related data can be categorised into five main sections:

- *Forms*: Direct questions to the learner are the initial information needed to construct the learner model. This method is an effective way to gather general information about the learner such as demographic data, interests, preferences, etc. Set of questions are used mainly for collecting contextual and personal information about the learner. This information will be used by the reasoning engine. While the information provides the ability to further personalise the experience for the user, an excessive number of questions is likely to disturb the learner [18] [139]. Existing adaptive systems commonly use self report and questionnaires to obtain initial information [91] [136].

- *Login*: In order to provide personalisation for individual learners, the system must first be able to identify the learner in order to collect the information required to perform the personalisation.

- *Feedback*: This includes the information obtained through interactions between the learner and the system, including the learner history page visits, access length and frequency, and outcomes given by the system. However, information gathered in this manner may not be completely reliable [166]. The system is able to develop a concept of the attitude of the learner through the interaction with the system, i.e. observing learners' actions and behaviours (e.g. the learner will be able to tell the system whether they like or dislike the adapted content using a like/dislike response). The like/dislike responses will have a significant impact on the structure of the learner profile, with feedback from the learner used to inform the changing learner profile, and hence to alter the interaction between the system and the learner.

- *Information from sensors*: A range of data can be collected from sensors, such as the learner's position from in-built GPS, time from the in-built clock, and

noise from the in-built microphone.

- *Assumption*: In some situations, more information about the learner is needed but cannot be obtained through the direct questions or sensors. For example, the learner background knowledge on the subject is unknown for the system. In such situations, the system makes an assumption at the beginning that the learner has no background knowledge. The system must offer a help option for inexperienced learners when they use the application for the first time [85].

The data obtained using the above methods should be transferred to variables and stored in a learner model. Variables stored in a learner model can have three forms: boolean, discrete, and continuous.

The system is able to offer a learning activity, and the learner will decide whether to accept or reject the learning activity. If the learner chooses not to carry out the learning activity, there is no further process. If the learner is willing to participate and wants to begin the activity, the system will issue a start up application to indicate that the learner wants to undertake a learning activity.

Data Classification:

Learner data is gathered and categorised into several class types as shown in Figure 5.1. It consists of:

- *Personal Context* - all attributes relevant to the learner throughout his/her use of the system; and
- *Shared Context* - attributes relevant to all learners when using the system.

Learner Model Creation:

The learner model aims to make information systems learner-friendly by adapting the behaviour of the system to the needs of the individual. The structure of the learner model will be discussed in Section 5.4. When the learning activity is in progress or

the learner is engaged in the learning activity, the system can interrupt this process to the benefit of the learner at any time, providing advice about how the learner can best deal with the learning materials. The interruption can take many means (e.g. device limitation, environmental condition). Learner Profile Representation will be discussed in Section 5.5 and the Reasoning Engine in Section 5.6.

Interface:

The interface layer is formed by events that are processed by the adaptation system, as well as the questions about the learner that he/she can answer. The interface forms a description of the way the application interacts with the adaptation system. The learner model contains the information about the learner that has been collected so far.

5.4 Enhanced Learner Model Structure

As shown in Figure 5.1, the goal of this research is to construct a system which will help learners to accomplish learning activities by delivering adapted learning content based on each learner's current context. The learner modelling process has three steps: gathering data related to the learner, creating the learner model and updating the learner model. To better understand the idea of the enhanced learner model structure, consider the following typical scenarios and the activities involved in each:

- *It is 8:00am Monday morning and Alison is travelling by bus/car/train from her flat to attend her lecture at the university which starts at 8:30am. Alison checks her PDA (3G network, 24kbps bandwidth) to review to her lecture notes.*
- *Sara is in a restaurant for 20 minutes and wants to review her assessments using her smart phone (GSM network, 4kbps bandwidth).*
- *It is the weekend and Mark is at home for an hour and wants to review his*

reading materials using his Nokia phone (3G network, 60kbps bandwidth).

Many contextual elements are contained in the scenarios described above. We can identify different activities, places, times, events, constraints, locations and other environmental assumptions. The overall activities in the above scenarios could be labeled as *Learning Activity*; these activities can be separated into different actions:

- Learning Materials (lecture notes, assignments, etc),
- Personal Information (name, gender, language, occupation, learning style, etc),
- Location (home, class, bus, train, road),
- Time (weekdays, weekend),
- Mobile Device (PDA, smart phones, etc),
- Environment (sunny, windy, noisy, raining, etc),
- Network (3G, GSM, WIFI), and
- Bandwidth (low, high, middle).

Based on the above scenarios, we combined learner information and contexts to achieve maximum integration, which will result in information-rich learner profiles.

This Enhanced Learner Model consists of four main components, namely the representation of the *Learner Status*, the *Situation Status*, the *Knowledge and Shared Properties Status* and *Educational Activity Status*. It should be noted that there is some intersection between the components, as shown in Figure 5.2 [8].

In many cases, the vast amounts of data obtained through the adaptation process acquisition methods creates complexity; while challenging, this complexity will provide improved and more reliable results. In order to describe the learner context and content, we need to add extra assumption to help the reasoning engine to make use from the data from learner and sensors and the extra assumptions.

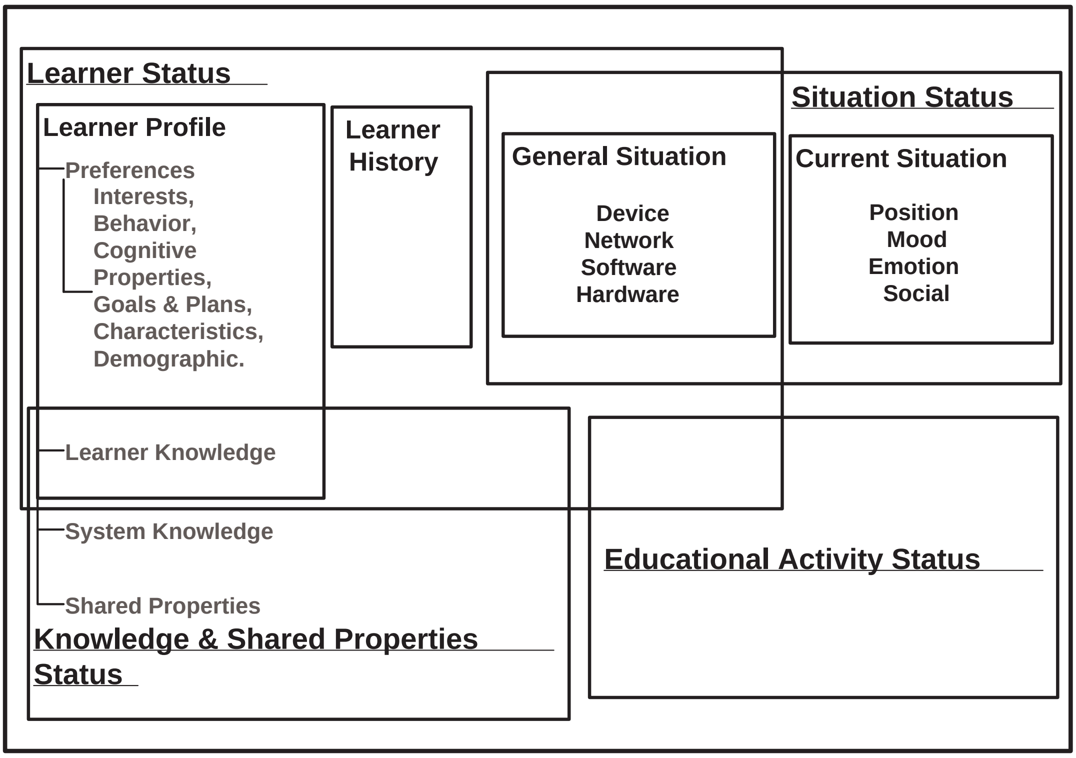

Figure 5.2: Enhanced Learner Model Components.

5.4.1 Learner Status

The first main component of the Enhanced Learner Model is the *Learner Status* which encompasses the *learner profile*, *learner history*, *general situation*, *learner knowledge* and the *educational activity.* It describes the learner based on assumptions about that user's knowledge and preferences, the user's interaction history and a description of the user's general situation. Preferences depict the user's interest in certain topics; this category captures all the common information that can constitute a learner profile. This is made up of:

- *Learner profile*: A record of specific variables that depict the learner preferences, it includes information such as email address, age, gender, education and profession, preferences and interests, objectives, aims and plans, health, physical abilities, cognitive abilities, social abilities, emotions and feelings, intentions,

schedule, current needs, and desires. It may also include a series of optional information. Behavioural preferences should represent some aspects of the user's general behaviour, such as 'the learner usually does not like to view his lecture notes in PDF format while he is on the train'.

- *Educational activity*: Captures learners' preferences, considering the means and the media on which the user receives educational content information while browsing the educational materials.
- *Learner history*: Helps to keep track of completed adapted queries.
- *General situation*: Describes the general situation, i.e. device, network, software and hardware.
- *Learner knowledge*: Provides references to information previously supplied by the learner, to allow sufficient interaction and to refine the learner preferences to make use of resources in a better way.

The learner modelling process involves the categorisation of learners based on learning styles. Learning content representation is determined on a category basis, to facilitate effective learning that matches the style of each learner.

5.4.2 Situation Status

The second main component of the Enhanced Learner Model is the *Situation Status*. The learner's *current situation* and *general situation* inform the *Situation Status*. This component describes the learner's current situation with its various characteristics in the real world.

The *Situation Status* includes contexts that are associated with:

- the mobile device being used to access the learning content in the mobile learning application; and

- the mobile connectivity used to deliver the learning content.

The *current situation* describes the various characteristics in the real world. It consist of the following:

- learner's current position (in-built GPS),
- the time of the learning activity (in-built clock),
- the noise during the learning activity (in-built microphone), and
- inferred assumptions derived from the learner model.

The *general situation* describes the overall status of the system. It consists of the following:

- devices,
- networks,
- hardware and software resources,
- device capabilities,
- bandwidth,
- processor speed,
- storage capacity,
- resolution,
- sound quality,
- sound power,
- battery,

- network capacity
- connectivity, and
- Plug in OS.

It is essential to specify the learner device capabilities and features due to the significant impact of these details when determining an appropriate learning content format to be deliver to the learner. The parameters/features that characterise mobile devices are the main factors that will be used to characterise this context and that will be used by the reasoning engine to determine the adapted learning content.

The learner device type used to access the learning content must be specified in order to determine device limitations and capabilities. For example, mobile devices can be connected to Internet through different wireless technologies. Each of them has different data transfer rate and can be used either for wide or limited coverage. GPRS, WAP and WiFi are the most common technologies that are used to connect to the Internet via PDAs, Smartphone and Cellular Phones as mention in Section 4.3.3. As a result, the system must be able to specify the type of wireless connection to help delivering the most suited learning format.

Mobile devices such as cell phones suffer from small screens, poor input methods and limited battery life. Therefore, the adaptation engine needs to take all these constraints into consideration. It is necessary to constantly monitor changes in network connectivity as users move locations; when the learner moves from a classroom to the library, ad hoc changes must be made to accommodate for changes in the environment and, as such, the user's context.

5.4.3 Knowledge and Shared Properties Status

The third main component of the Enhanced Leaner Model is the *Knowledge and Shared Properties Status*. This component represents the *learner's knowledge*, the *system's knowledge* and *shared properties*. The learner knowledge is referenced to existing information supplied by the system, improving interaction and further refining user preferences.

- *Learner knowledge*: Based on assumptions about the learner's knowledge of the system, relationships between inputs, and facts and rules regarding the mobile learning application. Learner knowledge is an essential factor in the provision of personalisation. Many approaches have been proposed to build adaptive systems. The design of each of these systems has taken learner knowledge into account. The main purpose of collecting learner knowledge is to adjust the learner interface presentation so that unnecessary explanations are removed. This removes the likelihood of learners becoming confused by details they cannot understand and making the learner bored [115] [88] [99] [81] [16] [134]. Examples of such systems are ELM-ART II [154], Hypadapter [64] and ISIS-Tutor [33].

- *System knowledge*: It is generally initialised either with default values or by querying the learner throughout his/her interaction with the system. The main purpose of collecting system knowledge is to deliver the learning content based on learner profile, and derive new facts about the learner.

- *Shared properties*: Represents the available tools that help learners to accomplish tasks that involve teamwork. They include tools that allow coordination between teamwork members, and provide resources that are needed to help learners during a specific activity.

5.4.4 Educational Activity Status

The fourth main component of the Enhanced Learner Model is the *Educational Activity Status*. It consists of any service or facility that supplies a learner with general electronic information and educational content that aids in the acquisition of knowledge, regardless of location and time [152]. Specifically, the component includes:

- subject outline,
- requirements,
- purpose,
- objectives,
- expected outcomes,
- pedagogical theory,
- content, presentation,
- self reading,
- resources,
- participants and teams,
- images, clips,
- animations,
- FAQs, and
- achievements and results.

The learning content format/type varies for different learning scenarios, contexts and interfaces, and each format provides distinct support options for learners. The learning content delivered to the learner may be delivered in any format/type, such as text, animation, audio, video, picture or slide show.

5.5 Learner Profile Representation

In order to achieve personalised services, it must be possible to specify user interests. This can be achieved using a machine learning algorithm, which takes a learner's information for input, and then compares and analyses the learner's need, interest and environment. As previously discussed, the proposed Enhanced Learner Model represents the upper layer (see Section 3.4.3 Learner Model Layer) which feeds all data to the next layer (see Section 3.4.5 Learner Profile Representation Layer) after extracting necessary data (see Section 3.4.4 Information Extraction Layer) based on the learner's current situation (see Figure 5.3)[7].

Creating Learner Profile Representation allows for improved accuracy of the results, given a sufficiently expressive keyword. Profiles are derived from a set of common keywords. A learner can indicate his interest in a specific domain or even single concept by specifying a value in a predefined range. Precise information allows the system to more accurately support learners' decision-making; this information represents the main inputs for the reasoning engine (see Figure. 5.1 and 5.4).

5.6 Reasoning Engine

The reasoning engine is part of the adaptation component (see Section 3.4.6). The reasoning engine transforms content from one state to another in order to meet the constraints of the learner's context. Machine learning techniques have been applied

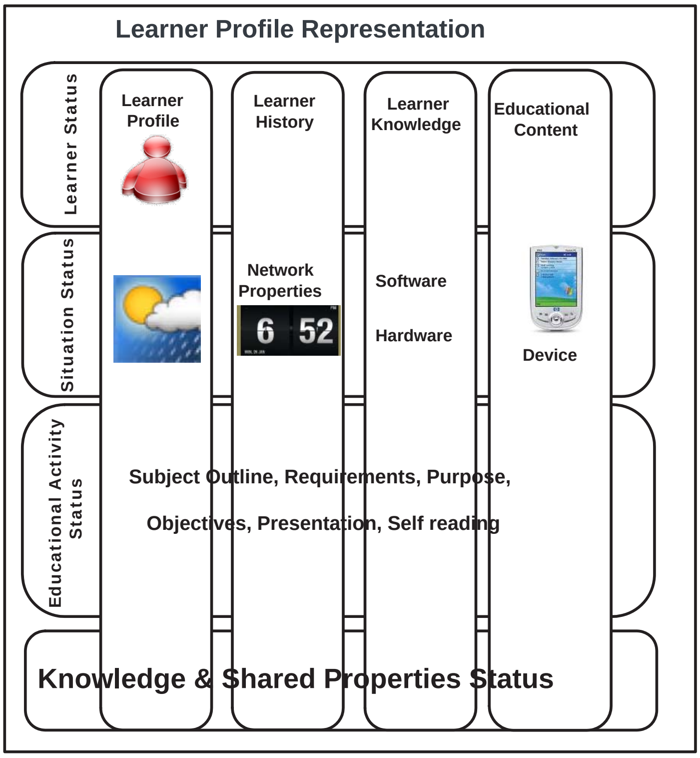

Figure 5.3: The Structure of Learner Profile Representation.

to learner modelling problems to assist in obtaining models of individual learners and grouping them into categories. This process is essential in the development of an accurate and useful system that can modify its behaviour over time. Machine learning techniques are applied to the data obtained in the earlier layers, in order to capture user behaviour patterns.

Any machine learning technique based on adaptation should consider the following conditions to provide a wide range of possibilities on mobile learning [110]:

- the amount of effort required to provide the system with necessary background knowledge,
- the amount of computational time required,
- the amount of input data required to be able to make useful decisions,
- the appropriate handling of noise,

- uncertainty (one of the most important problems in the construction of a learner model [35], and
- validity.

Learner modelling system make decisions using manually constructed rules about the problem developed by analysing various situations, in order to make proper decisions for the learner. In the system presented in this thesis, the prediction of the educational content is strongly based on the Learner's Profile Representation Layer (refer to Section 3.4.5). Once the required information about the learner is collected (implicitly or explicitly), the adapted appropriate content is delivered. The system observes learner behaviour and preferences, and recommends educational materials that are similar to those chosen in the past.

More advanced algorithms, such as clustering, are used to group together users having similar characteristics, and classification in order to map items into various classes. The reasoning engine works on the basis of processing the incoming learner data from the learner interface, and initiating the adaptation process with the help of a learner model and Learner Profile Representation generated on the layers above the Reasoning Layer.

The main purpose of this process is to provide the learner with education materials matched to the user's current context. The reasoning engine consists of two stages: Fuzzy Logic and Neural Networks . The overall system architecture is shown in Figure 5.4. Presentation of a detailed structure of the reasoning engine is presented in Chapter 7.

Fuzzy Logic was introduced by Zadeh 1965 [164], and is used in expert systems and artificial intelligence applications.

Neural Networks (NN) have the ability to learn. These networks can select desired outputs based upon sample inputs. They also have the ability to generalise, producing

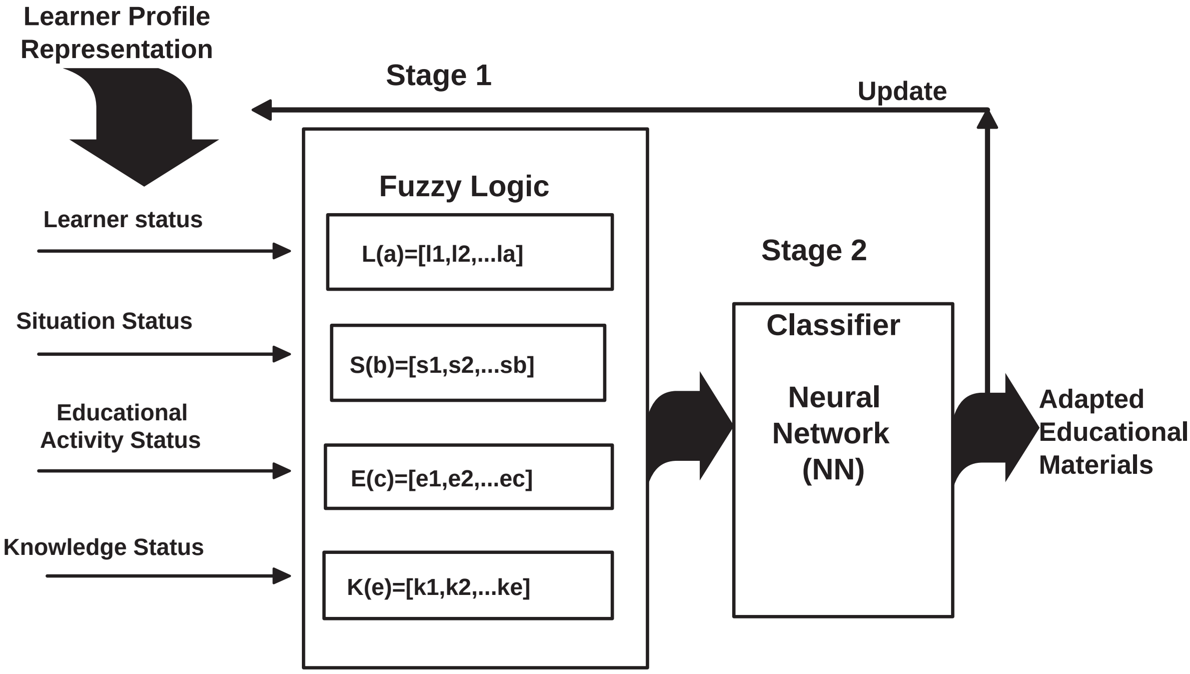

Figure 5.4: Overview of the Proposed Reasoning Engine.

reasonable outputs in response to inputs for which no training has been provided.

The performance is estimated on the test set. Cross-validation can be used by placing two sets for validation and then average the error for both sets. The input to the adaptation engine includes the *Learner Status*, the *Situation Status*, the *Knowledge and Shared Properties Status* and the *Educational Activity Status* in the form of Learner Profile Representation (see Figure 5.3). The variables that describe the input may be either declared by the user or measured. The output consists of the adapted educational activities based on the refined Learner Profile Representation. Learner model is logically partitioned into smaller elements or stereotypes in the form of Learner Profile Representation, which can represent the entire learning process.

5.7 Summary

This chapter has described the architecture that provides personalisation services based on the contextual learner model. Considering modelling of the learner as a complex problem, several techniques have been used for different purposes in the literature. Challenges and current solutions about learner modelling are listed in this chapter with the proposal of a learner modelling framework to be used in a mobile learning system. The ultimate goal of the Enhanced Learner Model is to enrich the learner profile, because this is the main input for the reasoning layer. The Enhanced Learner Model consists of four main components, namely the representation of the *Learner Status*, the *Situation Status*, the *Knowledge and Shared Properties Status* and *Educational Activity Status.* The proposed system provides personalisation by adopting a hybrid approach, combining two machine learning techniques (Fuzzy Logic and Neural Networks) to provide an improved experience for mobile learners. The effectiveness of the system was evaluated both quantitatively and qualitatively using a series of simulations. Findings are presented in Chapter 8.

Chapter 6

Fuzzy Logic, Artificial Neural Networks and Adaptive Neuro-Fuzzy Inference System (ANFIS)

6.1 Outline

This chapter addresses three key topics: Fuzzy Logic, Neural Networks and Adaptive Neuro-Fuzzy Inference Systems (ANFIS). The discussion of Fuzzy Logic provides an overview of *fuzzy sets*, *membership functions*, *fuzzy system components* and *inference algorithms*. Neural Networks learning methods and types are provided. The principles of Adaptive Neuro-Fuzzy Inference Systems (ANFIS), a general overview of Neuro-Fuzzy system types, Fuzzy Inference System (FIS) and the ANFIS hybrid learning algorithm are also presented.

6.2 Introduction

Fuzzy Logic was introduced by Zadeh (1965) [164]. It allow situations or problems to be described and processed in linguistic terms, such as "Hot" or "Tall", instead of precise numeric values, such as 130 degrees or 140cm. Fuzzy Logic was specifically designed to mathematically represent uncertainties associated with learner cognitive development.

An Artificial Neural Network (ANN) is an information processing paradigm inspired by the way biological nervous systems process information [50]. Neural Networks have the ability to detect patterns from complicated data, learning how to perform activities based on the training data. Neural Networks will be used to process the information received from the learner and the system in order to provide an improved experience.

In general, Neural Networks (NN) have the ability to learn, choosing desired outputs based upon sample inputs. The mobile learning application is exploited from the supervised learning in which certain characters are fed to the input layer of a Neural Network in which desired outputs are obtained. The character features are propagated

through the neurons; these features are basically the pixels' values of the characters. The training process performs iterative minimisation of errors between the input and the output over the training set. The Neural Network is trained on the training set, while the error is recorded on the validation set. Training stops when the error on the validation set is at its minimum, to avoid over-fitting.

Fuzzy Logic and Neural Networks techniques have been proposed in a variety of user and learner modelling approaches [60] [68] [114] [161], due to their abilities to learn from uncertainty and incomplete patterns of learner behaviour.

The modelling techniques used in mobile learning applications are important in the provision of accurate and optimal use of mobile devices for learners. In modelling of the mobile learning scenarios, the features of mobile devices are very important. In order to deliver learning content to mobile learners, internal and external factors must be considered. Examples of such factors are: environment, network bandwidth, change of location, battery life, screen size and time of learning activity. The mobile learning model based on Adaptive Neuro-Fuzzy Inference System (ANFIS) is built basically for adaptation purposes; it is very effective for simulation purposes.

An ANFIS with Neural Networks learning capability and the advantages of a rule-base fuzzy system can improve its performance significantly. For Neural Networks, the training process builds the system. However, using an ANFIS, the system is built by Fuzzy Logic definitions and is then refined using Neural Networks training algorithms. ANFIS is the Fuzzy Logic system that uses the learning abilities of Neural Networks to enhance the intelligent system's performance using a priori knowledge.

Each technique has particular properties (e.g. the ability to learn). For example, while Neural Networks are good at recognising patterns, they are not good at explaining how they reach their decisions. In contrast, Fuzzy Logic systems are good at explaining their decisions but they cannot automatically acquire the rules they use

to make those decisions.

6.3 Fuzzy Logic

Fuzzy Logic was introduced by Zadeh [164], as a new means to solve real world control problems by manipulating data that is not precise. This theory is a result of imprecise information in the real world, and mimics the human ability to effectively process imprecise and "fuzzy" information. Fuzzy Logic is used in the fields of expert systems and artificial intelligence, and provides an effective means of describing the behaviour of systems that are too complex. A Fuzzy Logic system uses Fuzzy Logic instead of boolean logic. The core of a fuzzy system is a knowledge base consisting of the so called fuzzy IF-THEN rules.

Fuzzy Logic has been implemented in many applications due to its simplicity of representing imprecise data, and it has been combined with Neural Networks to produce self-learning controllers, better known as Neuro-Fuzzy systems. The next section presents the basic principles of Fuzzy Logic.

6.3.1 Fuzzy Sets

In the classical set theory, a set of elements can be represented as follows:

$$A = \{a_1, a_2, a_3.....a_n\} \tag{6.1}$$

$a_i(i = 1, 2, 3...n)$ altogether represent the elements of set A which is a subset of the universal set X. The set A can be represented for all elements $x \in X$

$$\mu_A(X) = \begin{cases} 1 & \text{if } x \in A \\ 0 & \text{if } x \notin A \end{cases} \tag{6.2}$$

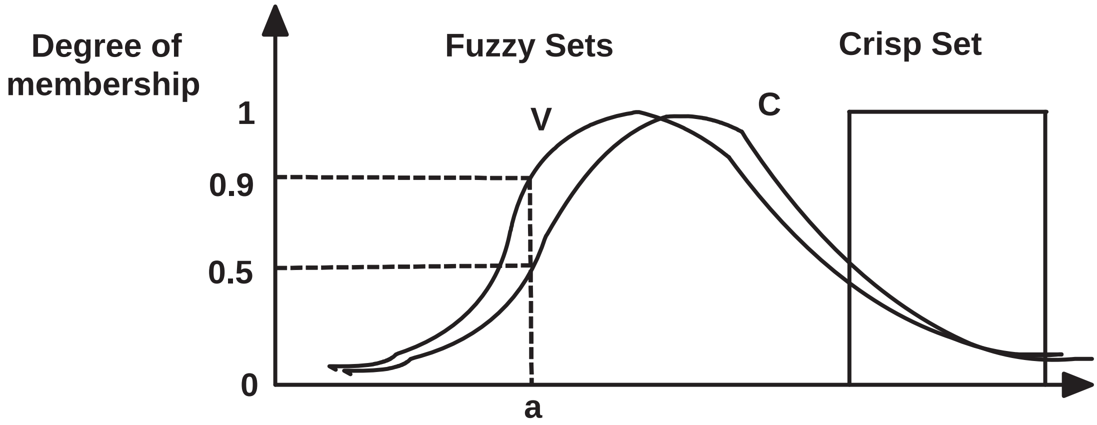

Figure 6.1: Crisp and Fuzzy Sets.

$\mu_A(X)$ has only the values 0 ("false") and 1 ("true").

The main focus in fuzzy systems is that truth values in Fuzzy Logic or in fuzzy sets are indicated by a value on the range $[0.0, 1.0]$, with 0.0 representing absolute falseness and 1.0 representing absolute truth. The main difference between fuzzy and boolean logic lies in the 'truth' value. The 'truth' in Boolean logic is either 1 or 0 (True or False) but in Fuzzy Logic is any value between 0 and 1, which indicates the degree to which something is true or false.

A fuzzy system is a system that uses Fuzzy Logic instead of boolean logic. Fuzzy Logic uses the whole interval between 0 (False) and 1 (True) to describe and reasoning human behaviour of systems that are too complex. Fuzzy Logic is further explained by the concepts of "crisp" and "fuzzy" sets. In a crisp set, an element will either not belong to the set ('False' value of 0) or belong to the set ('Truth' value of 1). In a fuzzy set, an element belongs to a set with certain possibility between 0 and 1.

This is shown in Figure 6.1. The element a in the fuzzy set has a possibility factor of 0.5, or we can say that element a belongs to set U with possibility of 0.5. In the crisp set, the element a belongs to multiple sets with varying degrees of possibility. Figure 6.1 shows that element a belongs to fuzzy set C with a possibility of 0.5 and

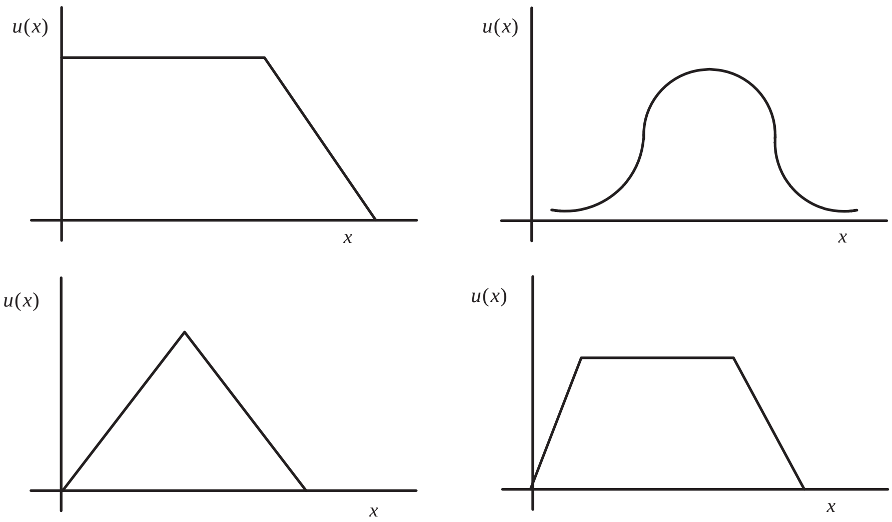

Figure 6.2: Different Shapes of Fuzzy Membership Functions.

may belong to another set of V with a possibility of 0.9.

6.3.2 Membership Functions

The membership function $\mu_A(X)$ describes the membership of the elements x of the base set X in the fuzzy set A. Different types and shapes of membership functions, such as triangular or trapezoidal functions, are show in Figure 6.2. The value of membership $\mu_A(X)$ of a membership function is in the unit interval $[0, 1]$.

Fuzzy sets have membership functions defined between 0 and 1 [164]. This means that if we take an attribute (e.g. 'blue') we can express the colour of any particular shirt as a value in this fuzzy set. We may say for example that it is 20% blue, therefore it has a fuzzy truth value or membership function of 0.2. The relation of the fuzzy truth value to the actual value depends upon the desired output from the real world, and this is arbitrary.

The most common membership function shapes used are Triangular, Trapezoidal, Gaussian, Generalized-bell and Sigmoidal$Z-$ and $S-$ functions, as shown in Figure 6.2.

Triangular Membership Function:

A triangular membership function is described by three parameters $\{a, b, c\}$,

$$f(x, a, b, c) = \left\{ \begin{matrix} 0, & x \leq a \\ \frac{x-a}{b-a}, & a \leq x \leq b \\ \frac{c-x}{c-b}, & b \leq x \leq c \\ 0, & c \leq x \end{matrix} \right\} \tag{6.3}$$

Or by an alternative formula:

$$f(x, a, b, c) = max\left(min\left(\frac{x-a}{b-a}, \frac{c-x}{c-b}\right), 0\right) \tag{6.4}$$

Trapezoidal Membership Function:

A trapezoidal membership function is described by four parameters $\{a, b, c, d\}$,

$$f(x, a, b, c, d) = \left\{ \begin{matrix} 0, & x \leq a \\ \frac{x-a}{b-a}, & a \leq x \leq b \\ 1, & b \leq x \leq c \\ \frac{d-x}{d-c}, & c \leq x \leq d \\ 0, & d \leq x \end{matrix} \right\} \tag{6.5}$$

Or by an alternative formula:

$$f(x, a, b, c, d) = max\left(min\left(\frac{x-a}{b-a}, 1, \frac{d-x}{d-c}\right), 0\right) \tag{6.6}$$

Gaussian Membership Function:

A Gaussian membership function has two parameters: c responsible for its center, and σ responsible for its width;

$$f(x, \sigma, c) = e^{\frac{-(x-c)^2}{2\sigma^2}} \tag{6.7}$$

Generalized Bell Membership Function:

A generalized bell membership function has three parameters: a responsible for its width, c responsible for its center, and b responsible for its slopes;

$$f(x,a,b,c) = \frac{1}{1 + \left|\frac{x-c}{a}\right|^{2b}} \tag{6.8}$$

Sigmoidal Membership Function:

A sigmoidal membership function has two parameters: a responsible for its slope at the crossover point $x = c$;

$$f(x,a,c) = \frac{1}{1 + e^{-a(x-c)}} \tag{6.9}$$

Gaussian Combination Membership Function:

A Gaussian combination membership function has two parameters: c responsible for its center, and σ responsible for its width;

$$f(x,\sigma,c) = e^{\frac{-(x-c)^2}{2\sigma^2}} \tag{6.10}$$

The function *gauss2mf* is a combination of two parameters. The first function, specified by $sig1$ and $c1$, determines the shape of the left-most curve. The second function specified by $sig2$ and $c2$ determines the shape of the right-most curve. Whenever $c1$ ¡ $c2$, the *gauss2mf* function reaches a maximum value of 1. Otherwise, the maximum value is less than one. The parameters are listed in the order: $[sig1, c1, sig2, c2]$.

Two Sigmoidally Shaped Membership Functions:

A Gaussian combination membership function has two parameters: c responsible for its center, and σ responsible for its width;

$$f(x,a,c) = \frac{1}{1 + e^{-a(x-c)}} \tag{6.11}$$

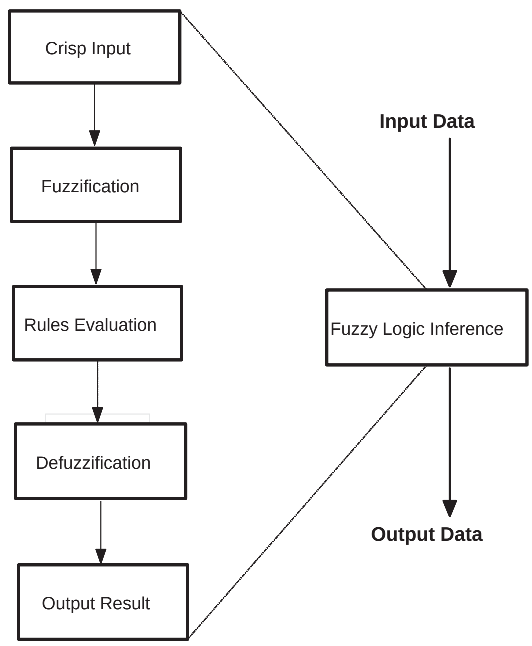

Figure 6.3: Fuzzy Logic Inference Algorithm.

psigmf is simply the product of two such curves plotted for the values of the vector x $f1(x; a1, c1)$ x $f2(x; a2, c2)$. The parameters are listed in the order $[a1c1a2c2]$.

6.3.3 Components of a Fuzzy System

The general Fuzzy Logic system structure can be summarised as follows; input data is first passed to a fuzzy algorithm, the fuzzy algorithm maps and compares the inputs with some desired data, and the output is generated. The design data of a fuzzy system is contained in the membership function (or fuzzy sets).

6.3.4 Inference Algorithm

The membership function is a graphical representation of the magnitude of participation of each input. It associates a weighting with each of the inputs that are processed to determine an output. This type of structure forms the basis of the work carried out in Section 6.7 and Chapter 7.

The most important part of a Fuzzy Logic system is the fuzzy inference algorithm. This process of fuzzy reasoning is combined into what is called a Fuzzy Inferencing System (FIS). It is comprised of three steps that process system inputs into appropriate system outputs. These steps are: *Fuzzification*, *Rule Evaluation*, and *Defuzzification* (see Figure 6.3).

Fuzzification

This is the first step in the fuzzy inferencing process. It involves input processing, which makes the data readable to the system. This is the transformation step where crisp inputs are transformed into fuzzy inputs. Crisp inputs are exact inputs measured by sensors and passed into the control system for processing. Each crisp input that is to be processed by the fuzzy inference system has membership functions or sets. The fuzzification process matches the input data with the conditions of the rules to determine how well the condition of each rule matches that particular input. There is a degree of membership for each linguistic term that applies to each input.

For a given input, calculate rule i strength as a fuzzy membership function $\mu_{A_i}(x)$.

$$W_i = \mu_{A_i}(x) \tag{6.12}$$

Rule Evaluation

Rule Evaluation consists of a series of **IF-THEN** rules statements. The fuzzy algorithm uses **NOT**, **OR** and **AND** operations combined with **IF-THEN** statements to combine input data with stored data to generate an output. The inference engine simply implies the combination of certain rules from a fuzzy rule base to get some output corresponding to the inferred rules. The most common operation types of inference used are listed below:

- The fuzzy **AND** connective

  The fuzzy intersection $\bigcap$ operator (**AND**) applied to two fuzzy sets A and B

with the membership functions $\mu_A(x)$ and $\mu_B(x)$ is:

$$\mu_{A \cap B}(x) = min \left\{ \mu_A(x), \mu_B(x) \right\}, x \in X. \tag{6.13}$$

- The fuzzy **OR** connective

  The fuzzy union $\bigcup$ operator (**OR**) applied to two fuzzy sets A and B with the membership functions $\mu_A(x)$ and $\mu_B(x)$ is:

$$\mu_{A \cup B}(x) = max \left\{ \mu_A(x), \mu_B(x) \right\}, x \in X. \tag{6.14}$$

- The fuzzy **NOT** operation

  The fuzzy complement operator (**NOT**) applied to the fuzzy sets A with the membership functions $\mu_A(x)$ is:

$$\overline{\mu_A}(x) = 1 - \mu_A(x), x \in X. \tag{6.15}$$

The output fuzzy sets of each rule are modified according to either the product rule:

$$\mu_i\left(y\right) = W_i \cdot \mu_{Bi}\left(y\right) \tag{6.16}$$

or the **min** inference rule:

$$\mu_i(y) = min(W_i \cdot \mu_{Bi}(y)) \tag{6.17}$$

Defuzzification

This involves the process of transforming the fuzzy outputs to crisp outputs. The preferred approach, called additive defuzzification, is to defuzzify each rule output

first, then combine the resulting crisp outputs.

$$y_i = \frac{\int y\mu_i\,(y)\,dy}{\int \mu_i\,(y)\,dy} = \frac{\int yw_i\mu_{Bi}\,(y)\,dy}{\int w_i\mu_{Bi}\,(y)\,dy} = \frac{\int y\mu_{Bi}\,(y)\,dy}{\int \mu_{Bi}\,(y)\,dy} \tag{6.18}$$

Next calculate the system output as:

$$y = \frac{\sum w_i y_i}{w_k} \tag{6.19}$$

The most commonly used techniques of fuzzy reasoning are the Mamdani direct method and the Takagi and Sugeno method. These are the most popular techniques due to their simplicity in representing most real world systems. The Mamdani direct method is commonly used because of ease of data manipulation, and it has the following premise and consequence structure:

IF x is A **AND** y is B, **THEN** z is C

The Takagi and Sugeno method is used with Neuro-fuzzy systems because it requires less computational time than the Mamdani method. It is very similar to the Mamdani direct method and only differs in the output. In this method, the output membership functions are replaced by a linear function as follows:

IF x is A **AND** y is B, **THEN** $z = a_x + b_y + c$.

These concepts and rules are explained in more details in Section 6.7 and Chapter 7.

6.4 Neural Networks

An Artificial Neural Network (ANN), usually called a "Neural Network" (NN), is a mathematical model inspired by the functioning of biological nervous systems, such as the brain. The key element of this information processing approach is that it consists

of an interconnected artificial neurons and processes information using a connectionist approach. Artificial Neural Networks (ANNs), like people, learn by example. ANNs are adaptive systems that change their behaviour based on external or internal information provided to the network during the learning phase.

Neural Networks are powerful techniques to solve real world problems. They have the ability to learn from experience in order to adapt to environmental changes. Furthermore, Neural Networks are able to deal with incomplete data very effectively in situations where is not possible to define all rules. The output of a Neural Network relies on the cooperation of the individual neurons within the network.

There are three major learning methods, each corresponding to a particular learning task. These are: *Supervised Learning*, *Unsupervised Learning* and *Reinforcement Learning*. All are types of machine learning.

- *Supervised learning* infers a function from supervised training data. Both the inputs and the outputs are provided.
- *Unsupervised learning* infers a function from unsupervised (unlabeled) data. This learning only provides the inputs, but not desired outputs.
- *Reinforcement learning* is learning by interacting with an environment. It allows machines and software agents to automatically determine the ideal behaviour within a specific context, in order to maximize its performance.

The term 'Neural Networks' has two distinct meanings:

- *Biological Neural Networks*: It is still unknown how the brain trains itself to process information; in general, the human nervous system is a very complex Neural Networks. Each neuron in the brain is composed of a body, one axon and multitude of dendrites.

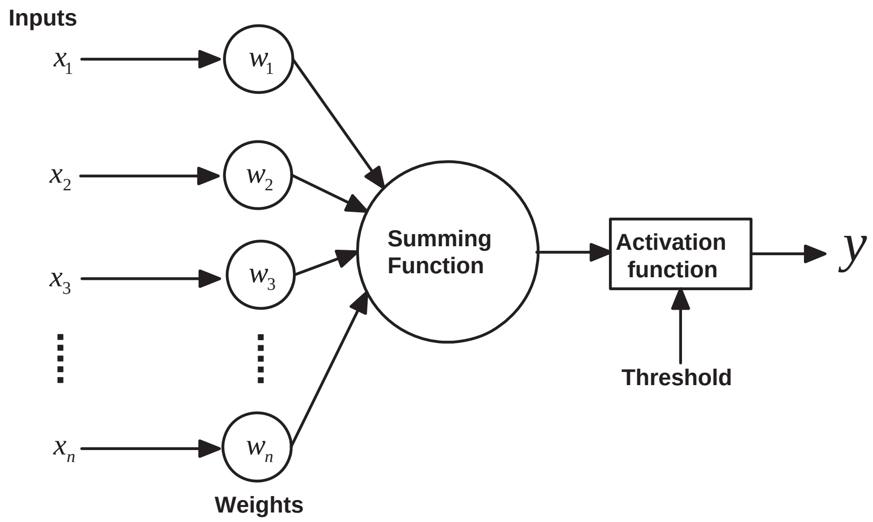

Figure 6.4: Artificial Neural Networks.

- *Artificial Neural Networks*: Represented by a set of nodes, they are often arranged in layers, connected by a set of weighted directed links. There are a wide variety of networks depending on the nature of information processing of individual nodes, the topology of the links, and the algorithm for adaptation of link weights (see Figure 6.4).

Some popular Artificial Neural Networks are:

- *Perceptron*: The earliest kind of Neural Network is a single-layer perceptron network, which consists of a single layer of output nodes; the inputs are fed directly to the outputs via a series of weights.

- *Multi-layered Perceptron (MLP)*: It has a layered architecture consisting of input, hidden and output layers. Each layer consists of a number of perceptrons. The output of each layer is transmitted to the input of nodes in other layers through weighted links.

- *Recurrent Neural Networks (RNN)*:): It is a class of Neural Networks consisting of a network of neurons with feedback connections.

- *Self-Organizing Maps (SOMs)*: SOMs or Kohonen networks have a grid topology, with unequal grid weights [87].

6.5 Types of Neuro-Fuzzy Systems

There are three main categories of 'Neuro-Fuzzy systems', i.e. combined Neural Networks and Fuzzy Logic: *Neural Fuzzy Systems*, *Fuzzy Neural Systems* and *Fuzzy-Neural Hybrid Systems*.

6.5.1 Neural Fuzzy Systems

Neural Fuzzy Systems are characterised by the use of Neural Networks to provide an automatic tuning method (Fuzzy Logic) without changing their functionality. One example of this approach is shown in Figure 6.5, which represents the use of Neural Networks for the membership function. In the training process, a Neural Network adjusts its weights in order to minimize the mean square error between the output of the network and the desired output. The weights of the Neural Network represent the parameters of the fuzzification function, fuzzy membership function, fuzzy rule and defuzzification function. This kind of combination is mostly used in control applications such as [148] [106] [125] [126] [159] [41] [67].

6.5.2 Fuzzy Neural Systems

The main idea of this approach is to fuzzify the elements of Neural Networks, using Fuzzy Logic (see Figure 6.6). This approach is mostly used in pattern recognition applications [92]. In this approach, the inputs are non fuzzy elements, but the weighting operations are replaced by membership functions. The result of each weighting operation is the membership value of the corresponding input in the fuzzy set.

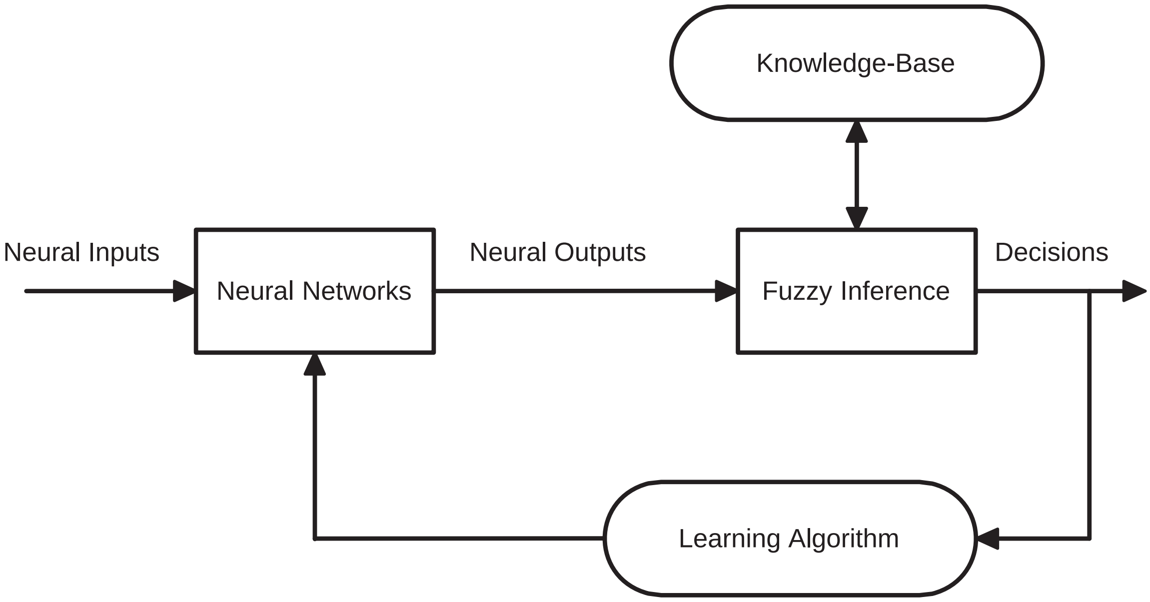

Figure 6.5: Neural Fuzzy System [55].

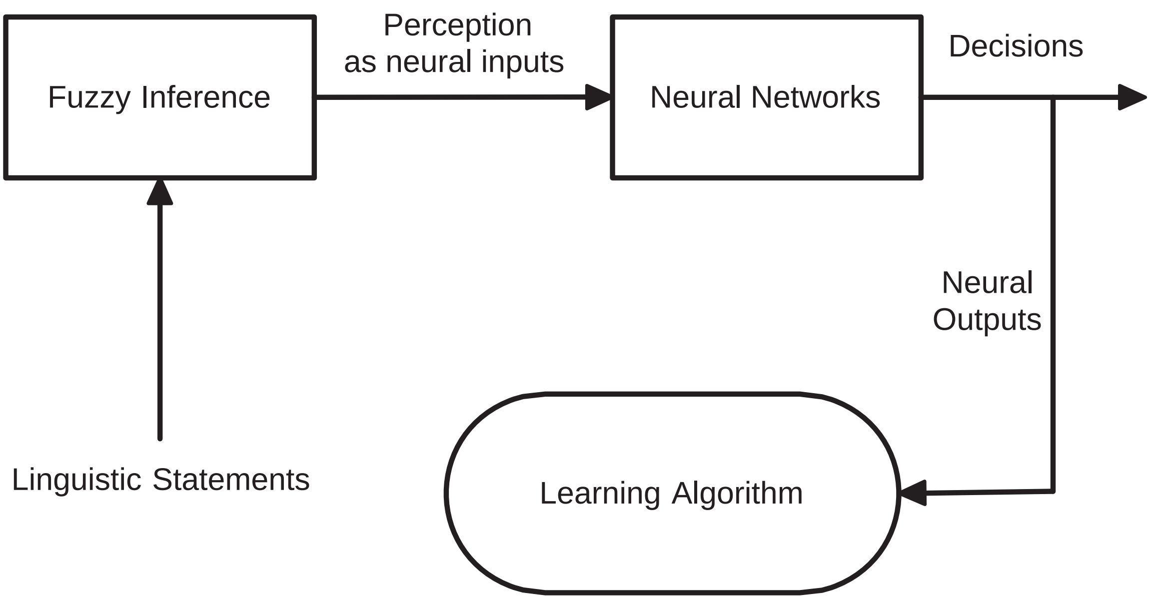

Figure 6.6: Fuzzy Neural System [55].

6.5.3 Hybrid Neuro-Fuzzy Systems

A Neuro-Fuzzy System is a combination of Neural Networks and fuzzy systems in such a way that the Neural Network is used to determine the parameters of the fuzzy system. The goal of the Neuro-Fuzzy approach is to use a fuzzy system to transform given inputs into a desired output throughout highly interconnected (Neural Networks) processing elements and information connections weights to map the numerical input into output.

A common way to apply a learning algorithm to a fuzzy system is to represent it in a special Neural Network like architecture. A learning algorithm, such as back propagation, is then used to train the system. However, Neural Network learning algorithms usually use a gradient descent method. This can not be applied directly to a fuzzy system, because the functions used to realise the inference process are usually not differentiable.

In this approach, both fuzzy and Neural Networks algorithms are used separately. Each one completes its own task in helping different functions in the system, incorporating and complementing each other in order to achieve a common goal. This kind of integration is suitable for both control and pattern recognition applications. The idea of a hybrid model is the interpretation of the fuzzy rule-base in terms of a Neural Network. In other words, fuzzy sets can be interpreted as weights, and the rules, input variables, and output variables can be represented as neurons. The learning algorithm results in Neural Networks. Examples of Hybrid Neuro-Fuzzy controllers include: GARIC [22], ANFIS [72], ARIC [21] and the NNDFR model [132]. These approaches consist of Neural Networks that are capable of learning from fuzzy sets.

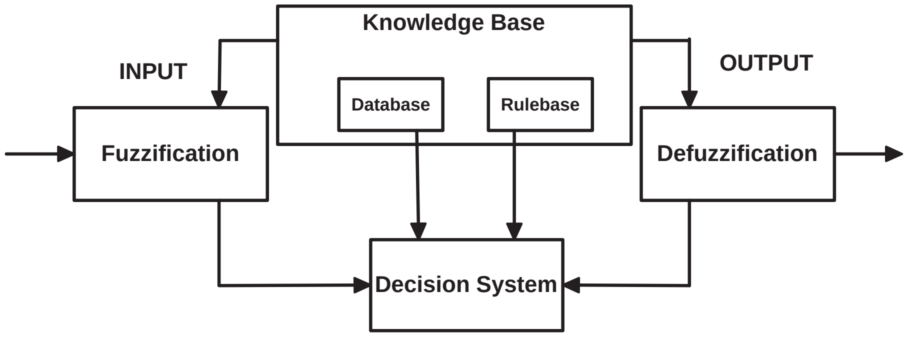

Figure 6.7: Fuzzy Inference System [72].

6.6 Fuzzy Inference Systems (FIS)

Fuzzy rules statements are expressions in the form of **IF** X **THEN** Z, where X and Z are labels of fuzzy sets [72] characterised by appropriate membership functions. Fuzzy **IF-THEN** rules are often used to capture the imprecise conditions of reasoning that play an important role in the process of decision making in an environment of uncertainty.

Fuzzy Inference Systems (FIS) are also known as fuzzy-rule-based systems, fuzzy models, or fuzzy controllers. Basically a fuzzy inference system is composed of five functional blocks (see Figure 6.7):

- a rule base containing a number of fuzzy **IF-THEN** rules;
- a database which defines the membership functions of the fuzzy sets used in the fuzzy rules;
- a decision-making unit which performs the inference operations on the rules;
- a fuzzification interface which transforms the crisp inputs into degrees of match with linguistic values; and
- a defuzzification interface which transforms the fuzzy results of the inference into a crisp output.

The use of Fuzzy Logic is based on many reasons and observations such as [3]:

- Easy to understand: The mathematical concepts behind fuzzy reasoning are very simple.
- Flexible: It is easy to add on more functionality without starting again from scratch.
- Tolerant of imprecise data: Very easy to represent imprecise data.
- Model nonlinear functions of arbitrary complexity: It is very easy to create a fuzzy system to match any set of input-output data using Fuzzy Logic Toolbox software.
- Fuzzy Logic can be built on top of the experience of experts.
- Fuzzy Logic can be blended with conventional control techniques.
- Fuzzy Logic is based on natural language.

The basis of Fuzzy Logic is human communication. Because Fuzzy Logic is built on the structures of qualitative description used in everyday language, Fuzzy Logic is easy to use [117].

6.7 ANFIS Architecture

The Adaptive Neuro-Fuzzy Inference System (ANFIS) technique was originally presented by Jang in 1993 [72]. ANFIS is a simple data learning technique that uses Fuzzy Logic to transform given inputs into a desired output throughout highly interconnected Neural Networks processing elements and information connections, which are weighted to map the numerical inputs into output.

ANFIS combines the benefits of the two machine learning techniques (Fuzzy Logic and Neural Networks) into a single technique [72]. An ANFIS works by applying Neural Networks learning methods to tune the parameters of a Fuzzy Inference System (FIS). There are several features that enable ANFIS to achieve great success [70] [71]:

- It refines fuzzy **IF-THEN** rules to describe the behaviour of a complex system;
- It does not require prior human expertise;
- It is easy to implement;
- It enables fast and accurate learning;
- It offers the desired data set, a greater choice of membership functions to use, strong generalisation abilities, and excellent explanation facilities through fuzzy rules;
- It is easy to incorporate both linguistic and numeric knowledge for problem solving.

Different rules cannot share the same output membership function, which means that the number of membership functions must be equal to the number of rules. To present the ANFIS architecture, two fuzzy **IF-THEN** rules based on a first order Sugeno model are considered (see Figure 6.8):

$$\text{Rule}_{(1)}\text{: } \textbf{IF } (x \text{ is } A_1) \textbf{ AND } (y \text{ is } B_1)\text{, } \textbf{THEN } f_1 = p_1x + q_1y + r_1.$$

$$\text{Rule}_{(2)}\text{: } \textbf{IF } (x \text{ is } A_2) \textbf{ AND } (y \text{ is } B_2)\text{, } \textbf{THEN } f_2 = p_2x + q_2y + r_2.$$

where x and y are the inputs, A_i and B_i are the fuzzy sets, f_i are the outputs within the fuzzy region specified by the fuzzy rule. p_i, q_i and r_i are the design parameters that are determined during the training process. Figure 6.9 illustrates the reasoning mechanism for this Sugeno model, which is the basis of the ANFIS model.

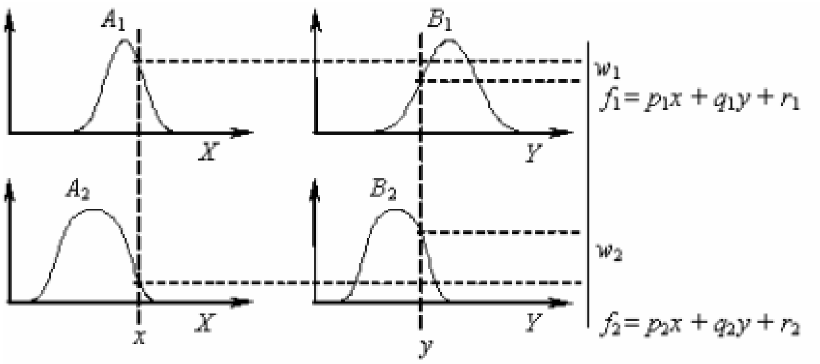

Figure 6.8: A Two-Input First Order Sugeno Fuzzy Model With Two Rules [72].

The ANFIS architecture to implement these two rules is shown in Figure 6.9. A circle indicates a fixed node, whereas a square indicates an adaptive node. ANFIS basically has a five-layer architecture. Each layer is explained in detail below.

In **Layer**$_{(1)}$, all the nodes are adaptive nodes. The outputs of Layer$_{(1)}$ are the fuzzy membership grade of the inputs, which are given by the following equations:

$$O_{1,i} = \mu_{Ai}(x),\ i = 1, 2 \tag{6.20}$$

$$O_{1,i} = \mu_{Bi-2}(y),\ i = 3, 4 \tag{6.21}$$

x and y are the inputs to node i, and A_i and B_i are the linguistic labels (high, low, etc) associated with this node function. $\mu_{Ai}(x)$ and $\mu_{Bi-2}(y)$ can adopt any fuzzy membership functions. For example, if the bell-shaped membership function is employed, $\mu_{Ai}(x)$ and is given by:

$$\mu_{Ai}(x) = \frac{1}{1 + [(\frac{x-c_i}{a_i})^2]b_i}\ ,\ i = 1, 2 \tag{6.22}$$

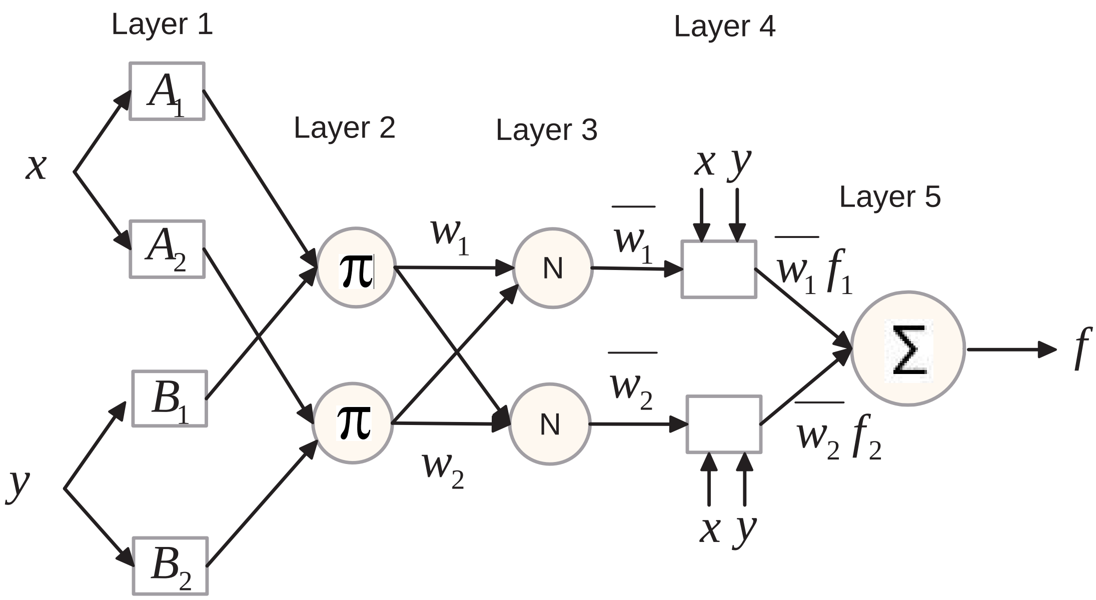

Figure 6.9: ANFIS Architecture [72].

or the Gaussian membership function by:

$$\mu_{Ai}(x) = exp\left[-\left(\frac{x - c_i}{a_i}\right)^2\right] \tag{6.23}$$

a_i, b_i and c_i are the parameters of the membership function.

In **Layer**$_{(2)}$, the nodes are fixed nodes. This layer involves fuzzy operators; it uses the **AND** operator to fuzzify the inputs. They are labeled with π, indicating that they perform as a simple multiplier. The output of this layer can be represented as:

$$O_{2,i} = w_i = \mu_{Ai}(x) * \mu_{Bi}(y) \ , \ i = 1, 2 \tag{6.24}$$

which are the so-called firing strengths of the rules.

In **Layer**$_{(3)}$, the nodes are also fixed nodes labeled by N, to indicate that they play a normalisation role to the firing strengths from the previous layer. The output of this

layer can be represented as:

$$O_{3,i} = \bar{w}_i = \frac{w_i}{w_1 + w_2} \ , \ i = 1, 2. \tag{6.25}$$

Outputs of this layer are called normalised firing strengths.

In **Layer**$_{(4)}$, the nodes are adaptive. The output of each node in this layer is simply the product of the normalised firing strength and a first order polynomial (for a first order Sugeno model). The output of this layer is given by:

$$O_{4,i} = \bar{w}_i f_i = \bar{w}_i(p_i x + q_i y + r_i), i = 1, 2. \tag{6.26}$$

$\bar{w}$ is the output of layer$_{(3)}$, and p_i, q_i, r_i are the consequent parameters.

In **Layer**$_{(5)}$, there is only one single fixed node labeled with $\sum$. This node performs the summation of all incoming signals. The overall output of the model is given by:

$$O_{5,i} = \sum_i \bar{w} f_i = \frac{\sum_i w_i f_i}{\sum_i w_i} \tag{6.27}$$

6.8 Hybrid Learning Algorithm

The learning algorithm for ANFIS is a hybrid algorithm, which is a combination between gradient descent and least squares method. In the forward pass of the hybrid learning algorithm, node outputs go forward until the Layer$_{(4)}$ and the consequent parameters are determined by the least-squares. In the backward pass, the error signals propagate backward and the premise parameters are updated by gradient descendent (see Table 6.1). The hybrid learning approach converges much faster, by reducing search space dimensions of the original Back-Propagation method [72]. The overall

output can be given by:

$$f = \frac{w_1}{w_1 + w_2} f_1 + \frac{w_2}{w_1 + w_2} f_2 \tag{6.28}$$

$$f = \bar{w}\,(p_1 x + q_1 y + r_1) + \bar{w}\,(p_2 x + q_2 y + r_2) \tag{6.29}$$

$$\begin{aligned} f = {} & (\bar{w}_1 x)\, p_1 + (\bar{w}_1 y)\, q_1 + (\bar{w}_1)\, r_1 + (\bar{w}_2 x)\, p_2 \\ & + (\bar{w}_2 y)\, q_2 + (\bar{w}_2)\, r_2 \end{aligned} \tag{6.30}$$

p_1, q_1, r_1, p_2, q_2 and r_2 are the linear consequent parameters. The least squares method is used to identify the optimal values of these parameters. When the premise parameters are not fixed, the search space becomes larger and the convergence of the training becomes slower.

It is noted that a hybrid algorithm of ANFIS combines two methods, the least squares method and the gradient descent method, to solve the problem of search space. The hybrid algorithm is composed of a forward and a backward pass as shown Table 6.1.

The least squares method (forward pass) is used to optimise the consequent parameters. The gradient descent method (backward pass) is used to adjust optimally the premise parameters. The output of the ANFIS is calculated by employing the consequent parameters found in the forward pass. The output error is used to adapt the premise parameters by means of a standard Back-Propagation algorithm. It has been proven that this hybrid algorithm is highly efficient in training the ANFIS systems [71].

	Forward Pass	Backward Pass
Premise parameters	Fixed	Gradient descent
Consequent parameters	Least-squares estimator	Fixed
Signals	Node outputs	Error signals

Table 6.1: The Two Passes in the Hybrid Learning Algorithm [70].

6.9 Summary

In this chapter an overview of Fuzzy Logic and Neural Networks was provided. Fuzzy Logic allows the representation of many imprecise inputs through simple and powerful process (**IF-THEN**) rules to produce some desired output. Neural Networks have the ability to learn from experience in order to adapt to the environmental changes through supervised and unsupervised learning. Combining Neural Networks and Fuzzy Logic brings together the adaptive abilities of Neural Networks with the reasoning capabilities of Fuzzy Logic. This combination is better known as a Neuro-Fuzzy system. ANFIS has numerous practical applications in the areas of control, prediction and inference. It is commonly used for nonlinear function modelling and chaotic time series prediction. It has also been used for many applications such as system design, database management, rainfall forecasting and forecasting of water resources. ANFIS is selected to solve the problem of continuous changes in the mobile learning environment to deliver adapted learning content based on learner's needs.

Chapter 7

ANFIS Modelling

7.1 Outline

The methodology of the research will employ a two-phased approach. The first phase is the building of the theoretical framework, to represent the mobile learning contexts and adaptation process which have been explained in Chapters 3, 4 and 5. The second phase (practical) is explained in this chapter. This chapter will use Adaptive Neuro-Fuzzy Inference System (ANFIS) modelling.

The ANFIS modelling is divided into two steps: Fuzzy Inference Systems then Adaptive Neuro-Fuzzy Inference System . Fuzzy inference system is the process of formulating the mapping from a given input to an output using Fuzzy Logic. There are two types of fuzzy inference systems that can be implemented in the Fuzzy Logic Toolbox: Mamdani type and Sugeno type. Using a given input/output data set, the toolbox function ANFIS constructs a Fuzzy Inference System whose membership function parameters are tuned using either a Back-Propagation algorithm or in combination with a Least Squares type of method.

7.2 Introduction

Humans make use of their surrounding environment through interaction with other humans; they interpret their current context and react to it. For example, when a student interacts with a peer, he automatically observes his peer's tone of voice and responds in an appropriate manner. Devices such as computers and mobile phones are not able to interpret the surrounding context in the same way as humans during the interaction process. These types of devices cannot make use of available information in a transparent way; therefore this context information must be explicitly supplied to these devices to allow the devices to respond appropriately.

The term 'context-awareness system' refers to a system that can use, extract and

interpret context information and adapt its services to the current context. While the concept is simple, the development of such a system is challenging due to the complexity of capturing, representing and processing contextual information. In addition to the difficulties associated with obtaining context information, context-aware systems must reason and make decisions about how to process the context information to deduce useful and meaningful services. Context reasoning is the most challenging issue in the area of adaptation and personalisation due to the nature of context (imprecise context).

The ANFIS modelling is divided into two steps:

- Fuzzy Inference Systems (FIS)
- Adaptive Neuro-Fuzzy Inference System (ANFIS)

The next section will explain FIS and ANFIS using the Sugeno type FIS structure. This includes creating, generating, training, testing, and validation of the FIS and ANFIS structure.

7.3 FIS Editor GUI

Fuzzy Inference System (FIS) is the process of formulating the mapping from a given input to an output using Fuzzy Logic. There are two types of FIS that can be implemented in the Fuzzy Logic Toolbox: Mamdani type and Sugeno type. In the Fuzzy Logic Toolbox, there are five steps of the fuzzy inference process:

- fuzzification of the input variables,
- application of the fuzzy operator (**AND** or **OR**) in the antecedent,
- implication from the antecedent to the consequent,

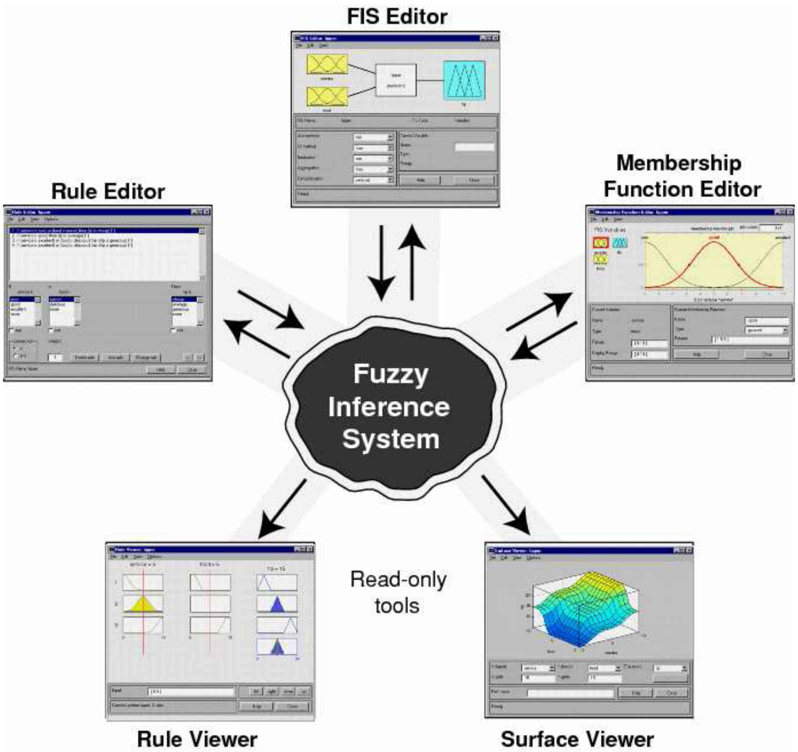

Figure 7.1: Fuzzy Logic Graphical User Interface (GUI) [3].

- aggregation of the consequents across the rules, and
- defuzzification.

There are five primary GUI tools for building, editing, and observing FISs in the Fuzzy Logic Toolbox package (see Figure 7.1):

- FIS Editor,
- Membership Function Editor,
- Rule Editor,
- Rule Viewer, and
- Surface Viewer.

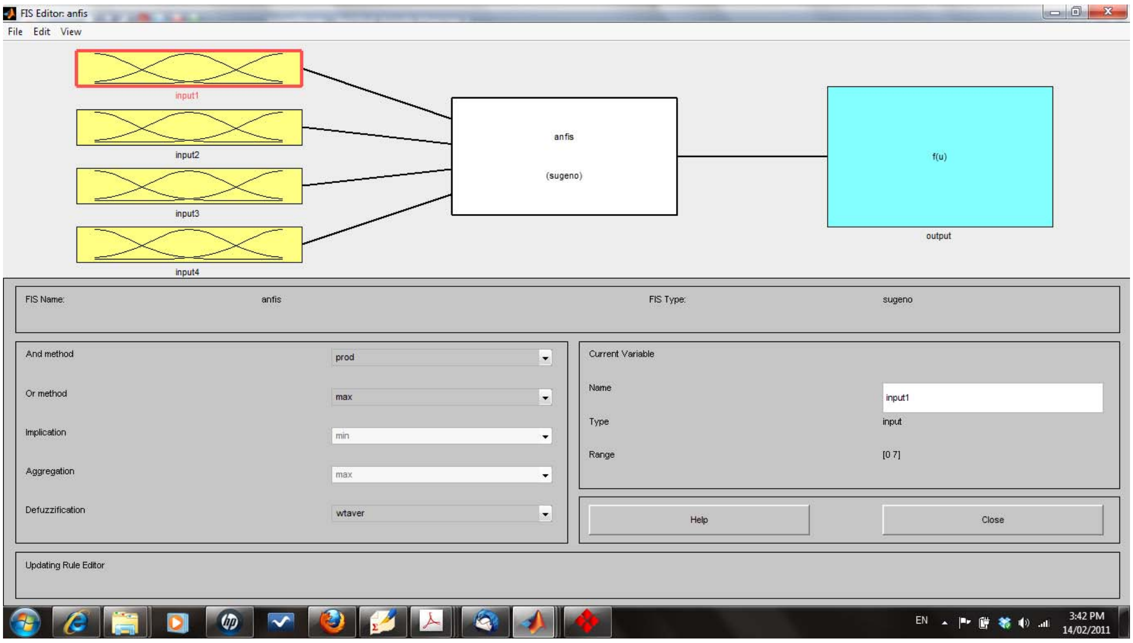

Figure 7.2: FIS Editor.

7.3.1 FIS Editor

The FIS Editor displays general information about a FIS. It is a simple diagram that shows the names of each input variable on the left, and those of each output variable on the right. The sample membership functions shown in the boxes are icons only, and do not depict the actual shapes of the membership functions (see Figure 7.2).

7.3.2 Membership Function Editor

The Membership Function Editor is used to define the shapes of all the membership functions associated with each variable. The Membership Function Editor is the tool that allows the user to display and edit all of the membership functions associated with all of the input and output variables for the entire FIS (see Figure 7.3).

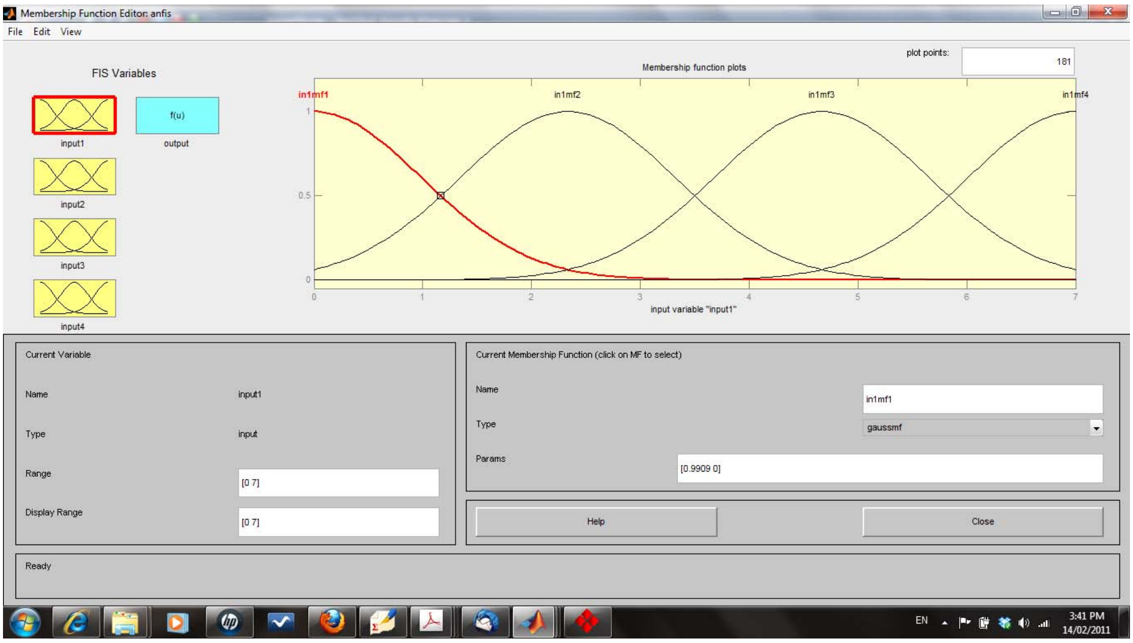

Figure 7.3: Membership Function Editor.

7.3.3 Rule Editor

The Rule Editor allows the user to construct the rule statements automatically, by clicking on and selecting one item in each input variable box, one item in each output box, and one connection item (see Figure 7.4). Rules may be changed, deleted or added by clicking the appropriate button.

7.3.4 Rule Viewer

The Rule Viewer allows the user to interpret the entire fuzzy inference process at once. The Rule Viewer also shows how the shape of certain membership functions influences the overall result (see Figure 7.5). The Rule Viewer shows one calculation at a time and in great detail. It presents a micro view of the FIS.

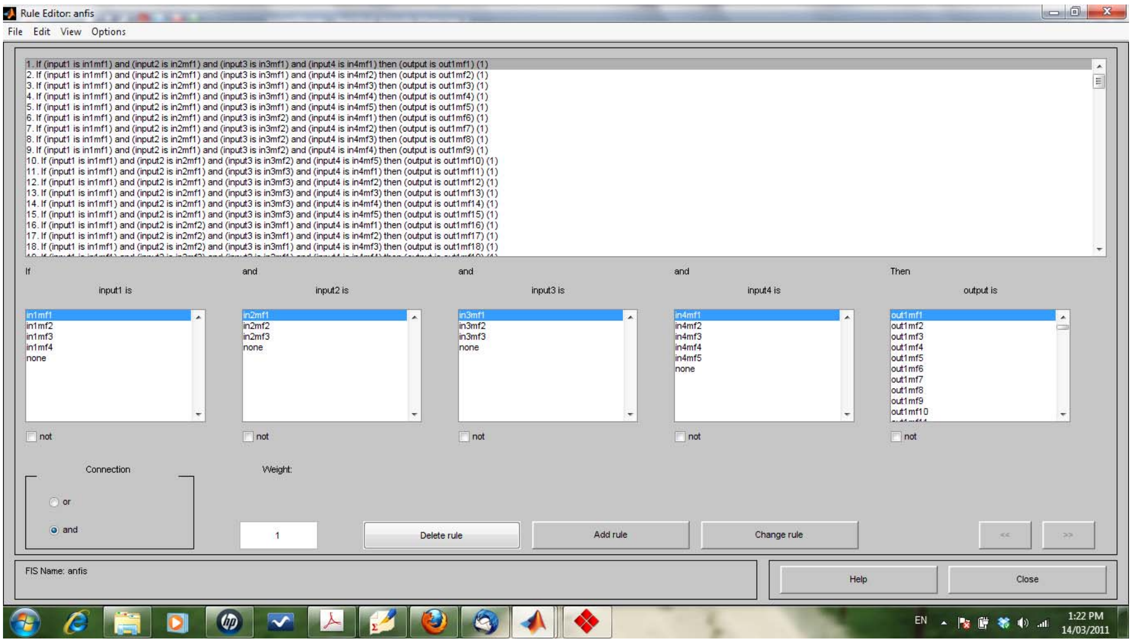

Figure 7.4: Rule Editor.

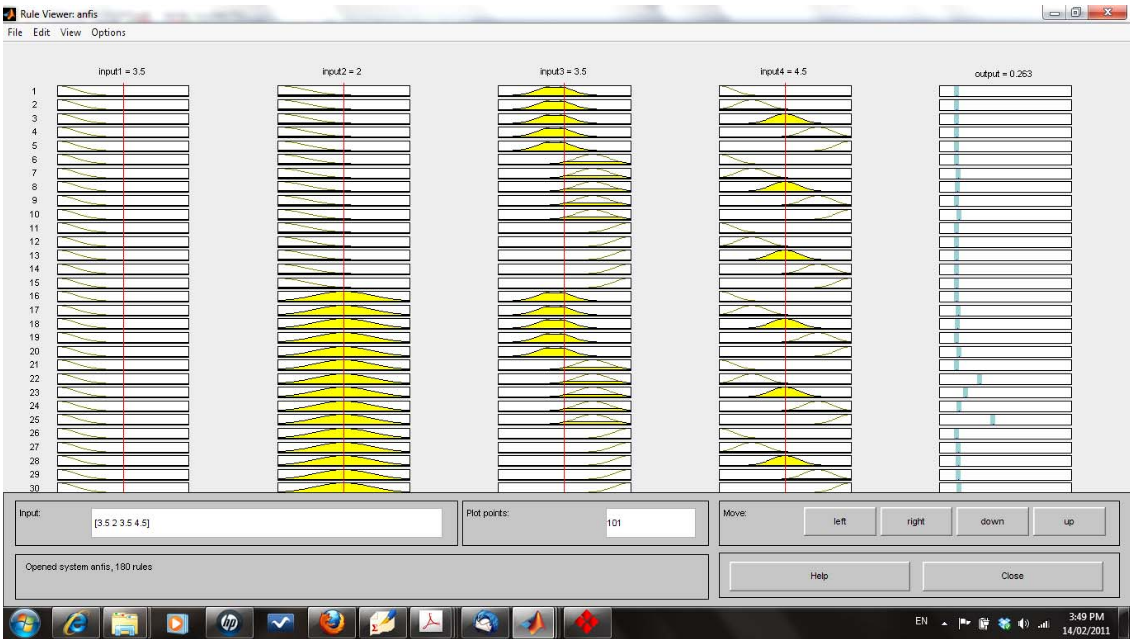

Figure 7.5: Rule Viewer.

7.4 ANFIS Editor GUI and Training

ANFIS derives its name from Adaptive Neuro-Fuzzy Inference Systems. Using a given input/output data set, the *anfis* function is used to construct a Fuzzy Inference System (FIS) that has membership function parameters tuned using either a Back-Propagation algorithm or in combination with a Least Squares method.

ANFIS only supports Sugeno type system, and must have the following constrains [3]:

- Be first or zero order Sugeno type systems,
- Have a single output,
- Output membership functions must be of same type and either linear or constant,
- No rule sharing, and
- Have unity weight for each rule.

An error will occur if the FIS structure does not comply with these constraints.

The ANFIS Editor GUI menu bar can be used to load a FIS training initialisation, save the trained FIS, and open a new Sugeno type system or any of the other GUIs to interpret the trained FIS model. Any data set that is loaded into the ANFIS Editor GUI (or that is applied to the command-line function ANFIS) must be in a matrix form, with the input data arranged as vectors in all but the last column. The output data must be in the last column. A sample of an ANFIS Editor GUI with inputs is shown in Figure 7.6.

7.4.1 Loading the Data

The training data set contains the desired input/output data set of the targeted system. Training a FIS begins with loading the training data set that contains the desired

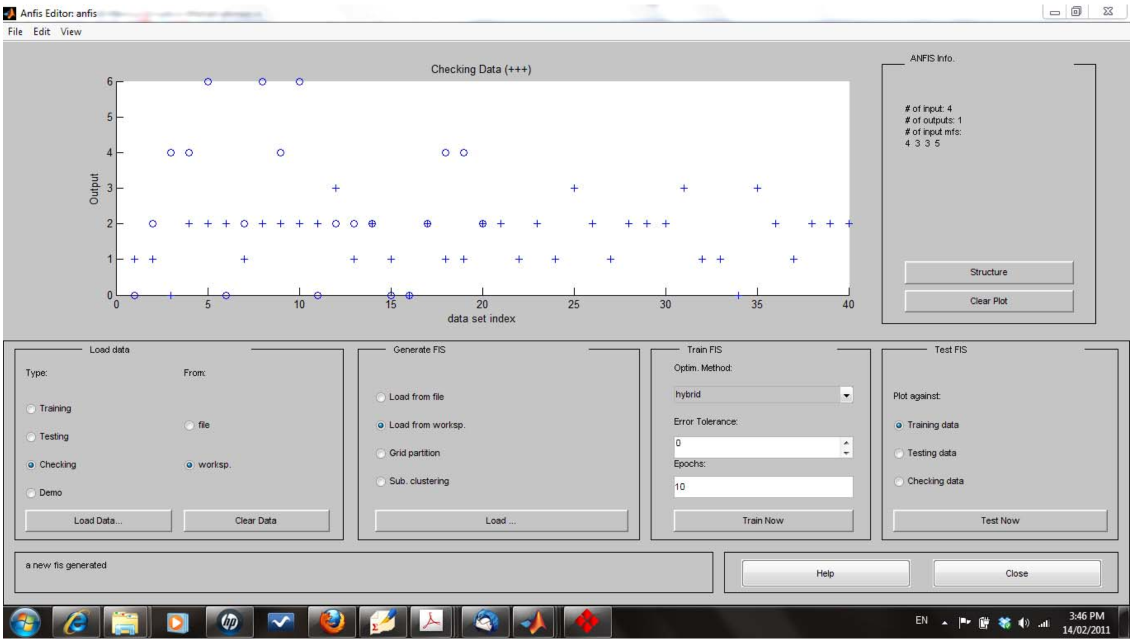

Figure 7.6: ANFIS Model Structure.

input/output data of the system to be modelled. The data set loaded must be in array form as column vectors, with the last column representing the output data. It can also be loaded as testing and checking data in the GUI. After loading the training data, a plot is displayed.

7.4.2 Generating the Initial FIS Structure

It is necessary to initialise the FIS structure before commencing the FIS training. Figure 7.7 shows the newly created Sugeno type FIS structure with four inputs:

- Learner Location (**LL**),
- Network Bandwidth (**NB**),
- Battery Life (**BL**), and
- Software Capability (**SC**).

The output is labeled as Adapted Learning content (**ALC**).

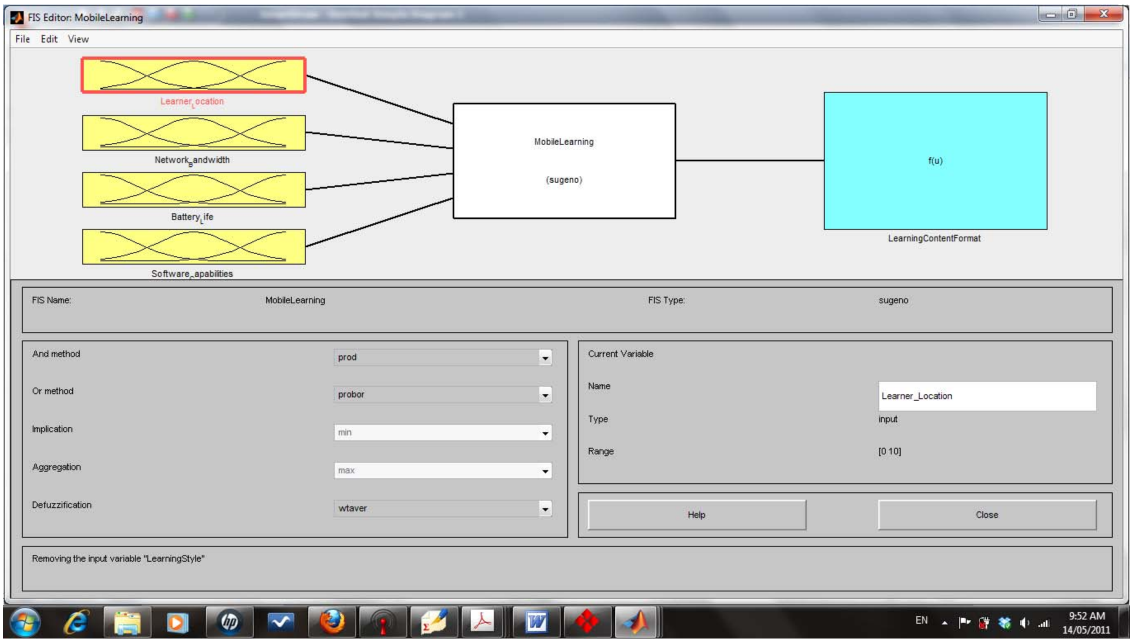

Figure 7.7: Sugeno Type FIS Structure.

To generate the initial FIS model, two partitioning techniques are available:

- *Grid partition*: Generates a single-output Sugeno type FIS by using grid partitioning on the data.

- *Sub clustering*: Generates an initial model for ANFIS training by first applying subtractive clustering on the data.

The FIS structure generation window appears after the training data is loaded (see Figure 7.8). It asks to specify the number of membership functions for each input, membership functions type, and output membership function type. Figure 7.9 shows the ANFIS model structure with rules mapping the output after specifying the initial conditions of the created Sugeno type FIS system. It depicts:

- 4 black circles - the given inputs (**LL**, **NB**, **BL** and **SC**),

- 9 white circles - the self generated input membership functions, '3*3*3*5' for four inputs,

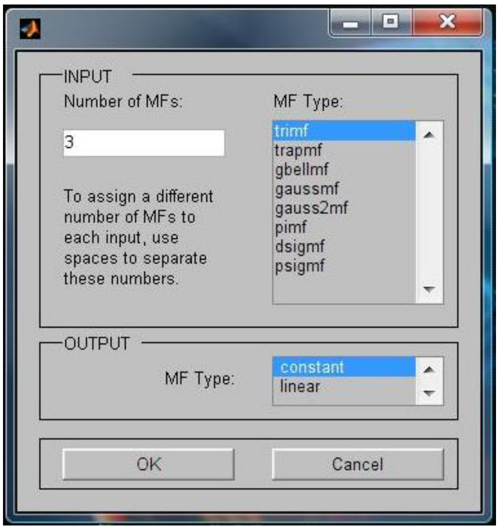

Figure 7.8: The FIS Structure Generation Window.

- 135 blue circles - the self generated rules,
- 135 white circles - the self generated output membership functions,
- 1 white circle - the mapping point for all the output membership functions, and
- 1 black circle - the output (**ALC**).

7.4.3 Model Training

After loading the training data and generating the initial FIS structure, the FIS training can begin (see Figures 7.10 and 7.11). The optimisation methods train the membership function parameters to emulate the training data. There are two different optimisation methods: Hybrid and Back-Propagation. The number of training Epochs and the training error tolerance are used to set the stopping criteria for training. The training process stops whenever the maximum Epoch number is reached or the training error goal is achieved.

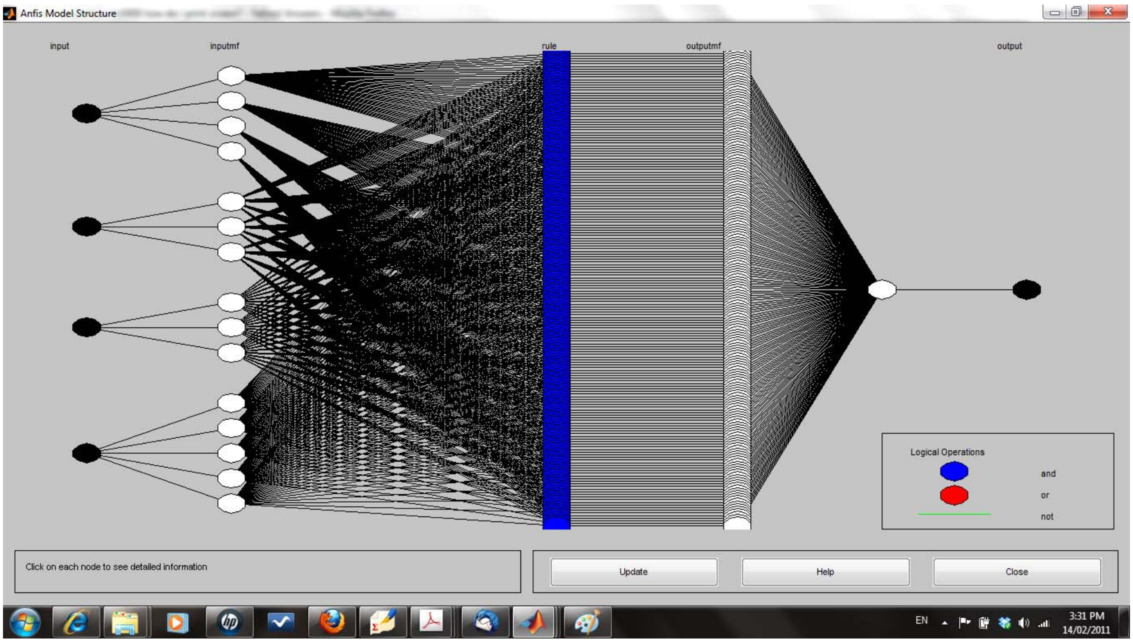

Figure 7.9: The ANFIS Structure With Rules Mapping Output.

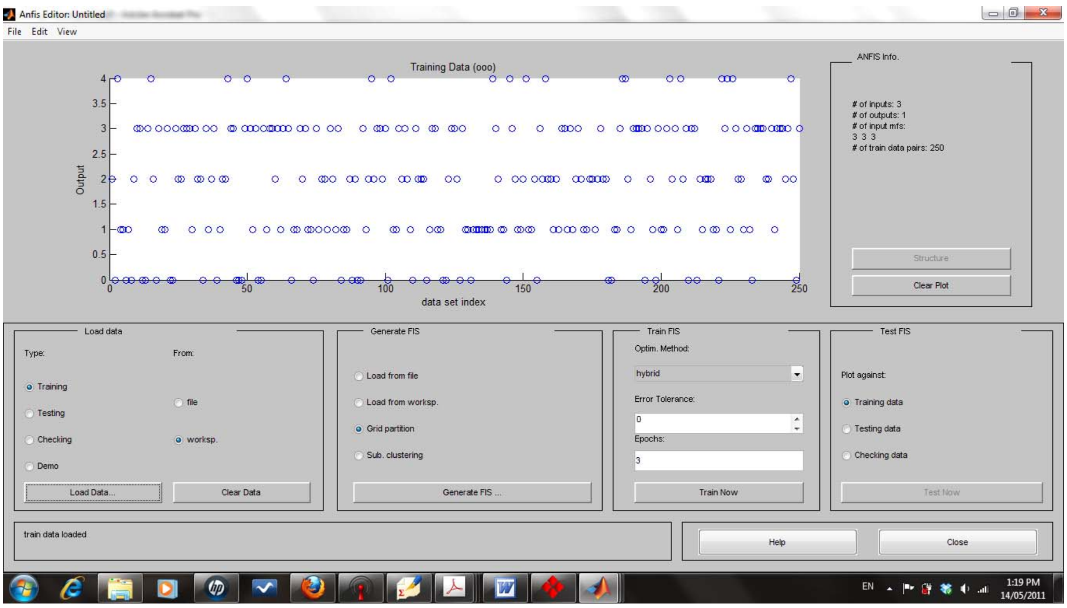

Figure 7.10: The Training Data Loaded in the ANFIS Structure.

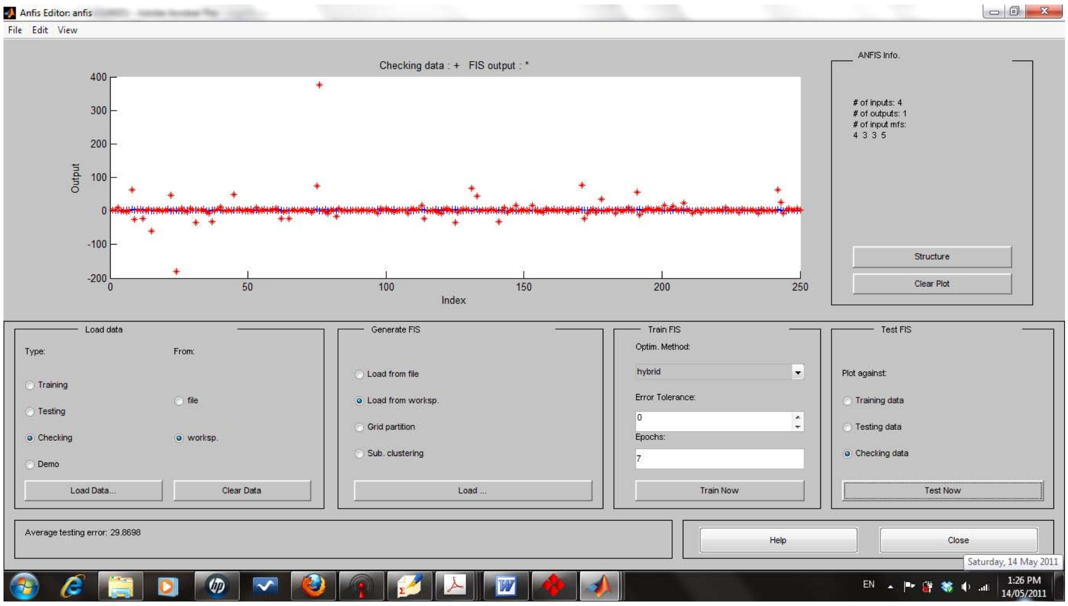

Figure 7.11: The Training Data Against the FIS Output.

7.4.4 Model Validation

After the FIS is trained, validation of the model is carried out using testing or checking data. This process is accomplished with the ANFIS Editor GUI using the testing and checking data set. The checking data set is used to control the potential for the model over-fitting the data. When the checking and training data are presented to ANFIS, the FIS model that has parameters associated with the minimum checking data model error is selected.

7.5 ANFIS Architecture for Adaptive Mobile Learning

ANFIS is an intelligent Neuro-Fuzzy technique used for the modelling and control of ill-defined and uncertain systems. ANFIS is based on the input/output data pairs of the system under consideration. ANFIS is selected to solve the problem of continuous

changes in a mobile learning environment to deliver adapted learning content. The proposed ANFIS model can be successfully used for modelling the learner context. The steps taken to take apply ANFIS for learner modelling are:

- defining input and output values,
- defining fuzzy sets for input values,
- defining fuzzy rules and
- creating and training the neural network.

The goal of thesis thesis is to construct a system that will help learners to accomplish learning activities by delivering adapted learning content based on the learner's current contexts. The process of learner modelling is composed of three steps:

- gathering data related to the learner,
- creating the learner model, and
- updating the learner model.

As reviewed in Chapter 6, hybrid learning is the main focus of the ANFIS structure. A Sugeno type fuzzy model provides a systematic way to generate fuzzy rules from a given input/output data set. For this research, an example of three inputs represent a mobile learning context scenario (refer to Section 5.4) will be used to explain the ANFIS modelling structure. The ANFIS structure proposed here contains three inputs, two membership functions for each input, eight fuzzy rules and the learning content format as output.

The three input conditions are:

- Learner Location (**LL**). **LL** has two membership functions, represented as *Class* and *outdoor*.

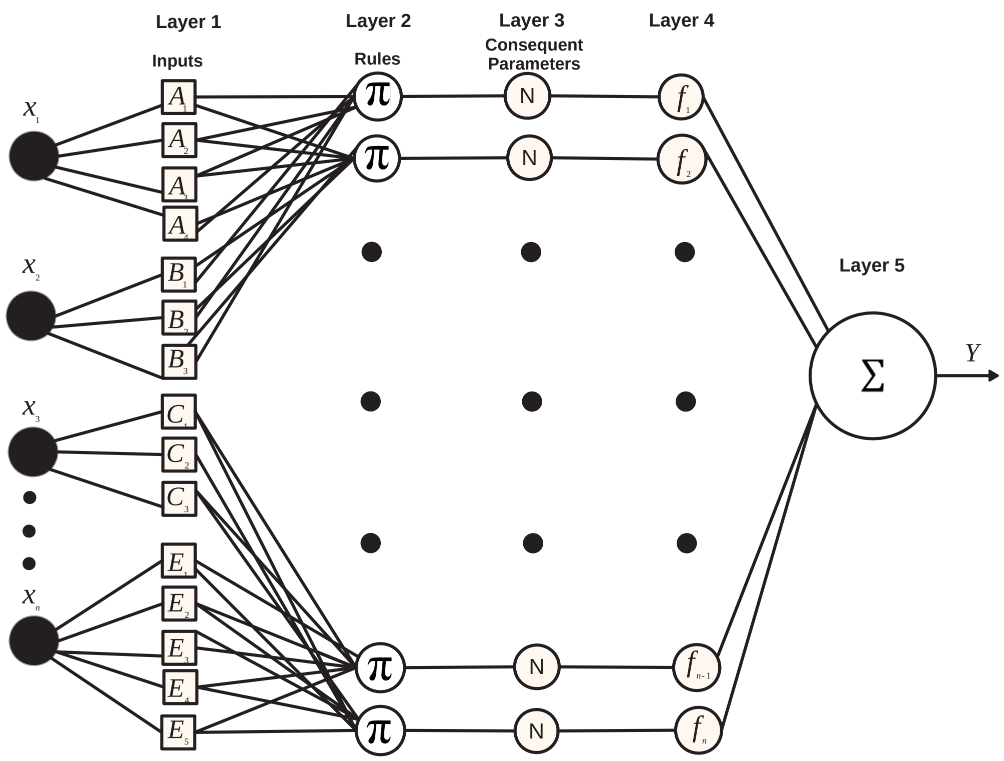

Figure 7.12: Adaptive Mobile Learning Using ANFIS Structure.

- Network Bandwidth (**NB**). **NB** has two membership functions, represented as *Low* and *High*.

- Battery Life (**BL**). **BL** has two membership functions, represented as *Low* and *Full*.

The output conditions are: *Text*, *PDF*, *Audio* or *Video*.

The ANFIS model for adaptive mobile learning used in this research is depicted in Figure 7.12. The ANFIS model is a multi-input single output system. In the network structure of the ANFIS model described above (see Figure 7.12 [9]), where n inputs are fed to the network from the left end and propagated through five layers until the output is generated at the right end.

Figure 7.12 depicted the ANFIS structure for n inputs with different types and number of membership functions. The ANFIS structure proposed here contains three

inputs (x_1, x_2 and x_3), two membership functions for each input (A_1, A_2, B_1, B_2, C_1, C_2), eight fuzzy rules and Output (Y). Fuzzy rules are carefully selected by a human expert or system designer, for example:

IF (Learner Location = *Outdoor*) **AND** (Network Bandwidth = *Low*) **AND** (Battery Life = *Full*) **THEN** (Learning Content Format = *PDF*).

As a result of the learner location being variable (i.e. the learner is moving) and the network bandwidth being very low, it is better to deliver the learning content in PDF format to save time, quality and to minimise learner boredom which may cause the learner to terminate the application.

The fuzzy rules in the ANFIS model are shown in the following form:

Rule 1: **IF** (x_1 is A_1) **AND** (x_2 is B_1) **AND** (x_3 is C_1), **THEN**
$$(f_1 = a_1x_1 + b_1x_2 + c_1x_3 + d_1)$$

Rule 2: **IF** (x_1 is A_1) **AND** (x_2 is B_1) **AND** (x_3 is C_2), **THEN**
$$(f_2 = a_2x_1 + b_2x_2 + c_2x_3 + d_2)$$

Rule 3: **IF** (x_1 is A_1) **AND** (x_2 is B_2) **AND** (x_3 is C_1), **THEN**
$$(f_3 = a_3x_1 + b_3x_2 + c_3x_3 + d_3)$$

Rule 4: **IF** (x_1 is A_1) **AND** (x_2 is B_2) **AND** (x_3 is C_2), **THEN**
$$(f_4 = a_4x_1 + b_4x_2 + c_4x_3 + d_4.)$$

Rule 5: **IF** (x_1 is A_2) **AND** (x_2 is B_1) **AND** (x_3 is C_1), **THEN**
$$(f_5 = a_5x_1 + b_5x_2 + c_5x_3 + d_5.)$$

Rule 5: **IF** (x_1 is A_1) *AND* (x_2 is B_1) **AND** (x_3 is C_1), **THEN**
$$(f_1 = a_1x_1 + b_1x_2 + c_1x_3 + d_1.)$$

Rule 6: **IF** (x_1 is A_2) **AND** (x_2 is B_1) **AND** (x_3 is C_2), **THEN**
$$(f_6 = a_6x_1 + b_6x_2 + c_6x_3 + d_6.)$$

Rule 7: **IF** (x_1 is A_2) **AND** (x_2 is B_2) **AND** (x_3 is C_1), **THEN**

$$(f_7 = a_7x_1 + b_7x_2 + c_7x_3 + d_7.)$$

Rule 8: **IF** (x_1 is A_2) **AND** (x_2 is B_2) **AND** (x_3 is C_2), **THEN**

$$(f_8 = a_8x_1 + b_8x_2 + c_8x_3 + d_8.)$$

where x_1, x_2 and x_3 are the inputs, A_i, B_i and $C_i, i = 1, 2$ are the fuzzy membership functions, f_i are the outputs within the fuzzy region specified by the fuzzy rule. The linear parameters $(a_i, b_i, c_i, d_i, i = 1, 2, ...8)$ in the rules set are determined in the forward pass, where the output of each layer is propagated forward layer by layer, until it reaches Layer$_{(4)}$ using recursive least square method; and in Layer$_{(5)}$, the overall output is obtained by summing all the normalised nodes output.

In Figure 7.12, the detailed equations and relationships among the layers for the above mobile learning context scenarios are specified in Section 6.7 and can be summarised as follows: Layer$_{(1)}$ (Equation 6.20, 6.21, 6.22 and 6.23), Layer$_{(2)}$ (Equation 6.24), Layer$_{(3)}$(Equation 6.25), Layer$_{(4)}$ (Equation 6.26) and Layer$_{(5)}$ (Equation 6.27).

7.5.1 Adaptive Mobile Learning Workflow

This project seeks to develop an adaptive mobile learning system for delivering learning content according to learners' needs, and to create a coordinated and interactive system that helps learners enhance their learning experience. Before creating the FIS and ANFIS that represent the mobile learning scenarios, the full workflow of mobile learning (or what happens when we start using the application) must be understood.

The reasoning engine consists of two stages: Fuzzy Logic and Neural Networks. The inputs to the adaptation engine include the *Learner Status*, the *Situation Status*, the *Knowledge and Shared Properties Status* and the *Educational Activity Status* in the form of Learner Profile Representation. In this section we describe the workflow

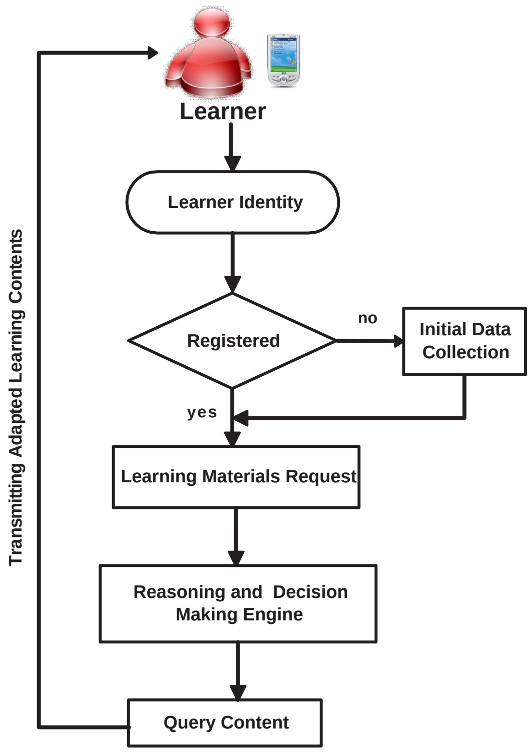

Figure 7.13: Mobile Learning Application Workflow.

of the mobile learning application (see Figure 7.13), which can be summarised in the following steps:

- The application begins when the learner carries out the learning activity by interacting with the application through the selection of some actions presented on the learner's device interface. If using the application for the first time, the learner will be asked to complete some forms to record personal information and other data through device sensors.

- If the learner is registered (i.e. has used the application before), then the learner is willing to participate and wants to begin the activity. The system issues a start up application to indicate that the learner wants to undertake a learning activity. Learner data is gathered to construct the learner model.

- The reasoning engine transforms the content from one state to another in order

to meet the constraints of the learner context. The system observes learner behaviour and preferences, and recommends educational materials that are similar to what have been chosen in the past.

- The reasoning engine selects the proper media type for the learning content based on the learner constraints, and transmits the adapted content to the learner's interface.

7.5.2 Inputs selection for the System

For mobile learning modelling, it is common to have tens of potential inputs for the system to be created. An excessive number of inputs will increase the computation time necessary for building the model using ANFIS. As a result, it is essential to exercise input selection to prioritise all system inputs and use them accordingly. To build an accurate model for prediction, significant inputs must be selected. In the literature, many techniques have been proposed for input selection. These include ARD [93], CART [26], and the δ-test [111] which determines dependencies within data in order to select relevant inputs.

In [127], the author listed some practical considerations for input selection:

- Remove noise/irrelevant inputs.
- Remove inputs that depend on other inputs.
- Make the underlying model more concise and transparent.
- Reduce the time for model construction.

The method presented in [127] takes advantage of the ANFIS structure [71]. It is based on the assumption that the ANFIS model with the smallest Root Mean Square Error (RMSE) after a small number of Epochs has a greater potential of achieving a lower

RMSE when given more Epochs of training. Hence, if there are ten candidate inputs and it is necessary to determine the three most influential inputs, then ($C_3^{10} = 120$) ANFIS models can be constructed, and the model with the smallest RMSE will be chosen.

In the case of mobile learning, hundreds of potential inputs can be selected. If there are seven inputs, the four most influential inputs can be determined. At each time, the RMSE is calculated for combinations of variables ($C_4^7 = 35$) and the minimum RMSE is recorded. The algorithm is run for one input at a time, then two inputs at a time, and so on, until the algorithm reaches seven inputs at a time.

7.5.3 ANFIS System Training Process

The ANFIS system training methodology is summarised in Figure 7.14. The process begins by obtaining a training data set (input/output data pairs) and checking data sets. The training data is a set of input and output vectors. Two vectors are used in order to train the ANFIS systems: the input vector and the output vector. The training data set is used to find the premise parameters for the membership functions. A threshold value for the error between the actual and desired output is determined.

The consequent parameters are found using the Least Squares method. If this error is larger than the threshold value, then the premise parameters are updated using the gradient decent method. The process is terminated when the error becomes less than the threshold value. The checking data set is then used to compare the model with the actual system [72].

The learning rule of ANFIS training is a hybrid learning which combines the Gradient Descent and the Least Squares method. The aim of using ANFIS for adaptive mobile learning is to achieve the best performance possible. ANFIS training starts by creating a set of suitable training data in order to be able to train the Neuro-Fuzzy

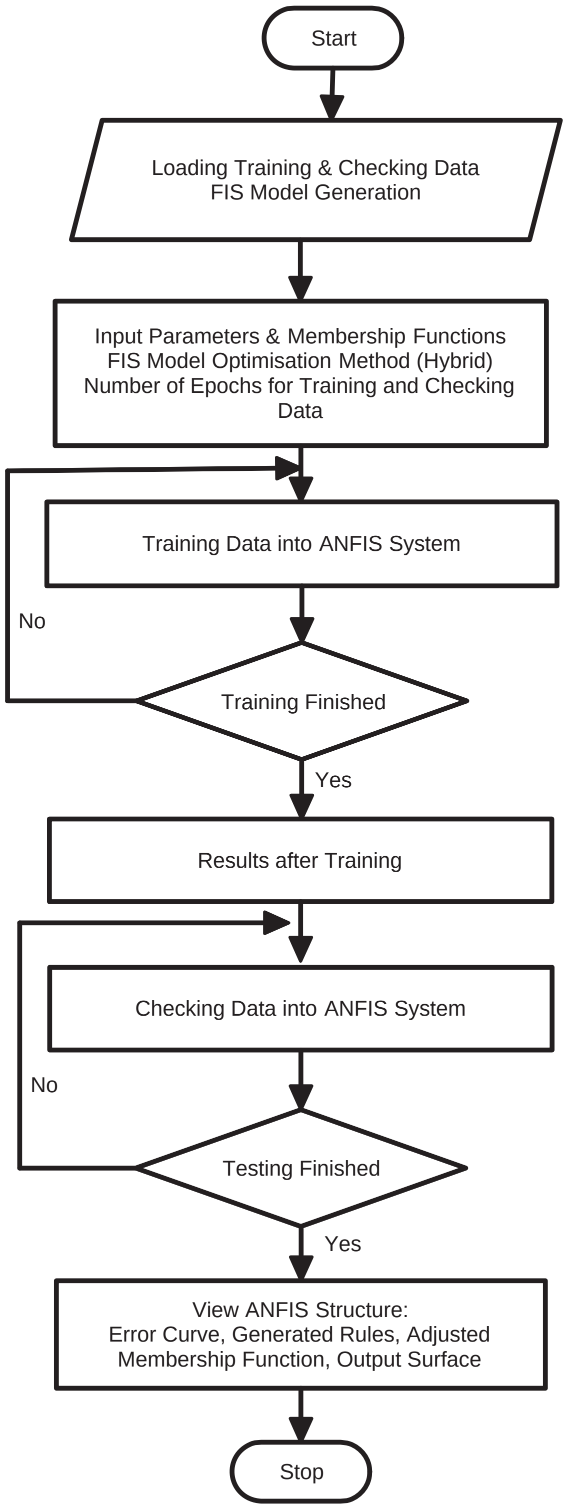

Figure 7.14: ANFIS Training System Steps.

system. The obtained training data must include as many mobile learning situations (different locations, devices, network bandwidth etc) as possible. ANFIS training uses the *anfis* function. The evaluation between the system and desired output is conducted using *evalfis* function.

The first step is to prepare the training data to work with ANFIS in MATLAB. The data set used as input to the *anfis* function must be in a matrix form, where the last column in the matrix is the output, and the matrix contains as many columns as needed to represent the inputs to the system. The rows will represent all the different data situations that exist. For example, if there is a system with three inputs and one output, Table 7.1 shows how the training data would be represented in a matrix. The creation of the membership functions is dependent on the system designer. The designer may create the parameters of the membership functions if they have knowledge of the expected shapes, or they can use the command *genfis1* from MATLAB to create the initial set of membership functions. The *genfis1* command is used to create the membership functions in this work (see Figure. 7.15).

$\text{Input}_{(1)}$	$\text{Input}_{(2)}$	$\text{Input}_{(3)}$	Output
4	2	2	3
1	3	5	2
...	...	...	...

Table 7.1: Example of Training Data Set.

Once the initial membership functions are created, the system training begins. The command provided by MATLAB to train an ANFIS system is *anfis*. In this function, the defined training data, membership functions created using the *fismat* command, and some training options are input in order to produce the most acceptable output.

When the training process is finished (*trn-fismat*), the final membership functions and training error from the training data set are produced (see Figure 7.16). The difference between the initial and final membership functions shows that the membership

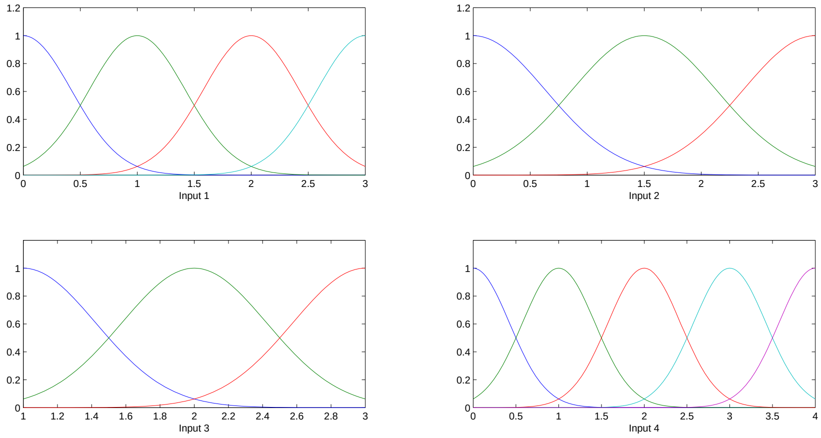

Figure 7.15: Initial Membership Functions.

function learns the training data and adjusts its shapes according to the dynamics of the system. A checking data set can be used beside the training data set for enhanced accuracy. ANFIS can function with only a training data set, however it is more effective to input some checking data as this will increase the range of possibilities to be understood by the system.

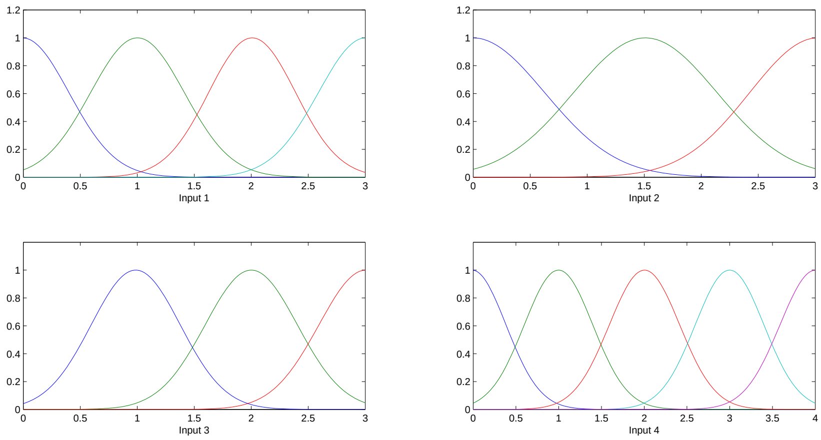

Figure 7.16: Final Membership Functions.

After the system training is complete, ANFIS provides a method to study and evaluate the system performance by using the *evalfis* function. This process starts

with entering input data sets without output values into the fuzzy system obtained when the training finished (*trn-fismat*). The output of this function represents the response of the system or the final output of the ANFIS system. This response output can be measured by means of correlations between the desired learner contexts and the learning content format as the system output (i.e. input/output). Once the ANFIS is trained, the system can be tested against different sets of data values to check the functionality of the proposed system.

To better understand the training steps (see Figure 7.14), the mobile learning example described below will demonstrate the ANFIS modelling.

Four input parameters were controlled, namely: Learner Location (**LL**), Network Bandwidth (**NB**), Battery Life (**BL**) and Device Software Capabilities (**SC**); and one output parameter was controlled: Adapted Learner Content (**ALC**) (see Figure 7.17).

The conditions that determine the decision made about the format of the learning content that will be delivered to learners depends solely on the criteria set by the human expert. Each one of the four input conditions is represented by the following term sets:

- **LL** has four membership functions, represented as *Home*, *Campus*, *Class* and *Outdoor*.
- **NB** has three membership functions, represented as *Low*, *Middle* and *High*.
- **BL** has three membership functions, represented as *Low*, *Half* and *Full*.
- **SC** has five membership functions, represented as *Text*, *Text+PDF*, *Text+Video*, *Text+PDF+Video* and *Text+Audio*.

The output conditions are: *Text*, *PDF*, *Audio* or *Video*.

The Sugeno type fuzzy rule based model has rules in the following form:

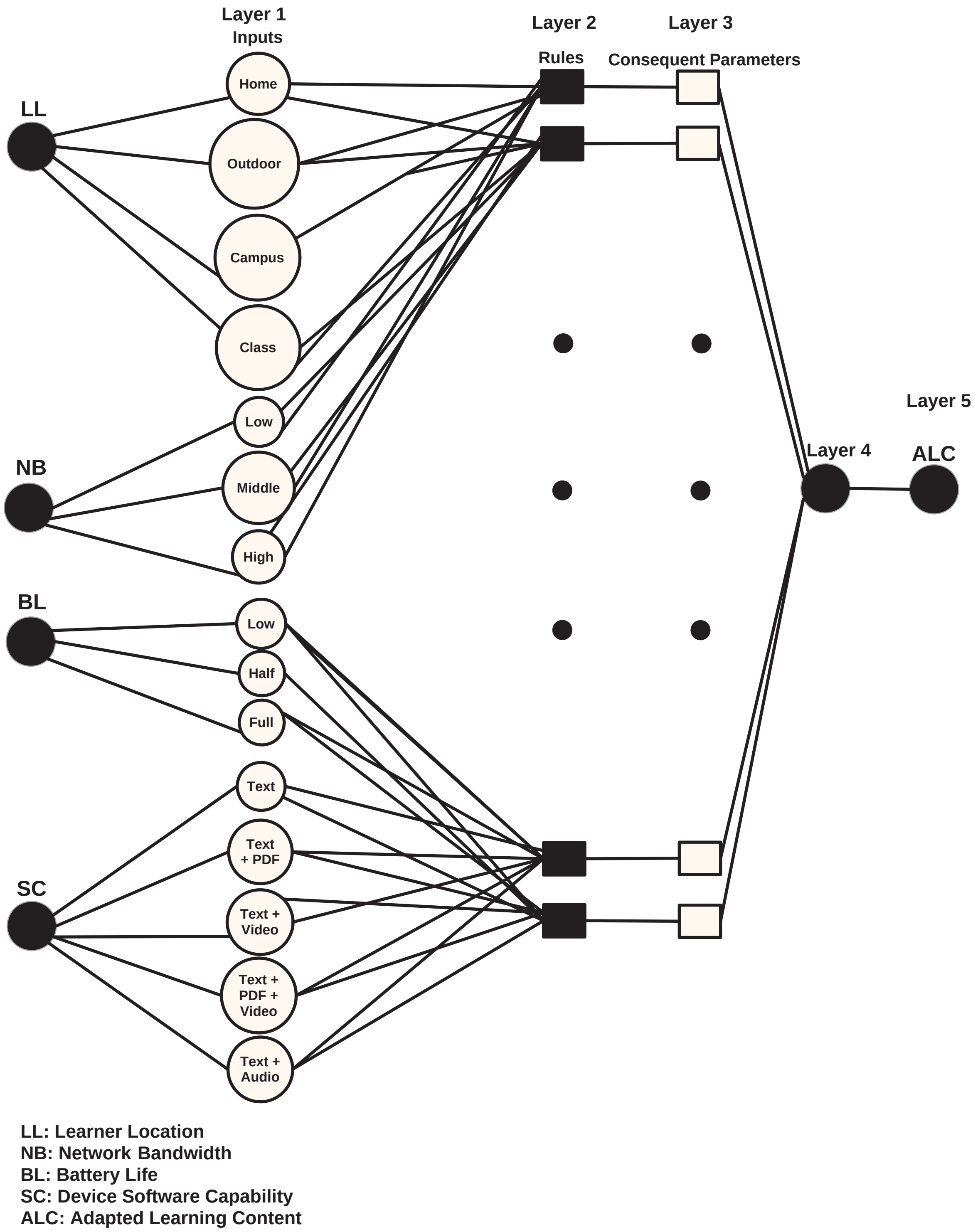

Figure 7.17: ANFIS Structure with Four Inputs and One Output.

IF (**LL** is *Outdoor*) **AND** (**NB** is *High*) **AND** (**BL** is *Full*) **AND** (**SC** is *Text+Video*) **THEN** (Learning Content Format is *Video*).

IF (**LL** is *Class*) **AND** (**NB** is *Low*) **AND** (**BL** is *Half*) **AND** (**SC** is *Text+PDF+Video*) **THEN** (Learning Content Format is *PDF*).

genfis1 is the function used to generate an initial single-output Fuzzy Inference System (FIS) matrix from training data. The example model must be initiated with default values for membership function numbers: "4*3*3*5"; and types: "Gaussian curve (*gaussmf*), Triangular-shaped (*trimf*) and Generalised bell-shaped (*gbellmf*)". These defaults provide membership functions on each of the four inputs, fifteen altogether. The generated FIS structure contains 180 fuzzy rules. Each rule generated by *genfis1* has one output membership function, which is of type "linear" by default.

Each of the fuzzy rules of ANFIS has a multi input single output structure. The adjustment of membership function parameters is computed by gradient vector, which shows how well the ANFIS is modeled by a given different set of training data consisting of certain conditions.

Several experiments have been conducted to assess whether the proposed modeled ANFIS has produced an acceptable result. ANFIS network training involves mapping inputs through input membership functions and mapping output through output membership functions. The parameters associated with each membership function will keep changing throughout the learning process.

The ANFIS training process starts by determining fuzzy sets, the number of sets of each input variable, and the shape of their membership function. All the training data passes through the ANN to adjust the input parameters, find the relationships between input/output and minimise the errors. Root Mean Square Error (RMSE) is

the function used to monitor the training errors. RMSE is used and defined as:

$$RMSE = \sqrt{\frac{1}{N}\sum_{j=1}^{N}(y_j - \hat{y}_j)^2} \tag{7.1}$$

Mean Average Error (MAE) is used and defined as:

$$MAE = \frac{1}{N}\sum_{j=1}^{N}|y_i - \hat{y}_i| \tag{7.2}$$

where N is the total number of prediction, $\hat{y}_j$ is the predicted time series and y_j is the original series.

Functions from the Fuzzy Logic Toolbox (FLT) were included in the MATLAB code, used to solve the input/output problem with different membership function numbers and types for each input. The difference of the RMSE between observed and modeled values was computed for each trial with different Epoch numbers, and the best structure was determined considering the lowest value of the RMSE.

Over-fitting is a common problem in ANFIS model building; this occurs when the data is over trained by ANFIS. Every data set that is trained using ANFIS has its maximum number of Epochs before over-fitting occurs; this causes the predicted output to be over its accuracy. The optimal number of Epochs can only be found through experiments.

Over-fitting is analysed by plotting the training and checking error values from the ANFIS simulation (see Figure 7.18). To avoid the over-fitting problem, the model will be tested by setting training Epoch numbers equal to different values to determine the optimal Epoch number with the lowest RMSE.

The Surface Viewer is used to display the dependency of one of the outputs on any one or two of the inputs. It generates and plots an output surface map for the system. The output is presented by a 3D surface model (see Figure 7.19). The predicted error

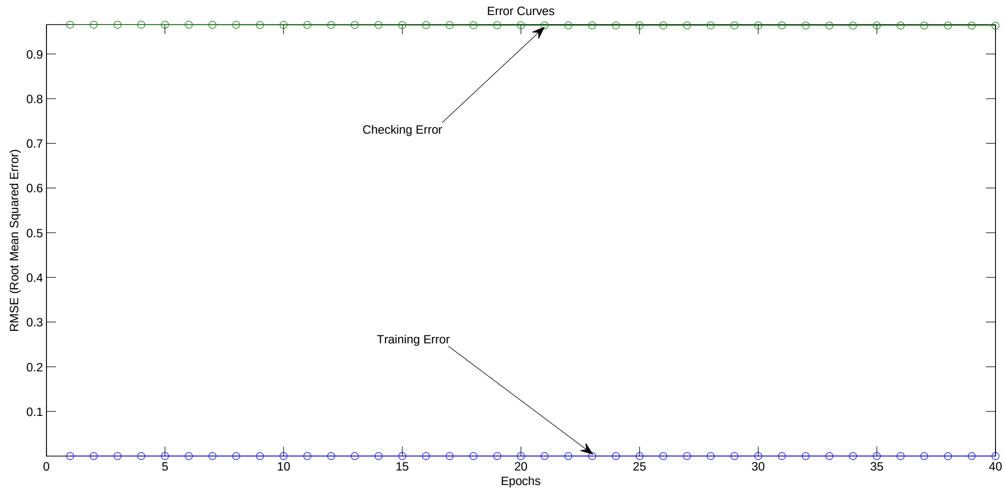

Figure 7.18: Error Curves.

for the proposed ANFIS is also shown (see Figure 7.20).

In this study, an ANFIS model based on both Neural Networks and Fuzzy Logic has been developed to adapt learning content format for mobile learning. The experiments were divided into four ANFIS structures to demonstrate learning activities in different contexts. For this purpose, computer simulations using MATLAB were undertaken, and statistical validation indexes were used for determining the performance of the best proposed model within each context. Once the ANFIS model inputs and output have been identified, it is necessary to validate the quality of the model results after training. Model validation will validate the accuracy and verify whether the model can be easily interpreted.

The performances of the ANFIS models of both training and checking data were evaluated and the best training/checking data set was selected according to RMSE value. Prediction accuracy was calculated by comparing the difference of predicted and measured values. In order to find the ideal ANFIS, the system was trained with different settings such as data set sample, Epoch number, membership function type and number, and number of inputs to achieve best performance. The validation tests

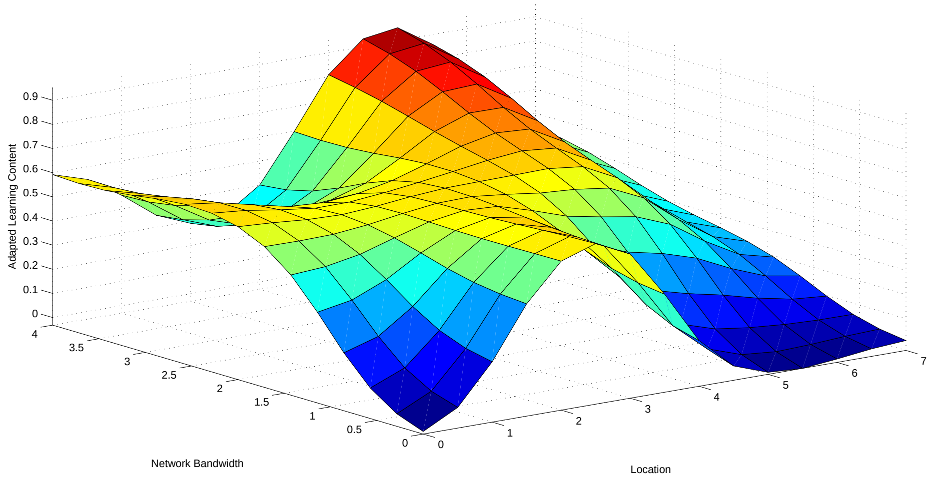

Figure 7.19: Surface Model of LL and NB for the ALC.

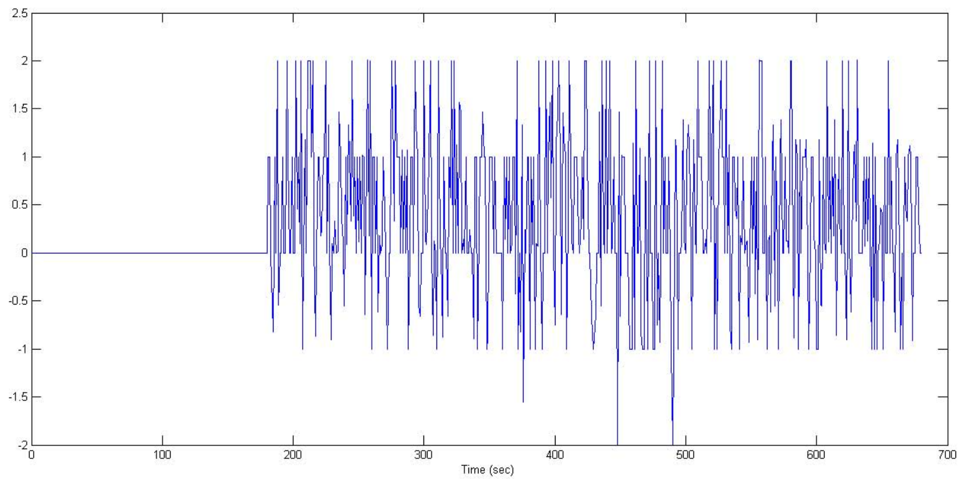

Figure 7.20: ANFIS Prediction Curve.

between the predicted results and the actual results for both training and testing phases are presented in Chapter 8.

7.6 Summary

In this chapter, ANFIS modelling was explained in the context of the mobile learning scenarios. An FIS controller was designed and simulated to perform adaptive learning content format for mobile learners. The ANFIS modelling is divided into two steps: Fuzzy Inference System (FIS) then Adaptive Neuro-Fuzzy Inference System (ANFIS). This chapter explained the FISs and ANFISs using the Sugeno type FIS structure, which involved creating, generating, training, testing, and validation of the FIS and ANFIS structure. An ANFIS model that attempts to account for dynamically changing inputs has been designed. The problem related to correctly tuning the fuzzy logic controller to the desired constraints was identified and pursued in the next chapter. Various experiments were conducted and the sizes of the training and checking sets were determined by taking into consideration the classification accuracies. An FIS has been developed and tested successfully. Before the final development phase, several rounds of checking of each membership function and the rules and structure conducted, and necessary changes were made according to its performance. The results are discussed in the next chapter and are compared with the results from other model structures.

Chapter 8

Simulation Results and Discussion

8.1 Outline

In Chapter 7, Fuzzy Inference System (FIS) and Adapted Neuro-Fuzzy Inference System (ANFIS) models were used to implement mobile learning scenarios to adapt learning content format according to learners' needs. The learner model developed in this thesis and presented (see Chapter 5) is logically partitioned into smaller elements or classes in the form of learner profiles, which together can represent the entire learning process. This chapter presents an ANFIS for delivering adapted learning content to mobile learners. The ANFIS model was designed using trial and error based on various experiments. In this study, an ANFIS model based on both Neural Network and Fuzzy Logic has been developed to adapt learning content format for mobile learning. The experiments were divided into five ANFIS structures to demonstrate learning activities in different contexts. For this purpose, computer simulations using MATLAB were carried out and statistical validation indexes were used for determining the performance of the best proposed model within each context.

This study was conducted to illustrate the potential effectiveness of ANFIS with hybrid learning, for the adaptation of learning content according to learners' needs. Study results show that ANFIS has been successfully implemented for learning content adaptation within different learning context scenarios. The performance of the ANFIS model was evaluated in Chapter 8 using standard error measurements which revealed the optimal setting necessary for better predictability (see Section 8.8). The MATLAB simulation results indicate that the performance of the ANFIS approach is valuable and easy to implement (see Sections 8.3, 8.4, 8.5, 8.6 and 8.7). The study results were based on an analysis of different model settings (inputs, membership functions and sample size); they confirm that the m-learning application is functional. However, it should be noted that an increase in the number of inputs being considered by the model will increase the response time for the mobile learner.

8.2 Introduction

As explained in Chapter 7, once the ANFIS model inputs and output have been identified, it is necessary to validate the quality of the model results after training. ANFIS applied a combination of the Least Squares method and the Back-Propagation method for training. Model validation will validate the accuracy and verify whether the model can be easily interpreted.

Various experiments were conducted and the sizes of the training and checking data sets were determined by taking into consideration the classification accuracies. The data set was divided into two separate data sets: the training data set and the checking data set. The training data set was used to train the ANFIS, whereas the checking data set was used to verify the accuracy and the effectiveness of the trained ANFIS model for the adaptation of learning content. In order to find out the optimal model to address the problem of mobile learning, a number of factors that play an important role in determining this model must be investigated.

The optimal ANFIS model setting will be selected based on comparing RMSE (refer to 7.1) values between different Epoch numbers and the number of membership functions assigned to each ANFIS structure. Four main aspects will be taken into consideration in relation to ANFIS system training: over-fitting, the number of membership functions, type of membership functions and training options (training data sample and Epoch number).

Numerous ANFIS models, each with a different number of inputs and membership functions, were trained to study the effect of data sample, number and type of membership functions. A number of inputs were controlled in each experiment, namely: Learner Location (**LL**), Network Bandwidth (**NB**), Battery Life (**BL**), Device Software Capabilities (**SC**), Environment Status (**ES**), Screen Size (**SS**); and one output was controlled: adapted learner content (**ALC**).

The validation tests between the predicted results and the actual results for both training and checking phases are presented (see Table 8.2, Table 8.4, Table 8.6, Table 8.8 and Table 8.10). These tables show the performance results from five ANFIS structures, namely ANFIS(1), ANFIS(2), ANFIS(3), ANFIS(4) and ANFIS(5). ANFIS has been selected to solve the problem of managing continuous changes in a mobile learning environment to delver adapted learning content. The learner modelling problem of mobile learning applications was explained in Chapter 5, using the Gaussian curve (*gaussmf*), Triangular-shaped (*trimf*), Generalized bell-shaped (*gbellmf*), Trapezidal-shaped (*trapmf*), Gaussian combination (*gauss2mf*) and Two Sigmoidally-shaped (*psigmf*) membership functions respectively. The membership functions of every input within each structure can be divided into areas, summarised in Tables 8.1, 8.3 8.5, 8.7 and 8.9.

MATLAB Toolbox requires several steps as below:

1. The *training data set* contains (10, 15 and 27 for $\text{ANFIS}_{(1)}$), (40, 60 and 81 for $\text{ANFIS}_{(2)}$), (100, 150 and 180 for $\text{ANFIS}_{(3)}$), (15, 25 and 31 for $\text{ANFIS}_{(4)}$) and (50, 70 and 96 for $\text{ANFIS}_{(5)}$) human experts conditions. The *checking data set* contains (40, 60 and 80 for $\text{ANFIS}_{(1)}$), (80, 100 and 180 for $\text{ANFIS}_{(2)}$), (200, 300 and 500 for $\text{ANFIS}_{(3)}$), (50, 70 and 90 for $\text{ANFIS}_{(4)}$) and (100, 200 and 300 for $\text{ANFIS}_{(5)}$) random cases.

2. Generate ('3*3*3' for $\text{ANFIS}_{(1)}$), ('3*3*3*3' for $\text{ANFIS}_{(2)}$), ('4*3*3*5' for $\text{ANFIS}_{(3)}$), ('2*2*2*2*2' for $\text{ANFIS}_{(4)}$) and ('2*2*2*3*2*2' for $\text{ANFIS}_{(5)}$) Membership Functions Structure respectively.

3. Check the *training data* against the membership functions shapes: Gaussian curve (*gaussmf*), Triangular-shaped (*trimf*), Generalized bell-shaped (*gbellmf*), Trapezidal-shaped (*trapmf*), Gaussian combination (*gauss2mf*) and Two Sigmoidally-shaped (*psigmf*).

4. Epoch Number: (10, 25 and 40 for ANFIS$_{(1)}$), (10, 25 and 40 for ANFIS$_{(2)}$), (10, 25 and 40 for ANFIS$_{(3)}$), (25, 30 and 40 for ANFIS$_{(4)}$) and (30, 40 and 50 for ANFIS$_{(5)}$).
5. Train the modeled FIS.
6. Model validation.

To build an ANFIS model, the first step is to select the type and the number of membership functions for the Layer$_{(1)}$ of the model (see Figure 7.12). In the case of mobile learning, a number of membership functions are used to test the sensitivity of the proposed ANFIS model to Gaussian curve (*gaussmf*, Equation 6.7), Triangular-shaped (*trimf*, Equation 6.3 or 6.4), Generalized bell-shaped (*gbellmf*, Equation 6.8), Trapezidal-shaped (*trapmf*, Equation 6.5 or 6.6), Gaussian combination (*gauss2mf*, Equation 6.10) and Two Sigmoidally-shaped (*psigmf*, Equation 6.9). Once the membership functions selected, five ANFIS model structures will run with ("3*3*3", "3*3*3*3", "4*3*3*5", "2*2*2*2*2" and "2*2*2*3*2*2") membership functions for each node of input data.

8.3 ANFIS$_{(1)}$: 3 Inputs and Membership Functions Number (3*3*3)

The first system, ANFIS$_{(1)}$, was structured by selecting three inputs and feeding them to the network. The selection of the inputs was based on the impact of these inputs on the output format for mobile learning. As shown in Figure 8.1, three input parameters were controlled (see Table 8.1):

- Learner Location (**LL**), which has three membership functions represented as "*Home*, *Class* and *Outdoor*"

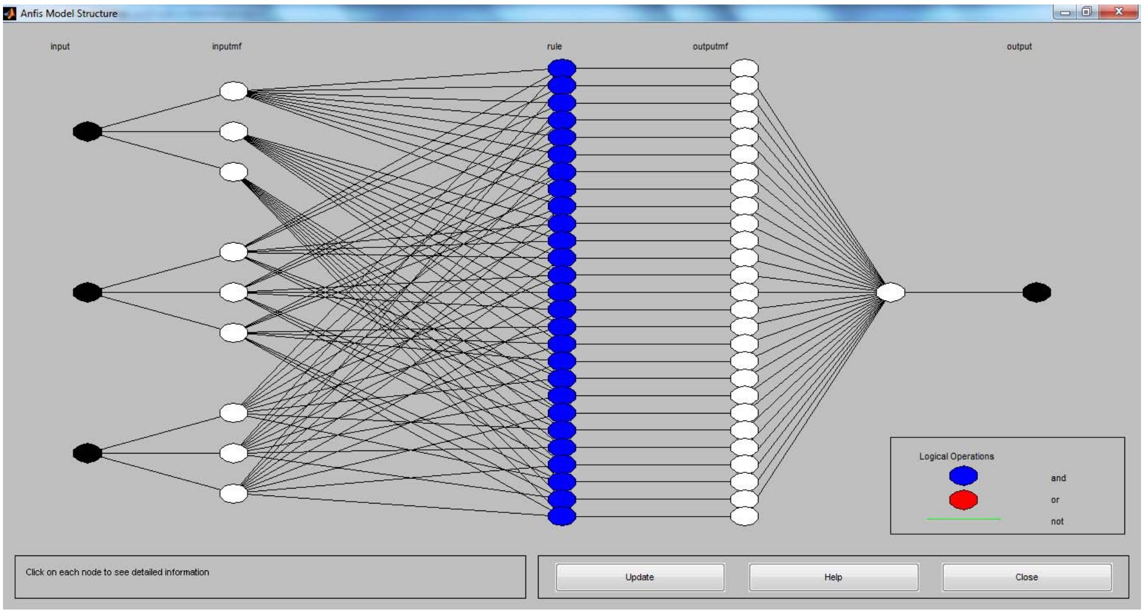

Figure 8.1: ANFIS$_{(1)}$ Model Structure.

- Network Bandwidth (**NB**), which has three membership functions represented as "*Low*, *Middle* and *High*"

- Software Capability (**SC**), which has three membership functions represented as "*Text*, *Text+PDF* and *Text+PDF+Video*"

The model will first be tested with default values for membership functions number "3*3*3"; membership function types "Gaussian, Triangular-shaped, Generalized bell-shaped, Trapezidal-shaped, Gaussian combination and Two Sigmoidally-shaped" are used. These defaults provide membership functions on each of the three inputs; nine altogether. The generated FIS structure contains 27 fuzzy rules. Table 8.2 shows that the Triangular-shaped (*trimf*) membership function performs most effectively with minimum RMSE during validation with the Epoch numbers 10, 25 and 40, and the Trapezidal-shaped (*trapmf*) membership function produced the worst results with 10 and 25 Epoch numbers. Gaussian (*gaussmf*) membership function produced the worst results with 40 Epoch numbers.

Input	Learner Location (LL)	Network Bandwidth (NB)	Software Capability (SC)
Membership Functions Values	Home, Class, Outdoor	Low, Middle, High	Text, Text+PDF, Text+PDF+Video

Table 8.1: Inputs and Membership Functions Values for ANFIS$_{(1)}$.

ANFIS Parameter Type	ANFIS (1)			ANFIS (2)			ANFIS (3)			ANFIS (4)			ANFIS (5)			ANFIS (6)		
Number of Input	3																	
Membership Functions type	Gaussian curve			Triangular-shaped			Generalized bell-shaped			Trapezidal-shaped			Gaussian combination			Two Sigmoidally-shaped		
Number of Membership Functions	3*3*3																	
Training Data Set	10	15	27	10	15	27	10	15	27	10	15	27	10	15	27	10	15	27
Checking Data Set	40	60	80	40	60	80	40	60	80	40	60	80	40	60	80	40	60	80
Epoch Number	10	25	40	10	25	40	10	25	40	10	25	40	10	25	40	10	25	40
Number of Nodes	78																	
Number of Linear Parameters	108																	
Number of Nonlinear Parameters	18			27						36								
Total Number of Parameters	126			135						144								
Number of Fuzzy Rules	27																	
Input Combinations	LL, NB, BL																	
Root Mean Square Error (RMSE)	1.0177	0.9652	0.8265	0.9274	0.863	0.7426	1.0094	0.965	0.8217	1.0392	0.9832	0.8258	1.0392	0.9823	0.8258	1.0389	0.9827	0.826
Mean Average Error (MAE)	0.7085	0.6611	0.5348	0.58	0.5377	0.4795	0.7037	0.6654	0.5314	0.72	0.6827	0.5313	0.72	0.6812	0.5315	0.7198	0.6817	0.5313

Table 8.2: The ANFIS$_{(1)}$ Structure Information (Membership Functions Number: 3*3*3).

8.4 ANFIS$_{(2)}$: 4 Inputs and Membership Functions Number (3*3*3*3)

ANFIS$_{(2)}$ is similar to ANFIS$_{(1)}$; the only difference is the inclusion of another input, which is Battery Life (**BL**). Four inputs were considered in the second experiment (see Table 8.3):

- Learner Location (**LL**), which has three membership functions represented as "*Home*, *Class* and *Outdoor*"
- Network Bandwidth (**NB**), which has three membership functions represented as "*Low*, *Middle* and *High*"
- Battery Life (**BL**), which has three membership functions represented as "*Low*, *Half* and *Full*"
- Software Capability (**SC**), which has three membership functions represented as "*Text*, *Text+PDF* and *Text+PDF+Video*".

As shown in Figure 8.2, **LL**, **NB**, **BL** and **SC** are the model inputs, with default values for membership functions number "3*3*3*3"; membership function types "Gaussian, Triangular-shaped, Generalized bell-shaped, Trapezidal-shaped, Gaussian combination and Two Sigmoidally-shaped" are used. These defaults provide membership functions on each of the four inputs; twelve altogether. The generated FIS structure contains 81 fuzzy rules. The summary of the tested membership functions number and type with their respective Epoch numbers is tabulated in Table 8.4.

Table 8.4 shows, for the model with four parameters, that the Trapezidal-shaped (*trapmf*) membership function with the Epoch numbers 10 and 40 and the Generalized bell-shaped (*gbellmf*) membership function with the Epoch number 25 are selected as

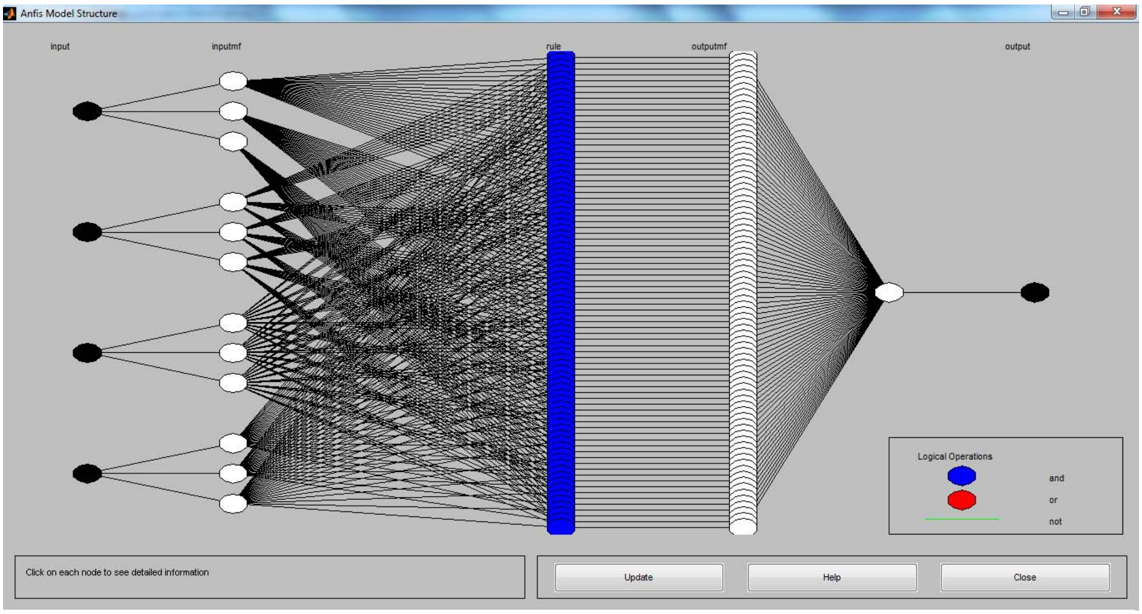

Figure 8.2: ANFIS$_{(2)}$ Model Structure.

the best fit model for describing the flow of mobile learning scenarios. The Triangular-shaped (*trimf*) membership function with Epoch numbers 25 and 40 and the Gaussian combination (*gauss2mf*) membership function with Epoch number 10 perform the worst with maximum RMSE during validation.

Input	Learner Location (LL)	Network Bandwidth (NB)	Battery Life (BL)	Software Capability (SC)
Membership Functions Values	Home, Class, Outdoor	Low, Middle, High	Low, Middle, Full	Text, Text+PDF, Text+PDF+Video

Table 8.3: Inputs and Membership Functions Values for ANFIS$_{(2)}$.

ANFIS Parameter Type	ANFIS (1)			ANFIS (2)			ANFIS (3)			ANFIS (4)			ANFIS (5)			ANFIS (6)		
Number of Input	4																	
Membership Functions type	Gaussian curve			Triangular-shaped			Generalized bell-shaped			Trapezidal-shaped			Gaussian combination			Two Sigmoidally-shaped		
Number of Membership Functions	3 3 3 3																	
Training Data Set	40	60	81	40	60	81	40	60	81	40	60	81	40	60	81	40	60	81
Checking Data Set	80	100	180	80	100	180	80	100	180	80	100	180	80	100	180	80	100	180
Epoch Number	10	25	40	10	25	40	10	25	40	10	25	40	10	25	40	10	25	40
Number of Nodes	193																	
Number of Linear Parameters	405																	
Number of Nonlinear Parameters	24			36						48								
Total Number of Parameters	429			441						453								
Number of Fuzzy Rules	81																	
Input Combinations	LL, NB, BL, SC																	
Root Mean Square Error (RMSE)	0.9524	0.8202	0.7708	0.919	0.825	0.775	0.9442	0.8129	0.7665	0.9165	0.8173	0.764	0.9734	0.8174	0.7643	0.9729	0.8171	0.7636
Mean Average Error (MAE)	0.6501	0.5041	0.494	0.626	0.5069	0.4965	0.6439	0.4995	0.4898	0.6178	0.4957	0.4877	0.6571	0.496	0.4881	0.6562	0.4953	0.4873

Table 8.4: The ANFIS$_{(2)}$ Structure Information (Membership Functions Number: 3*3*3*3).

8.5 ANFIS$_{(3)}$: 4 Inputs and Membership Functions Number (4*3*3*5)

Similar to the ANFIS$_{(2)}$ and as shown in Figure 8.3, four inputs were controlled in ANFIS$_{(3)}$: **LL**, **NB**, **BL** and **SC**, but with default values for membership functions number "4*3*3*5". Membership function types "Gaussian, Triangular-shaped, Generalized bell-shaped, Trapezidal-shaped, Gaussian combination and Two Sigmoidally-shaped" are used. The inputs are (see Table 8.5):

- Learner Location (**LL**), which has four membership functions represented as "*Home*, *Campus*, *Class* and *Outdoor*"
- Network Bandwidth (**NB**), which has three membership functions represented as "*Low*, *Middle* and *High*"
- Battery Life (**BL**), which has three membership functions represented as "*Low*, *Half* and *Full*"
- Software Capability (**SC**), which has three membership functions represented as "*Text*, *Text+PDF*, *Text+video*, *Text+PDF+Video* and *Text+Audio*".

These defaults provide membership functions on each of the four inputs; fifteen altogether. The generated FIS structure contains 180 fuzzy rules.

This model was previously explained and is shown in Chapter 7 (Figure 7.17). Table 8.6 shows, for this model, that the Triangular-shaped (*trimf*) membership function with the Epoch numbers 10, 25 and 40 is the best fit model, and that the Trapezidal-shaped (*trapmf*) and Gaussian (*gaussmf*) membership functions perform the worst.

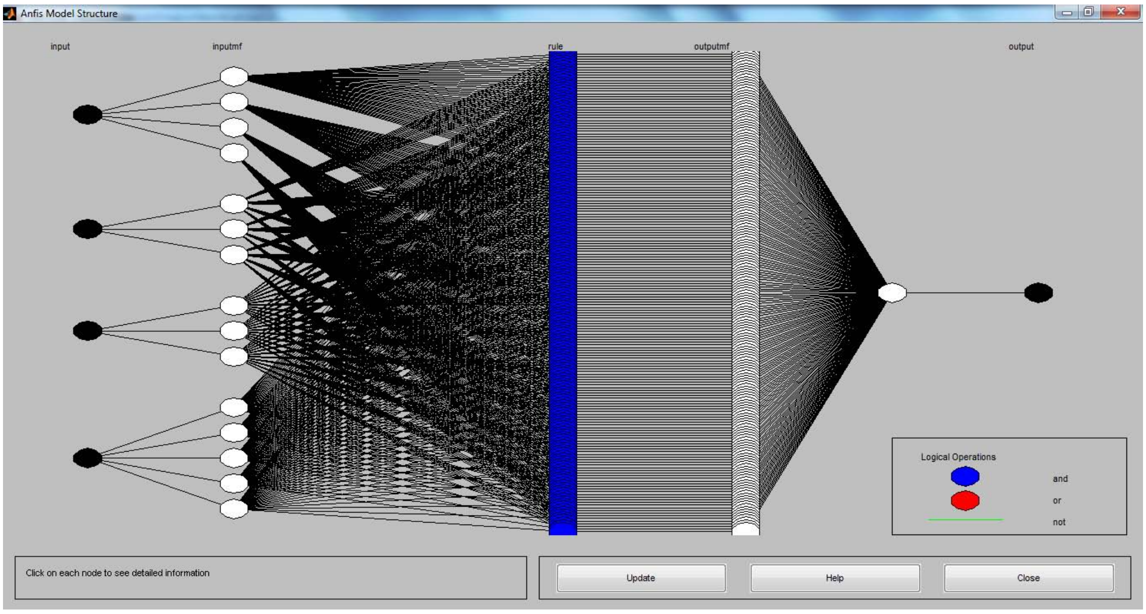

Figure 8.3: ANFIS$_{(3)}$ Model Structure.

Input	Learner Location (LL)	Network Bandwidth (NB)	Battery Life (BL)	Software Capability (SC)
Membership Functions Values	Home, Campus, Class, Outdoor	Low, Middle, High	Low, Middle, Full	Text, Text+PDF, Text+Video, Text+PDF+Video, Text+Audio

Table 8.5: Inputs and Membership Functions Values for ANFIS$_{(3)}$.

ANFIS Parameter Type	ANFIS (1)			ANFIS (2)			ANFIS (3)			ANFIS (4)			ANFIS (5)			ANFIS (6)		
Number of Input	4																	
Membership Functions Type	Gaussian curve			Triangular-shaped			Generalized bell-shaped			Trapezidal-shaped			Gaussian combination			Two Sigmoidally-shaped		
Number of Membership Functions	4 3 3 5																	
Training Data Set	100	150	180	100	150	180	100	150	180	100	150	180	100	150	180	100	150	180
Checking Data Set	200	300	500	200	300	500	200	300	500	200	300	500	200	300	500	200	300	500
Epoch Number	10	25	40	10	25	40	10	25	40	10	25	40	10	25	40	10	25	40
Number of Nodes	397																	
Number of Linear Parameters	900																	
Number of Nonlinear Parameters	30			45						60								
Total Number of Parameters	930			945						960								
Number of Fuzzy Rules	180																	
Input Combinations	LL, NB, BL, SC																	
Root Mean Square Error (RMSE)	0.8153	0.8047	0.8263	0.8059	0.7921	0.808	0.8185	0.8012	0.8248	0.8803	0.8044	0.8249	0.8769	0.8044	0.8249	0.8738	0.8042	0.8249
Mean Average Error (MAE)	0.5093	0.4971	0.5336	0.5182	0.5032	0.5354	0.5132	0.4958	0.5336	0.5485	0.4968	0.5344	0.5464	0.4968	0.5343	0.5446	0.4966	0.5344

Table 8.6: The ANFIS$_{(3)}$ Structure Information (Membership Functions Number: 4*3*3*5).

8.6 ANFIS$_{(4)}$: 5 Inputs and Membership Function Number (2*2*2*2*2)

Similar to the previous ANFIS structures, the only difference with ANFIS$_{(4)}$ is that five inputs included, with the inclusion of the Environment Status (**ES**) input. As shown in Figure 8.4, five inputs were considered in the fourth experiment with default values for membership functions number "2*2*2*2*2". Membership function types "Gaussian, Triangular-shaped, Generalized bell-shaped, Trapezidal-shaped, Gaussian combination and Two Sigmoidally-shaped" are used. The inputs are (see Table 8.7):

- Learner Location (**LL**), which has two membership functions represented as "*Home* and *Outdoor*"
- Network Bandwidth (**NB**), which has two membership functions represented as "*Low* and *High*"
- Battery Life (**BL**), which has two membership functions represented as "*Low* and *Full*"
- Software Capability (**SC**), which has two membership functions represented as "*PDF* and *PDF+Video*"
- Environment Status (**ES**), which has two membership functions represented as "*Sunny* and *Raining*".

These defaults provide membership functions on each of the five inputs; ten altogether. The generated FIS structure contains 32 fuzzy rules.

Table 8.8 shows, for this model, that the Generalized bell-shaped (*gbellmf*) membership function with the Epoch numbers 25, 30 and 40 is the best fit model, and that the Trapezidal-shaped (*trapmf*), Gaussian Combination (*gauss2mf*) and Two

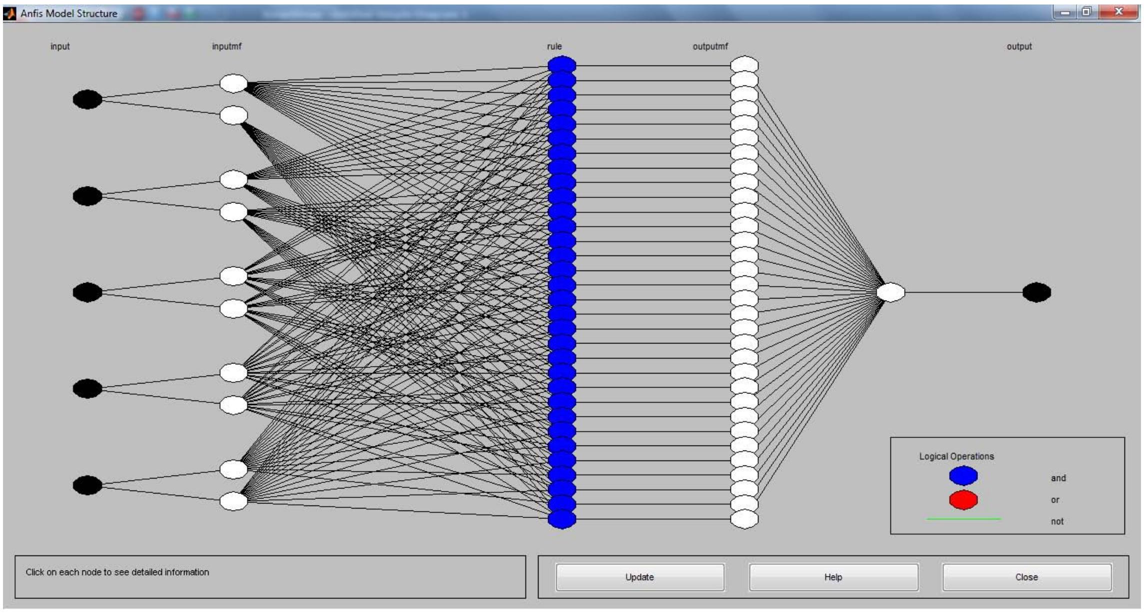

Figure 8.4: ANFIS$_{(4)}$ Model Structure.

Sigmoidally-shaped (*psigmf*) membership functions with Epoch numbers 25, 30 and 40 perform the worst.

Input	Learner Location (LL)	Network Bandwidth (NB)	Battery Life (BL)	Software Capability (SC)	Environment Status (ES)
Membership Functions Values	Home, Outdoor	Low, High	Low, Full	PDF, PDF+Video	Sunny, Raining

Table 8.7: Inputs and Membership Functions Values for ANFIS$_{(4)}$.

ANFIS Parameter Type	ANFIS (1)			ANFIS (2)			ANFIS (3)			ANFIS (4)			ANFIS (5)			ANFIS (6)		
Number of Input	5																	
Membership Functions type	Gaussian curve			Triangular-shaped			Generalized bell-shaped			Trapezidal-shaped			Gaussian combination			Two Sigmoidally-shaped		
Number of Membership Functions	2 2 2 2 2																	
Training Data Set	15	25	31	15	25	31	15	25	31	15	25	31	15	25	31	15	25	31
Checking Data Set	50	70	90	50	70	90	50	70	90	50	70	90	50	70	90	50	70	90
Epoch Number	25	30	40	25	30	40	25	30	40	25	30	40	25	30	40	25	30	40
Number of Nodes	92																	
Number of Linear Parameters	192																	
Number of Nonlinear Parameters	20			30						40								
Total Number of Parameters	212			222						232								
Number of Fuzzy Rules	32																	
Input Combinations	LL, NB, BL, SC, ES																	
Root Mean Square Error (RMSE)	0.5885	0.6198	0.6165	0.5948	0.6241	0.6166	0.5858	0.6171	0.6162	0.5948	0.6241	0.6166	0.5948	0.6241	0.6166	0.5948	0.6241	0.6166
Mean Average Error (MAE)	0.3544	0.3882	0.3802	0.3538	0.3895	0.3802	0.3535	0.3867	0.3804	0.3538	0.3895	0.3802	0.3539	0.3895	0.3802	0.3538	0.3895	0.3802

Table 8.8: The ANFIS$_{(4)}$ Structure Information (Membership Functions Number: 2*2*2*2*2).

8.7 ANFIS$_{(5)}$: 6 Inputs and Membership Functions Number (2*2*2*3*2*2)

Similar to the previous ANFIS structures, the only difference with ANFIS$_{(5)}$ is that six inputs are included, with the inclusion of the Screen Size (**SS**) input. As shown in Figure 8.5, six inputs were considered in the fifth experiment with default values for membership functions number "2*2*2*3*2*2". Membership function types "Gaussian, Triangular-shaped, Generalized bell-shaped, Trapezidal-shaped, Gaussian combination and Two Sigmoidally-shaped" are used. The inputs are (see Table8.9):

- Learner Location (**LL**), which has two membership functions represented as "*Home* and *Outdoor*"
- Network Bandwidth (**NB**), which has two membership functions represented as "*Low* and *High*"
- Battery Life (**BL**), which has two membership functions represented as "*Low* and *Full*"
- Software Capability (**SC**), which has three membership functions represented as "*Text*, *Text+PDF* and *PDF+Video*"
- Environment Status (**ES**), which has two membership functions represented as "*Sunny* and *Raining*"
- Screen Size (**SS**), which has two membership functions represented as "*Small* and *Large*".

These defaults provide membership functions on each of the six inputs; thirteen altogether. The generated FIS structure contains 96 fuzzy rules.

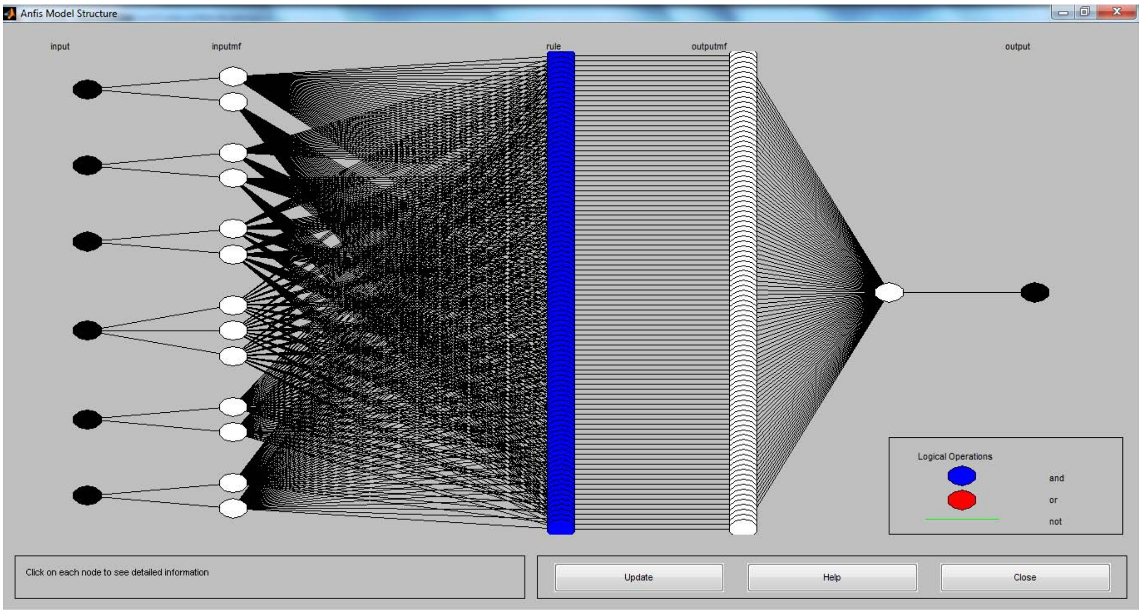

Figure 8.5: ANFIS$_{(5)}$ Model Structure.

Table 8.10 shows, for this model, that the Generalized bell-shaped (*gbellmf*) membership function with the Epoch numbers 30, 40 and 50 is the best fit model, and that the Trapezidal-shaped (*trapmf*) and Triangular-shaped (*trimf*) membership functions with Epoch numbers 30, 40 and 50 perform the worst.

Input	Learner Location (LL)	Network Bandwidth (NB)	Battery Life (BL)	Software Capability (SC)	Environment Status(ES)	Screen Size(SS)
Membership Functions Values	Home, Outdoor	Low, High	Low, Full	Text, Text+PDF, PDF+Video	Sunny, Raining	Small , Large

Table 8.9: Inputs and Membership Functions Values for ANFIS$_{(5)}$.

ANFIS Parameter Type	ANFIS (1)			ANFIS (2)			ANFIS (3)			ANFIS (4)			ANFIS (5)			ANFIS (6)		
Number of Input	6																	
Membership Functions Type	Gaussian curve			Triangular-shaped			Generalized bell-shaped			Trapezidal-shaped			Gaussian combination			Two Sigmoidally-shaped		
Number of Membership Functions	2 2 2 3 2 2																	
Training Data Set	50	70	96	50	70	96	50	70	96	50	70	96	50	70	96	50	70	96
Checking Data Set	100	200	300	100	200	300	100	200	300	100	200	300	100	200	300	100	200	300
Epoch Number	30	40	50	30	40	50	30	40	50	30	40	50	30	40	50	30	40	50
Number of Nodes	227																	
Number of Linear Parameters	672																	
Number of Nonlinear Parameters	26			39						52								
Total Number of Parameters	698			711						724								
Number of Fuzzy Rules	96																	
Input Combinations	LL, NB, BL, SC, ES, SS																	
Root Mean Square Error (RMSE)	0.5834	0.6838	0.7403	0.5888	0.6858	0.7403	0.5721	0.6773	0.7403	0.5888	0.6858	0.7403	0.5886	0.6857	0.7403	0.5887	0.6858	0.7403
Mean Average Error (MAE)	0.3206	0.3973	0.4571	0.3201	0.3963	0.4571	0.3201	0.3966	0.4571	0.32	0.3963	0.4571	0.32	0.3963	0.4571	0.32	0.3963	0.4571

Table 8.10: The ANFIS$_{(5)}$ Structure Information (Membership Functions Number: 2*2*2*3*2*2).

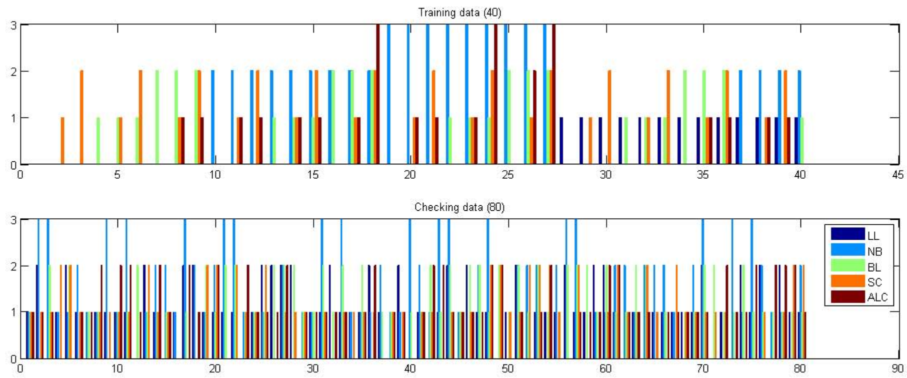

Figure 8.6: Training and Checking Data.

8.8 Discussion and Performance Analysis

Various experiments were conducted and the sizes of the training and checking data sets were determined by taking into consideration the classification accuracies. The data set was divided into two separate data sets: the *training data set* and the *checking data set*. The *training data set* was used to train the ANFIS, whereas the *checking data set* was used to verify the accuracy and the effectiveness of the trained ANFIS model for the adaptation of learning content. The partition of data used is $\text{ANFIS}_{(2)}$ shown in Figure 8.6. In order to find out the optimal model to address the problem of mobile learning, a number of factors which play an important role in determining this model must be investigated.

The optimal ANFIS model setting will be selected based on comparing RMSE values between different Epoch numbers. The number of membership functions assigned to each ANFIS structure is chosen by trial and error. Four main aspects will be taken into consideration in relation to ANFIS system training: over-fitting, number of membership functions, type of membership functions and training options (training data sample and Epoch number).

The ANFIS models are evaluated based on their performance in training and checking sets as shown in Table 8.2, Table 8.4, Table 8.6, Table 8.8 and Table 8.10. The

ANFIS models showed significant variations in the criteria of the performance evaluation in terms of data sample, number of membership functions and type of membership functions.

Findings show that the ANFIS models are accurate and consistent in different subsets, where all the values of Root Mean Square Error (RMSE, 7.1) and Mean Average Error (MAE, 7.2) are minimum. They also show that:

- In $ANFIS_{(1)}$, the lower value of RMSE is 0.7426 and the highest value of the RMSE is 1.0392,
- In $ANFIS_{(2)}$, the lower value of RMSE is 0.7636 and the highest value of the RMSE is 0.9734, and
- In $ANFIS_{(3)}$, the lower value of RMSE is 0.7921 and the highest value of the RMSE is 0.8803.

The values of the RMSE and MAE in $ANFIS_{(3)}$ are much lower than in $ANFIS_{(1)}$ and $ANFIS_{(2)}$. $ANFIS_{(3)}$, which consists of four inputs and "4*3*3*5" membership functions, has shown the highest efficiency and correlation, and the minimum RMSE and MAE.

They also show that:

- In $ANFIS_{(4)}$, the lower value of RMSE is 0.5858 and the highest value of the RMSE is 0.6241, and
- In $ANFIS_{(5)}$, the lower value of RMSE is 0.5721 and the highest value of the RMSE is 0.7403.

As a result, $ANFIS_{(3)}$, $ANFIS_{(4)}$ and $ANFIS_{(5)}$ were selected as the three best-fit models to deliver learning content for mobile learning.

Both the type and number of membership functions are important in building the ANFIS architecture. To investigate the effects of the number and types of membership functions, varied training and checking data sets were used. Based on the experiment results in Table 8.2, Table 8.4, Table 8.6, Table 8.8 and Table 8.10, a number of membership functions are needed to achieve desired performance of the model.

$\text{ANFIS}_{(3)}$, which has the same number of inputs as $\text{ANFIS}_{(2)}$ but varyies in the number of membership functions and the training sample, was more successful than $\text{ANFIS}_{(2)}$. This indicates that the number of membership functions and the training data sample have a positive training effect on the production of acceptable system output. Although the time required to create the network for $\text{ANFIS}_{(3)}$ is slightly higher than to create a network for $\text{ANFIS}_{(2)}$, it produced better results.

The required number of membership functions is determined through trials, and is based on error values. Findings indicate that each ANFIS model is very sensitive to the type and number of membership functions. Increasing the number of membership functions per input does not necessarily increase model performance and usually leads to model over-fitting.

The RMSE values were used to determine the best membership function and Epoch number in order to select the best fit model. From these results, the Triangular-shaped (*trimf*) membership function with optimized Epoch numbers of 10, 25 and 40 was found to be the best model with the lowest RMSE value for two model structures ("3*3*3" and "4*3*3*5"). The Trapezidal-shaped (*trapmf*) membership function with Epoch numbers 10 and 40 and the Generalized bell-shaped (*gbellmf*) membership function with Epoch numbers 25 (in $\text{ANFIS}_{(2)}$) were found the best fit model with lowest RMSE value for model structure ("3*3*3*3").

The Generalized bell-shaped (*gbellmf*) membership function with Epoch numbers 25, 30 and 40 (in $\text{ANFIS}_{(4)}$) and 30, 40 and 50 (in $\text{ANFIS}_{(5)}$) were found to be

the best model with the lowest RMSE value for model structures ("2*2*2*2*2") and ("2*2*2*3*2*2").

The Trapezidal-shaped (*trapmf*) membership function with optimized Epoch numbers of 10 and 25 (in $\text{ANFIS}_{(1)}$) and 10 (in $\text{ANFIS}_{(3)}$) and the Gaussian (*gaussmf*) membership function with Epoch numbers 40 (in $\text{ANFIS}_{(1)}$) and 25 and 40 (in $\text{ANFIS}_{(3)}$) were found to be the worst model with the highest RMSE value for two model structures ("3*3*3" and "4*3*3*5"). The Triangular-shaped (*trimf*) membership function with optimized Epoch numbers of 25 and 40 was found to be the worst model with the highest RMSE value for model structures ("3*3*3*3").

The Trapezidal-shaped (*trapmf*), Gaussian Combination (*gauss2mf*) and Two Sigmoidally-shaped (*psigmf*) membership functions with Epoch numbers 25, 30 and 40 were found to be the worst models for describing the flow of mobile learning scenarios with the highest RMSE value for model structures ("2*2*2*2*2").

Table 8.6 and Table 8.8 show different approximations with regarding to different numbers of Epoch and membership functions ('4*3*3*5' and '2*2*2*2*2'). The data shows that the number of training samples has influenced the ANFIS behaviour; adding new possibilities produces more acceptable results. The reinforces the importance in fuzzy logic of training the system with a set of training data that includes as many different possibilities of data sample.

Training options are another important factor for building the ANFIS architecture. This includes the Epoch number, Error Goal, Initial Step size and Step size decrease or increase rate. For $\text{ANFIS}_{(1)}$, $\text{ANFIS}_{(2)}$, $\text{ANFIS}_{(3)}$, $\text{ANFIS}_{(4)}$ and $\text{ANFIS}_{(5)}$, different Epoch numbers are used. It can be concluded that the increase in Epoch number for a training data set does not necessarily improve the system performance significantly, however it does help to overcome the problem of over-fitting. As mentioned before, to enhance system performance, the training data should cover all different possibilities.

Therefore, in the case of mobile learning, different sets of *training data set* that change according to specific input(s) are used. Looking at the different tests shown, each ANFIS structure has it own best and worse fit model according to the current ANFIS settings. A number of tests were undertaken in relation to the number of Epochs. From these tests, it can be concluded that the most important factors for achieving good performance are the training data sample and the number/type of membership functions, rather than the number of Epochs or the training options.

Results in the Table 8.2, Table 8.4, Table 8.6, Table 8.8 and Table 8.10, demonstrate that the ANFIS models performance is, in general, accurate and good, where some rules are covered by human expert training data set and the missing rules are detected by ANFIS. The results of the ANFIS models demonstrate that ANFIS can be successfully applied to adapt learning content according to each learner's current situation or needs.

From the above tables, the following conclusions can be drawn:

- The type of membership functions and the number of membership functions are important in building the ANFIS architecture. These are chosen empirically by trial and error.
- ANFIS is influenced more greatly by the number of membership functions for each input variable than by the number of training samples.
- To avoid over-fitting, a small number of membership functions should be used first. A larger number of membership functions leads to a small error on the training set but does not necessarily lead to a small error on the test set.
- The time required for successful ANFIS learning, in order to achieve FIS model convergence, is always a matter of concern. Such concern becomes more critical when performing ANFIS training to model the behaviour of mobile learning

system.

- Although the ANFIS shows good performance, it is still a problem to find the optimum number of fuzzy rules in the fuzzy model.
- A large number of Fuzzy IF-THEN rules, used to represent the system behaviour, will significantly increase the convergence time such that the ANFIS performance might not be reliable in real-time applications.
- The number of rules of the FIS that are initiated by the system depends on the overall number of input Fuzzy subsets. Each possible combination of a subset of Fuzzy inputs will result in a new rule. This large number of rules will disrupt the performance of ANFIS training such that it will lag behind the requirements of real-time applications.
- In order to overcome the problem of large number of Fuzzy IF-THEN rules, an efficient inputs selection is required, which uses a sufficient amount of input-output data pairs for the effective compaction of the initial FIS size.

In this work, the applicability and capability of the ANFIS method have been investigated through the use of a number of data sets, membership functions and types of membership functions. Thirty ANFIS models were constructed using combinations of inputs, sample training data, and membership function numbers and types for the purpose of mobile learning. These ANFIS models were trained and tested. The results of the ANFIS models and observation were compared and evaluated. ANFIS models were compared based on their performance in training and checking data sets.

One of the motivations behind the design of an Adaptive Mobile Learning System was to support the learner throughout the learning activity by realising the adaptation principle (Chapter 3 and 4), which consists of:

- capturing the learner's context (implicit and explicit) and then

- utilising the captured learner context to form a Learner Model (Chapter 5).

If the captured learner context which forms the Enhanced Learner Model truly represents the entire learning activity with its changes over time, the use of ANFIS for a reasoning engine would keep up with the change in learner context and form accurate adaptation decisions.

A key contribution of this work is the Adaptation Components, with a specific focus on the detailed Learner Model Components. Results presented in Table (8.2, Table (8.4, (8.6, Table (8.8 and Table (8.10) confirm that the m-Learning application is feasible and will be effective with ANFIS modelling. The feasibility of the Reasoning Layer confirms assumptions made in Chapters 3, 4 and 5 of this thesis. These results were based on analysis of a number of inputs; the research confirms that the m-learning application is functional, however an increase in the number of inputs being considered will increase the response time. One of the key challenges in m-learning is to ensure that an appropriate balance is struck between conveniences provided by adaptation and the time taken to provide the user with a personalised experience.

8.9 Summary

This chapter has shown a successful implementation of ANFIS for learning content adaptation for mobile learners. This study was conducted to illustrate the potential effectiveness of ANFIS with hybrid learning, for the adaptation of learning content format for mobile learning users. The performance of ANFIS was evaluated using standard error measurements which revealed the optimal sitting necessary for better predictability. The numbers of fuzzy rules obtained from the human experts are insufficient, therefore the ANFIS adopts a hybrid approach that combines the FIS with the Neural Network in determining a complete fuzzy rule system.

The ANFIS approach has successfully solved the problem of incompleteness in the

decisions made by the human expert(s) by completing all possible rules. By training the Neural Network with different training data set conditions based on the human expert(s), the neural network is expected to recognize other decisions that were previously not detected. Model rules, inputs, membership function numbers and types, Epoch numbers and training samples were examined to determine the optimal model for different context scenarios.

Chapter 9

Conclusion and Recommendation for Further Research

9.1 Conclusions

Learning via a mobile device is a new form of learning that has emerged as a result of advances in wireless technology and hand-held devices in recent years. One main advantage of wireless networks is the ability to provide connections anywhere at anytime. New advanced phones are capable of exchanging voice, text, picture and video.

Mobile learning allows learners to take control over time and location. 'On the move' learners have the option to choose learning content based on their interest, thus making learning learner-centric. Instant access to relevant and interesting information, based on each individual learner's requirements, increases engagement because learners are able to access the information they want wherever they are.

This research, and its associated objectives, are justified by a number of factors. Most current learning content was designed for use with desktop computers and high-speed network connections. It typically contains rich media data such as images, audio, and video. Such learning content is often not suitable for presentation on devices with limited capability and limited network bandwidth.

Moreover, the widespread problem in e-learning environments is that they cannot offer personalisation for the student and that they can only present identical content to all learners. Mobile-based education is already reaching a large number of learners and it offers a valuable advantage over traditional teaching with the possibility to adapt to individual learners, which is difficult to achieve in a common teaching environment and using traditional teaching methods.

This thesis has proposed a Learning Context Framework (see Chapter 3), Learning Content Framework (see Chapter 4) and Enhanced Learner Model (see Chapter 5) for a mobile learning system by describing and examining important characteristics and requirements for such a system.

Information about the learner in form of the learner profile is essential to inform

adaptive mobile learning. It is impossible for an adaptive system to produce any acceptable outcomes without information about the learner. Therefore, learner information must be available to adaptive systems. The section of the system that manages and stores learner information is called the Enhanced Learner Model (see Chapter 5) ; it represents all aspects of the system related to the learner.

The success of an adaptive mobile learning application depends primarily on the individual learning needs and preferences of learners, therefore learner modelling is the key element to provide personalisation. This thesis has presented a context-aware (see Chapter 3) and content frameworks (see Chapter 4) that depicts the process of adapting learning content to satisfy individual learner characteristics . To effectively address the adaptation problem we must consider the problem from both sides - the learning context and the learning content - by the integration of two frameworks. The learning context framework uses a machine learning based approach for acquiring, representing, storing, reasoning and updating each learner acquired profile. The learning content framework is a system with three stages. It allows consideration of individual learning styles, learner context, application capabilities, and material structure, leading to a customisation of the type and delivery format of learning information in response to the user. This thesis has explained the major concepts/steps in learner modelling by focusing on the most significant inputs that impact the mobile learning experience. The learner model must be defined first, learner characteristics are gathered, then model construction can begin and the updating phases can be undertaken, as specified in the framework layers (see Section 3.4) and stages (see Section 4.4).

This thesis addresses a number of issues related to mobile learning, considered through the mobile learning context framework design, mobile learning content framework design, learner model structure and the delivery of adapted learning content to mobile learning applications to fit learner needs. This thesis has presented a Neuro-

Fuzzy model for delivering adapted learning content to mobile learners. The adaptation of learning content is based on Adaptive Neural-Fuzzy Inference System (ANFIS) (see Section 6.7). ANFIS has been recognised for its flexible and adaptive characteristics. ANFIS is a powerful approach to develop fuzzy systems that are capable of learning by providing IF-THEN fuzzy rules in linguistic form. ANFIS approach is adopted to determine all possible conditions which cannot be determined by using individual technique.

In general, the following aspects were considered and investigated within this thesis:

- Adaptive Context Frameworks for Mobile Learning (Chapter 3).
- Adaptive Content Frameworks for Mobile Learning (Chapter 4).
- Enhanced Learner Model for Mobile Learning (Chapter 5).
- Reasoning Layer based on ANFIS Learning Techniques and Modelling Mobile Learning Scenarios using ANFIS (Chapters 6 and 7).
- Simulation of an Adaptive Neuro-Fuzzy Inference System (ANFIS) for Modelling Learners Context in Mobile Learning (Chapter 8).

This thesis presented both the theoretical and practical issues associated with adaptive mobile learning. Initially, the relevant issues were identified and examined (theory). The outcomes of this examination were used to inform and realise a simulation solution approach (practical).

In Chapter 2, the basic principles and terminology of mobile learning were discussed. This chapter presented related literature on mobile learning and its technologies. The concepts of distance, electronic and mobile learning were discussed. The classification of mobile and wireless technologies were explained, and mobile device platforms were described. This chapter highlighted the move from electronic learning to mobile learning, and described the concepts of adaptation, personalisation and

learner model. A review of the literature covering some machine learning applications in a variety of educational institutions was also provided.

Using the basic principles explored in Chapter 2, adaptive m-learning systems require a logical structure for the process of adapting learning content. Chapter 3 presented a framework that depicts the process of adapting learning content to satisfy individual learner's needs, and a machine learning based algorithm for acquiring, representing, storing, reasoning and updating each learner acquired profile. Together, the framework and algorithm describe both the learning context and the structural issues related to the adaptive m-learning framework.

There are a number of factors that impact on a typical learning experience and many more when that learning experience becomes 'mobile'. Chapter 4 presented a framework to describe the factors that play an important role in delivering learning content to mobile learners, and their relationship with each other. The Learning Content Framework allows consideration of individual learning styles and scenarios, device and application capabilities, and material structure, leading to a customisation of the type and delivery format of learning information in response to each learner. Ultimately, the personalised response to each user (whether they are working independently or in communication with other learners) improves user engagement and the overall learning experience, as well as saving time.

In order to improve the performance of adaptive mobile learning, careful design and construction of the learner model is essential. Therefore, it is of importance to analyse in detail all the elements of the learner model. Personalisation and learner modelling, taking into consideration each learner's interests, preferences and contextual information, are important in the area of mobile learning applications. The learner model determined which learner contexts contribute most significantly to accurate implementation of adaptive mobile learning. In order to apply a learner model it is necessary

to construct, initialise and keep the user model up-to-date. Chapter 5 addressed modelling the learner and all possible contexts in an extensible way that can be used for personalisation in mobile learning. Challenges and current solutions related to learner modelling were discussed in this chapter, and the Enhanced Learner Model structure to be used in a mobile learning system was proposed.

An overview of Fuzzy Logic and Neural Network was presented in Chapter 6. When the concepts of fuzzy logic and neural networks are combined (i.e. into Neuro-Fuzzy systems), a powerful method for adaptation through supervised and un-supervised training data sets is created. This chapter also introduced the principles of Fuzzy Inference Systems (FIS), Adaptive Neuro-Fuzzy Inference Systems (ANFIS), and ANFIS hybrid learning algorithms.

An ANFIS controller was developed to deliver adaptive learning content for mobile learning application in Chapter 7. A Fuzzy Logic controller was designed and simulated to perform adaptive learning content based on different context settings. This chapter also presented the design and implementation of the modelling system that describes the learner model process of the proposed approach. ANFIS was introduced as a reasoning engine to deliver learning content for mobile learning application. This chapter also presented the ANFIS data processing, Neuro-Fuzzy model coding, simulation, assumptions, results, adaptation and training using MATLAB.

Chapter 8 presented a series of simulations that were conducted to illustrate the potential effectiveness of ANFIS with hybrid learning, for the adaptation of learning content for mobile learners. Various experiments were conducted and the appropriate sizes of the training and testing sets were determined by considering the classification accuracies. The performance of ANFIS was evaluated using standard error measurements which revealed the optimal setting necessary for better predictability. Chapter 8 showed that the feasibility of the Reasoning Layer confirms assumptions made in

Chapters 3, 4 and 5 of this thesis. These results were based on analysis of a number of inputs; the research results confirm that the m-learning application is functional.

In summary, this thesis has made significant contributions in the following areas;

- Proposing an adaptive framework to address the problems and limitations of mobile learning, contributing to the field of learning technology (Chapters 3, 4, 5, 7 and 8).

- Overcoming the limitations of using mobile learning through the development of a framework that provides personalisation and tackles adaptation using a machine learning technique which is applied to an obtained learner profile for mobile learning purposes. The learner profile contains preferences, knowledge, plans and place, and optionally other relevant aspects that are used to provide personalised adaptations. The framework depicts the process of adapting learning content to satisfy individual learner characteristics by taking into consideration the learner's needs. The framework reduces the complexity inherent to different mobile device settings and learning environments. The framework is designed to adapt learning content in a way that matches the learner's preferences, supports the learning context (location, noise level, device type, availability of resources, network), and is compatible with the learning objectives. This framework is the main contribution of the thesis (Chapter 3).

- Contributing to the efficient use of portable devices (old and new) to accomplish routine learning activities in different contexts; there is limited literature in this area. Existing literature largely focuses on adaptation using new technologies. This thesis develops a framework to describe the factors that play an important role in delivering learning content to mobile learners, and their relationship with each other. It allows consideration of individual learning styles and scenarios, device and application capabilities, and material structure, leading to customisa-

tion of the type and delivery format of learning information in response to each learner. The concern of content adaptive learning is to develop the strategies and methods for creating learning content that always meets learners' needs. To effectively address the adaptation problem, the problem has been considered from both sides - the learning context and the learning content - through the integration of two frameworks (Chapter 4).

- Modelling the learner and all possible contexts related to the learner's current situation in an extensible way that can be used for personalisation by investigating the relationship between context, learning content and learning activities considering context as the combination of the physical as well as social features of learning activities. The proposed learner model consists of five main components, namely: representation of the *Learner Status*, the *Situation Status*, the *Knowledge and Shared Properties Status* and the *Educational Activity Status*. Through the gathering of information of information about the learner, the adaptive system is able to adapt itself to the learner's characteristics and preferences. The required information is stored and managed in form of a Learner Profile and Learner Model (Chapter 5).

- Developing a reasoning engine based on Machine Learning techniques, which uses a Neuro-Fuzzy model for delivering adapted learning content to mobile learners. The ANFIS approach was applied to the adaptation of learning content to determine all possible conditions which cannot be determined through the use of individual techniques (Chapters 6, 7 and 8).

- Designing, modelling and simulating an ANFIS capable of reflecting mobile learning scenarios encountered by learners while accomplishing a learning task (Chapter 7 and 8).

9.2 Recommendation for Further Research

Based on analysis of all aspects of this research, areas requiring further work are identified below.

Due to the lack of real samples available for this research, theoretical data simulation was used to represent mobile learning scenarios. This simulation data identified good performance in relation to different kinds of training settings (the number of membership functions, type of membership functions, the training data sample and the Epoch number) in order to deliver adapted learning content based on learners' needs. Based on the data collected in this thesis, the ANFIS system is likely to be able to deal with real samples and work correctly. Future work is important to confirm the correct performance of the system, and to continue to improve the system accuracy, performance and response in relation to different real environments.

After confirmation of the ANFIS system performance in real environments, full implementation of the proposed adaptive mobile learning system for designing and adapting learning content for mobile learners can commence (Chapter 3). This will require designing comprehensive educational content according to the approach presented in this thesis (Chapter 4 and 5), implementing the adaptation engine (Chapter 7) on the learner profile ,and finally testing the effectiveness of the system, both quantitatively and qualitatively, using a series of simulations and a small number of human users to work with the system.

References

[1] Merriam-webster's online dictionary. `http://www.merriam-webster.com/dictionary/adapt` – last visited 7^{th} June 2011.

[2] Oxford english dictionary. `http://oald8.oxfordlearnersdictionaries.com/dictionary/adaptive` – last visited 7^{th} June 2011.

[3] Fuzzy logic toolbox user's guide, December 30 2008. `http://www.mathworks.com/access/helpdesk_r13/help/pdf_doc/fuzzy/fuzzy_tb.pdf`.

[4] Mike Addlesee, Rupert W. Curwen, Steve Hodges, Joseph Newman, Pete Steggles, Andy Ward, and Andy Hopper. Implementing a sentient computing system. *IEEE Computer*, 34(8):50–56, 2001.

[5] Epstein R.G Aiken, R.M. Ethical guidelines for artificial intelligece in education: Starting a conversation. In *International Journal of Artificial Intelligence in Education*, 11, 163-176, 2000.

[6] A. Al-Hmouz and A. Freeman. Learning on location: An adaptive mobile learning content framework. In *IEEE International Symposium on Technology and Society (ISTAS), 2010*, pages 450 –456, 2010.

[7] Ahmed Al-Hmouz, Jun Shen, and Jun Yan. A machine learning based framework for adaptive mobile learning. In Marc Spaniol, Qing Li, Ralf Klamma, and

Rynson W. H. Lau, editors, *ICWL*, volume 5686 of *Lecture Notes in Computer Science*, pages 34–43. Springer, 2009.

[8] Ahmed Al-Hmouz, Jun Shen, Jun Yan, and Rami Al-Hmouz. Enhanced learner model for adaptive mobile learning. In Gabriele Kotsis, David Taniar, Eric Pardede, Imad Saleh, and Ismail Khalil, editors, *IIWAS*, The Association for Computing Machinery, pages 781–784. ACM, 2010.

[9] Ahmed Al-Hmouz, Jun Shen, Jun Yan, and Rami Al-Hmouz. Modeling mobile learning system using ANFIS. In *ICALT*, pages 378–380. IEEE Computer Society, 2011.

[10] G. Andaloro and L. Bellomonte. Student knowledge and learning skill modeling in the learning environment 'forces'. *Comput. Educ.*, 30(3-4):209–217, 1998.

[11] Alessandro Souza Ferreira Rubim De Assis. *Inferring dynamic learner behavior for user modeling in continuously adapting hypermedia.* PhD thesis, Troy, NY, USA, p. 27, 2006. AAI3220183.

[12] Jill Attewell. *Mobile technologies and learning: A technology update and m-learning project summary.* Learning and Skills Development Agency, London, 0 2005. Attewell 2005 - Mobile technologies and learning k, 27.05.2007.

[13] Jill Attewell and Carol Savill-Smith. m-learning and social inclusion - focusing on the learners and learning, 5 2003.

[14] Waller B. and Wilson J. A definition for e-learning. *BAOL Open Learning Today*, Issue 58, October, 2001.

[15] Yunju Baek, Hyuntae Cho, and B. Kim. Uses of learning objects in a wireless internet based learning system. In *ICCE*, pages 898–900, 2002.

[16] Afzal Ballim and Yorick Wilks. Beliefs, stereotypes and dynamic agent modeling. *User Modeling and User-Adapted Interaction*, 1(1):33–65, 1991.

[17] Maria Barra, Delfina Malandrino, and Vittorio Scarano. "common" web paths in a group adaptive system. In *Hypertext*, pages 218–219. ACM, 2003.

[18] Ian Beaumont. User modeling in the interactive anatomy tutoring system ANATOM-TUTOR. *User Modeling and User-Adapted Interaction*, 4(1):21–45, 1994.

[19] J. Beck, M. Stern, and B. P. Woolf. Using the student model to control problem difficulty. In Anthony Jameson, Cécile Paris, and Carlo Tasso, editors, *Proceedings of the 6th International Conference on User Modeling (UM-97)*, volume 383 of *CISM*, pages 277–288, Wien, June 02–05 1997. Springer.

[20] Joseph Beck, Mia Stern, and Erik Haugsjaa. Applications of AI in education. pp.11 - 15, 1996.

[21] Hamid R. Berenji. A reinforcement learning-based architecture for fuzzy logic control. *Int. J. Approximate Reasoning*, 6:267–292, February 1992.

[22] Hugues Bersini, Jean-Pierre Nordvik, and Andrea Bonarini. A simple direct adaptive fuzzy controller derived from its neural equivalent. In *Proc. IEEE Int. Conf. on Fuzzy Systems 1993*, pages 345–350, San Francisco, March 1993.

[23] Jan Blom, Anu Kankainen, Tomi Kankainen, and Sauli Tiitta. Location-Aware Multiuser Messaging: Exploring the Evolution of Mobile Text-Based Communication Services. HIIT Technical Report 2003-2, Institute of Information Technology and Creadesign, Helsinki, August 2003. `http://www.hiit.fi/files/admin/publications/Technical_Reports/hiit2003-2.pdf` – last visited 7^{th} January 2011.

[24] A. Bobick, S. Intille, J. Davis, F. Baird, C. Pinhanez, L. Campbell, Y. Ivanov, A. Schutte, and A. Wilson. The KidsRoom: A perceptually-based interactive and immersive story environment. Perceptual Computing Group Technical Report 398, MIT Media Laboratory, E15, 20 Ames Street, Cambridge MA 02139, December 1996.

[25] Fabian Bohnert. Constraint-aware user modelling and personalisation in physical environments. In *Adjunct Proceedings of the Sixth International Conference on Pervasive Computing (Pervasive-08)*, pages 167–172, 2008.

[26] L. Breiman, J. Friedman, R. Olshen, and C. Stone. *Classification and Regression Trees*. Wadsworth and Brooks, Monterey, CA, 1984.

[27] J. S. Brown, A. Collins, and P. Duguid. Situated cognition and the culture of learning. *Educational Researcher*, 18:32–34, 1989.

[28] T.H. Brown. The role of m-learning in the future of e-learning in africa. Presentation at the 21st ICDE World Conference, 2003.

[29] P. Brusilovsky. Adaptive navigation support in educational hypermedia: the role of student knowledge level the case for meta-adaption. *British Journal of Educational Technology*, 34(4):487–497, 2003.

[30] P Brusilovsky. The construction and application of student models in intelligent tutoring systems, journal of computer and system sciences international, 32(1), 70 - 89, 1994.

[31] Peter Brusilovsky, John Eklund, and Elmar Schwarz. Web-based education for all: a tool for development adaptive courseware. *Computer Networks and ISDN Systems*, 30(1–7):291–300, April 1998.

[32] Peter Brusilovsky and Mark T. Maybury. From adaptive hypermedia to the adaptive Web. *Communications of the ACM*, 45(5):30–33, May 2002.

[33] Peter Brusilovsky and Leonid Pesin. Isis-tutor: An adaptive hypertext learning environment for cds/isis users, Online Proceedings of CLCE'94, Joensuu, Finland, 1994.

[34] Susan Bull, Yanchun Cui, Harald Röbig, and Mike Sharples. Adapting to different needs in different locations: Handheld computers in university education. In *WMTE*, pages 48–52. IEEE Computer Society, 2005.

[35] Cory J. Butz, Shan Hua, and R. Brien Maguire. A web-based bayesian intelligent tutoring system for computer programming. *Web Intelligence and Agent Systems*, 4(1):77–97, 2006.

[36] David J. Carmichael, Judy Kay, and Bob Kummerfeld. Consistent modelling of users, devices and sensors in a ubiquitous computing environment. *User Model. User-Adapt. Interact*, 15(3-4):197–234, 2005.

[37] T. Chan, M. Sharples, G. Vavoula, and P. Lonsdale. Educational metadata for mobile learning. 2004.

[38] Guanling Chen and David Kotz. A survey of context-aware mobile computing research. Technical Report TR2000-381, Dartmouth College, Computer Science, Hanover, NH, November 2000.

[39] Qiyang Chen, Anthony F. Norcio, and John Wang. Neural network based stereotyping for user profiles. *Neural Computing and Applications*, 9(4):259–265, 2000.

[40] Yuh-Shyan Chen, Tai-Chien Kao, Jang-Ping Sheu, and Chao-Yu Chiang. A mobile scaffolding-aid-based bird-watching learning system. In Marcelo Milrad,

Heinz Ulrich Hoppe, and Kinshuk, editors, *WMTE*, pages 15–22. IEEE Computer Society, 2002.

[41] Kwang Bo Cho and Bo-Hyuen Wang. Radial basis function based adaptive fuzzy systems and their applications to system identification and prediction. *Fuzzy Sets Syst.*, 83:325–339, November 1996.

[42] Li-Der Chou, Ching-Chao Lee, Ming-Yu Lee, and Chun-Yen Chang. A tour guide system for mobile learning in museums. In Jeremy Roschelle, Tak-Wai Chan, Kinshuk, and Stephen J. H. Yang, editors, *WMTE*, pages 195–196. IEEE Computer Society, 2004.

[43] Cristina Conati, Abigail S. Gertner, and Kurt VanLehn. Using bayesian networks to manage uncertainty in student modeling. *User Model. User-Adapt. Interact*, 12(4):371–417, 2002.

[44] Nigel Davies and Hans-Werner Gellersen. Beyond prototypes: Challenges in deploying ubiquitous systems. *IEEE Pervasive Computing*, 1:26–35, 2002.

[45] Dey, Anind K., Abowd, Gregory D., and Daniel Salber. A conceptual framework and a toolkit for supporting the rapid prototyping of context-aware applications. *Human-Computer Interaction*, 16(2/4):97–166, 2001.

[46] Anind K. Dey, Gregory D. Abowd, and Andrew Wood. Cyberdesk: A framework for providing self-integrating context-aware services. In *IUI*, pages 47–54, 1998.

[47] Henry Kautz Dieter, Dieter Fox, Oren Etzioni, Gaetano Borriello, and Larry Arnstein. An overview of the assisted cognition project. In *In AAAI-2002 Workshop on Automation as Caregiver: The Role of Intelligent Technology in Elder*, 2002.

[48] Rita S. Dunn and Kenneth J. Dunn. *Teaching Students Through Their Individual Learning Styles: A Practical Approach.* Prentice Hall College Div, illustrated edition edition, 1978.

[49] Sanaz Fallahkhair, Lyn Pemberton, and Richard N. Griffiths. Dual device user interface design for ubiquitous language learning: Mobile phone and interactive television (iTV). In *WMTE*, pages 85–92. IEEE Computer Society, 2005.

[50] Laurene Fausett. *Fundamentals of Neural Networks.* Prentice-Hall, Inc., 1994.

[51] R. M. Felder and L. K. Silverman. Learning and teaching styles in engineering education. *Engineering Education*, 78(7):674, 1988.

[52] Gerhard Fischer. User modeling in human-computer interaction. *User Model. User-Adapt. Interact*, 11(1-2):65–86, 2001.

[53] E Frias-martinez, S Y Chen, and X Liu. Survey of data mining approaches to user modeling for adaptive hypermedia. November 2006.

[54] E. Frias-Martinez, G. Magoulas, S. Chen, and R. Macredie. Modeling human behavior in user-adaptive systems: recent advances using soft computing techniques. 2005.

[55] R. Fuller. *Introduction to Neuro-Fuzzy Systems.* Advances in Soft Computing Series, Springer-Verlag, Berlin/Heildelberg, 2000.

[56] Patricio García, Analía Amandi, Silvia N. Schiaffino, and Marcelo R. Campo. Evaluating bayesian networks' precision for detecting students' learning styles. *Computers & Education*, 49(3):794–808, 2007.

[57] D. Gayeski. *Learning unplugged: Using mobile technologies for organizational training and performance improvement.* In New York: AMACOM, 2002.

[58] Tiong-Thye Goh and Kinshuk. A discussion on mobile agent based mobile web-based ITS. In *ICCE*, pages 1514–1515, 2002.

[59] William G. Griswold, Patricia Shanahan, Steven W. Brown, Robert Boyer, Matt Ratto, R. Benjamin Shapiro, and Tan Minh Truong. ActiveCampus: Experiments in community-oriented ubiquitous computing. *Computer*, 37(10):73–81, October 2004.

[60] L. Hawkes, S. Derry, and E. Rundenesteiner. Individualized tutoring using an intelligent, fuzzy, temporal relational database. *International J. of Man and Machine*, 33:409–429, 1990.

[61] Lois Wright Hawkes and Sharon J. Derry. Advances in local student modeling using informal fuzzy reasoning. *Int. J. Hum.-Comput. Stud*, 45(6):697–722, 1996.

[62] Mariana Hentea, Mary Jo Shea, and Lisa Pennington. A perspective on fulfilling the expectations of distance education. In *Proceedings of the 4th conference on Information technology curriculum*, CITC4 '03, New York, NY, USA, 160–167, 2003. ACM.

[63] Eelco Herder. *Forward, back and home again : analyzing user behavior on the web*. PhD thesis, Enschede, 2006.

[64] Hubertus Hohl, Heinz-Dieter Böcker, and Rul Gunzenhäuser. Hypadapter: An adaptive hypertext system for exploratory learning and programming. *User Modeling and User-Adapted Interaction*, 6(2-3):131–156, 1996.

[65] Sherry Hsi. The electronic guidebook: A study of user experiences using mobile web content in a museum setting. In Marcelo Milrad, Heinz Ulrich Hoppe, and Kinshuk, editors, *WMTE*, pages 48–54. IEEE Computer Society, 2002.

[66] Chenn-Jung Huang, Ming-Chou Liu, San-Shine Chu, and Chih-Lun Cheng. An intelligent learning diagnosis system for web-based thematic learning platform. *Computers & Education*, 48(4):658–679, 2007.

[67] Hidetomo Ichihashi and I. Burhan Türksen. A neuro-fuzzy approach to data analysis of pairwise comparisons. *Int. J. Approx. Reasoning*, 9(3):227–248, 1993.

[68] Anthony Jameson. Numerical uncertainty management in user and student modeling: An overview of systems and issues. *User Modeling and User-Adapted Interaction*, 5(3-4):193–251, 1996.

[69] Paris C. Jameson, A. and Tasso. readers guide, in user modeling, Proc of the Sixth International Conference UM97, Springer-Verlag, New York.1997.

[70] J. S. R. Jang and C. T. Sun. Neuro-fuzzy modeling and control. *Proc. IEEE*, 83(3):378–406, 1995.

[71] J. S. R. Jang, C. T. Sun, and E. Mizutani. *Neuro-Fuzzy and Soft Computing*. Prentice Hall, 1997.

[72] Jyh-Sing Roger Jang. ANFIS: Adaptive-network-based fuzzy inference system. *IEEE Transactions on Systems, Man, and Cybernetics*, 23:665–684, 1993.

[73] Eija Kaasinen. User needs for location-aware mobile services. *Personal and Ubiquitous Computing*, 7(1):70–79, 2003.

[74] Robert Kass. Building a user model implicitly from a cooperative advisory dialog. *User Model. User-Adapt. Interact*, 1(3-4):203–258, 1991.

[75] J. W Keefe. *Learning Style: An Overview, in J. W. Keefe (ed.), Student learning styles: Diagnosing and prescribing programs.* NASSP, 1979.

[76] Desmond Keegan. The future of learning: From elearning to mlearning, November 2002.

[77] Harri Ketamo. mlearning for kindergarten's mathematics teaching. In Marcelo Milrad, Heinz Ulrich Hoppe, and Kinshuk, editors, *WMTE*, pages 167–168. IEEE Computer Society, 2002.

[78] Harri Ketamo. xtask - adaptable working environment. In Marcelo Milrad, Heinz Ulrich Hoppe, and Kinshuk, editors, *WMTE*, pages 55–62. IEEE Computer Society, 2002.

[79] Sung Woo Kim, Sang Hyun Park, Jung Bong Lee, Young Kyu Jin, Hyun mi Park, Amy Chung, Seung Eok Choi, and Woo Sik Choi. Sensible appliances: applying context-awareness to appliance design. *Personal and Ubiquitous Computing*, 8(3-4):184–191, 2004.

[80] Won Kim. Personalization: Definition, status, and challenges ahead. *Journal of Object Technology*, 1(1):29–40, 2002.

[81] Alfred Kobsa. A taxonomy of beliefs and goals for user models in dialog systems. In Alfred Kobsa and Wolfgang Wahlster, editors, *User models in dialog systems*, pages 52–68. Springer, Berlin, 1989.

[82] Alfred Kobsa. Generic user modeling systems. *User modelling and user-adapted interaction*, 11:49–63, 2001.

[83] Alfred Kobsa, Jrgen Koenemann, and Wolfgang Pohl. Personalized hypermedia presentation techniques for improving online customer relationships, March 12 2001.

[84] Alfred Kobsa, Dietmar Müller, and Andreas Nill. KN-AHS: An adaptive hypertext client of the user modeling system BGP-MS. In *Proceedings of the 4th*

International Conference on User Modeling, pages 99–106, Bedford, MA, USA, August 1994. MITRE Corporation.

[85] Nora Koch. Software engineering for adaptive hypermedia systems: Reference model, modeling techniques and development process. *Softwaretechnik-Trends*, 21(1), p. 51, 2001.

[86] Anders Kofod-Petersen and Jörg Cassens. Using activity theory to model context awareness. In Thomas Roth-Berghofer, Stefan Schulz, and David B. Leake, editors, *MRC*, volume 3946 of *Lecture Notes in Computer Science*, pages 1–17. Springer, 2005.

[87] T. Kohonen. The self-organising map. *Proceedings of the IEEE*, 78(9):1464–1480, 1990.

[88] Andr J. Kok. A review and synthesis of user modelling in intelligent systems. *The Knowledge Engineering Review*, 6:pp 21–47, 1991.

[89] Luigi Di Lascio, Antonio Gisolfi, and Vincenzo Loia. Uncertainty processing in user-modeling activity. *Inf. Sci*, 106(1-2):25–47, 1998.

[90] Franz Lehner, Holger Nösekabel, and Hans Lehmann. Wireless e-learning and communication environment: Welcome at the university of regensburg. In *M-Services*, pp. 23-41, 2002.

[91] Carla Limongelli, Filippo Sciarrone, Marco Temperini, and Giulia Vaste. Adaptive learning with the LS-plan system: A field evaluation. *TLT*, 2(3):203–215, 2009.

[92] C. T. (Ching Tai) Lin and C. S. G. Lee. *Neural fuzzy systems: a neuro-fuzzy synergism to intelligent systems.* Prentice-Hall PTR, pub-PHPTR:adr, 1996.

[93] David J.C. MacKay and Cavendish Laboratory. Bayesian non-linear modelling for the prediction competition. In *In ASHRAE Transactions, V.100, Pt.2*. ASHRAE, pp. 1053-1062, 1994.

[94] Jihen Malek, Mona Laroussi, and Alain Derycke. A multi-layer ubiquitous middleware for bijective adaptation between context and activity in a mobile and collaborative learning. In *ICSNC*, page 39. IEEE Computer Society, 2006.

[95] Natalia Marmasse and Chris Schmandt. Location-aware information delivery with commotion. In Peter J. Thomas and Hans-Werner Gellersen, editors, *HUC*, volume 1927 of *Lecture Notes in Computer Science*, pages 157–171. Springer, 2000.

[96] Estefanía Martín, Nuria Andueza, and Rosa M. Carro. Architecture of a system for context-based adaptation in M-learning. In *ICALT*, pages 252–254. IEEE Computer Society, 2006.

[97] Estefanía Martín, Nuria Andueza, Rosa M. Carro, and Pilar Rodríguez. Recommending activities in collaborative m-learning. *Lecture Notes in Learning and Teaching (ISSN 1649-8623)*, pages 197–204, 2006.

[98] Estefanía Martín, Rosa M. Carro, and Pilar Rodríguez. A mechanism to support context-based adaptation in M-learning. In Wolfgang Nejdl and Klaus Tochtermann, editors, *EC-TEL*, volume 4227 of *Lecture Notes in Computer Science*, pages 302–315. Springer, 2006.

[99] Michael F. McTear. User modelling for adaptive computer systems: a survey of recent developments. *Artif. Intell. Rev*, 7(3-4):157–184, 1993.

[100] Felix Mdritscher, Victor M. Garca-Barrios, and Christian Gtl. The past, the present and the future of adaptive e-learning. In *In: Proceedings of the International Conference on Interactive Computer Aided Learning (ICL2004)*, 2004.

[101] Fatma Elsayed Meawad and Geneen Stubbs. An initial framework for implementing and evaluating probabilistic adaptivity in mobile learning. In *ICALT*, pages 166–168. IEEE Computer Society, 2005.

[102] P. Mellow. The media generation: Maximise learning by getting mobile. In *Proceedings of ASCILITE*, 2005.

[103] Eva Millán and José-Luis Pérez de-la Cruz. A bayesian diagnostic algorithm for student modeling and its evaluation. *User Model. User-Adapt. Interact*, 12(2-3):281–330, 2002.

[104] Tom M. Mitchell. *Machine Learning.* WCB/McGraw-Hill, Boston, Massachusetts, 1997.

[105] Yasuto Nakanishi, Takayuki Tsuji, Minoru Ohyama, and Katsuya Hakozaki. Context aware messaging service: A dynamical messaging delivery using location information and schedule information. *Personal and Ubiquitous Computing*, 4(4):221–224, 2000.

[106] Detlef Nauck. A fuzzy perceptron as a generic model for neuro–fuzzy approaches. In *Proc. Fuzzy–Systeme'94*, Munich, October 1994.

[107] Andronikos Nedos, Alex O'Connor, Siobhán Clarke Graham Abell, and Vinny Cahill. LATTE: Location and time triggered email. In *International Conference on Pervasive Computing and Communications, PCC-04*, pages 813–819. CSREA Press, June 2004.

[108] Illah R. Nourbakhsh. Robots and education in the classroom and in the museum: On the study of robots, and robots for study, May 02 2000.

[109] Kristóf Nyíri. Towards a philosophy of M-learning. In Marcelo Milrad, Heinz Ulrich Hoppe, and Kinshuk, editors, *WMTE*. IEEE Computer Society, pp.121-124, 2002.

[110] Hiroaki Ogata and Yoneo Yano. Context-aware support for computer-supported ubiquitous learning. In *Proceedings of the 2nd IEEE International Workshop on Wireless and Mobile Technologies in Education (WMTE'04)*, WMTE '04, pages 27–, Washington, DC, USA, 2004. IEEE Computer Society.

[111] Mattias Ohlsson, Carsten Peterson, Hong Pi, Thorsteinn Rögnvaldsson, and Bo Söderberg. Predicting system loads with artificial neural networks — methods and results from "the Great Energy Predictor Shootout". In *1994 Annual Proceedings of the American Society of Heating, Refrigerating and Air-Conditioning Engineers, Inc.*, volume 100, pp. 1063-1074, 1994.

[112] Reinhard Oppermann, editor. *Adaptive user support: ergonomic design of manually and automatically adaptable software*. L. Erlbaum Associates Inc., Hillsdale, NJ, USA, 1994.

[113] Morten Flate Paulsen. *Teaching techniques for computer-mediated communication*. PhD thesis, 1998. AAI9915916.

[114] C. L. Posey and Lois Wright Hawkes. Neural networks applied to knowledge acquisition in the student model. *Inf. Sci*, 88(1-4):275–315, 1996.

[115] Elaine Rich. Stereotypes and user modeling. In Alfred Kobsa and Wolfgang Wahlster, editors, *User Models in Dialog Systems*, pages 35–51. Springer-Verlag, Berlin, 1989.

[116] G. Eggan S. Katz, A. Lesgold and M. Gordin. Modeling the student in sherlock ii. *Int'l J. Artificial Intelligence in Education*, 3:495–518, 1992.

[117] J. s. Roger Jang and Ned Gulley. MATLAB fuzzy logic toolbox - user's guide version 1, 2007.

[118] Glória Curilem Saldías, Andréa R. Barbosa, and Fernando Mendes de Azevedo. Intelligent tutoring systems: Formalization as automata and interface design using neural networks. *Computers & Education*, 49(3):545–561, 2007.

[119] Corina Sas and Ronan Reilly. G.: A connectionist model of spatial knowledge acquisition in a virtual environment, 2003.

[120] Carol Savill-Smith. The use of palmtop computers for learning: a review of the literature. *British Journal of Educational Technology*, 36(3):567–568, 2005.

[121] Eileen Scanlon, Ann Jones, and Jenny Waycott. Mobile technologies: prospects for their use in learning in informal science settings. November 23 2007.

[122] Bill N. Schilit. *A System Architecture for Context-Aware Mobile Computing.* PhD thesis, May 1995. http://sandbox.parc.xerox.com/parctab/.

[123] Sushil K. Sharma and Fred L. Kitchens. Web services architecture for m-learning. *Electronic, Journal on e-Learning*, 2:2004.

[124] Mike Sharples. Big issues in mobile learning: Report of a workshop by the kaleidoscope network of excellence mobile learning initiative. Technical report, University of Nottingham, 2006.

[125] Yan Shi and Masaharu Mizumoto. A new approach of neuro-fuzzy learning algorithm for tuning fuzzy rules. *Fuzzy Sets Syst.*, 112:99–116, May 2000.

[126] Yan Shi and Masaharu Mizumoto. Some considerations on conventional neuro-fuzzy learning algorithms by gradient descent method. *Fuzzy Sets Syst.*, 112:51–63, May 2000.

[127] Jyh shing Roger Jang. Input selection for ANFIS learning, September 07 1996.

[128] Regina Stathacopoulou, Maria Grigoriadou, George D. Magoulas, and Denis Mitropoulos. A neuro-fuzzy approach in student modeling. In Peter Brusilovsky, Albert T. Corbett, and Fiorella de Rosis, editors, *User Modeling*, volume 2702 of *Lecture Notes in Computer Science*, pages 337–341. Springer, 2003.

[129] Shanghua Sun, Mike Joy, and Nathan Griffiths. To support adaptivity in agent-based learning systems - the use of learning objects and learning style. In *ICALT*, pages 846–847. IEEE Computer Society, 2005.

[130] Antti Syvänen, Russell Beale, Mike Sharples, Mikko Ahonen, and Peter Lonsdale. Supporting pervasive learning environments: Adaptability and context awareness in mobile learning. In *WMTE*, pages 251–253. IEEE Computer Society, 2005.

[131] Michael Szabo. Cmi theory and practice: Historical roots of learning managment systems. In Margaret Driscoll and Thomas C. Reeves, editors, *Proceedings of World Conference on E-Learning in Corporate, Government, Healthcare, and Higher Education 2002*, pages 929–936, Montreal, Canada, 2002. AACE.

[132] H. Takagi and I. Hayashi. NN-driven fuzzy reasoning. *Int. J. Approximate Reasoning*, 5:191–212, 1991.

[133] Peter Tarasewich. Designing mobile commerce applications. *Communications of the ACM*, 46(12):57–60, December 2003.

[134] Jasper A. Taylor, Jean Carletta, and Chris Mellish. Requirements for belief models in cooperative dialogue. *User Modeling and User-Adapted Interaction*, 6(1):23–68, 1996.

[135] Knut Ola Topland. Mobile learning: Technological challenges on multi-channel e-learning services. Master's thesis, Saarbrücken, Germany, Germany, 2009.

[136] Pomportsis A Triantafyllou E., Demetriadis S. Adaptive hypermedia and cognitive styles: Can performance be influenced?. In *In Proceedings of the 9th Panhellenic Conference in Informatics, Thessaloniki.*, pp. 510 - 518, 2003.

[137] Anna Trifonova and Marco Ronchetti. A general architecture to support mobility in learning. In Kinshuk, Chee-Kit Looi, Erkki Sutinen, Demetrios G. Sampson, Ignacio Aedo, Lorna Uden, and Esko Kähkönen, editors, *ICALT*. IEEE Computer Society, 2004.

[138] Brigitte Trousse. Evaluation of the prediction capability of a user behaviour mining approach for adaptive web sites., April 05 2000.

[139] Victoria Tsiriga and Maria Virvou. Initializing student models in web-based ITSs: A generic approach. In *ICALT*, pages 42–46. IEEE Computer Society, 2003.

[140] Victoria Tsiriga and Maria Virvou. A framework for the initialization of student models in web-based intelligent tutoring systems. *User Model. User-Adapt. Interact*, 14(4):289–316, 2004.

[141] Lorna Uden. Activity theory for designing mobile learning. *Int. J. Mob. Learn. Organ.*, 1:81–102, October 2007.

[142] T.A. Urdan and C.C. Weggen. E-learning corporate e-learning: Exploring a new frontier, WR Hambrecht + Co research reports 2(10) - eServices: Internet Services, 2000.

[143] Kurt Vanlehn and Zhendong Niu. Bayesian student modeling, user interfaces and feedback: A sensitivity analysis, 2008.

[144] Giasemi N. Vavoula, Paul Lefrere, Claire O'Malley, Mike Sharples, and Josie Taylor. Producing guidelines for learning, teaching and tutoring in a mobile environment. In Jeremy Roschelle, Tak-Wai Chan, Kinshuk, and Stephen J. H. Yang, editors, *WMTE*, pages 173–176. IEEE Computer Society, 2004.

[145] M. Felisa Verdejo, Carlos Celorrio, Emilio Lorenzo, and Teresa Sastre-Toral. An educational networking infrastructure supporting ubiquitous learning for school students. In *ICALT*, pages 174–178. IEEE Computer Society, 2006.

[146] D. Amandi A. Villaverde, J. E. Godoy. Learning styles recognition in e-learning environments with feed-forward neural networks. *Journal of Computer Assisted Learning*, 22:197–206, 2006.

[147] Dimitris Vrakas, Dimitris Vrakas, and Ioannis Pl Vlahavas. *Artificial Intelligence for Advanced Problem Solving Techniques*. Information Science Reference - Imprint of: IGI Publishing, Hershey, PA, 2008.

[148] L. Wang and J. Mendel. Back-propagation fuzzy system as nonlinear dynamic system identifiers. In *Proceedings of IEEE International Conferenceon Fuzzy Systems, pp.14091416*, 1992.

[149] Yuan-Kai Wang. Context awareness and adaptation in mobile learning. In Jeremy Roschelle, Tak-Wai Chan, Kinshuk, and Stephen J. H. Yang, editors, *WMTE*, pages 154–158. IEEE Computer Society, 2004.

[150] Wang, Jingtao and Mankoff, Jennifer. Theoretical and architectural support for input device adaptation. In *Proceedings of the 2003 ACM Conference on Universal Usability*, Design, pages 85–92, 2003.

[151] Kai Warendorf and Su Jen Tsao. Application of fuzzy logic techniques in the bss1 tutoring system. *J. Artif. Intell. Educ.*, 8(1):113–146, 1997.

[152] Geoffrey I. Webb, Michael J. Pazzani, and Daniel Billsus. Machine learning for user modeling. *User Model. User-Adapt. Interact*, 11(1-2):19–29, 2001.

[153] Carine Webber. From errors to conceptions - an approach to student diagnosis. In James C. Lester, Rosa Maria Vicari, and Fábio Paraguaçu, editors, *Intelligent Tutoring Systems*, volume 3220 of *Lecture Notes in Computer Science*, pages 710–719. Springer, 2004.

[154] G. Weber and M. Specht. User modeling and adaptive navigation support in WWW-based tutoring systems. In Anthony Jameson, Cécile Paris, and Carlo Tasso, editors, *Proceedings of the 6th International Conference on User Modeling (UM-97)*, volume 383 of *CISM*, pages 289–300, Wien, June 02–05 1997. Springer.

[155] Stephan Weibelzahl. Evaluation of adaptive systems. In Mathias Bauer, Piotr Gmytrasiewicz, and Julita Vassileva, editors, *User Modeling 2001*, volume 2109 of *Lecture Notes in Computer Science*, pages 292–294. Springer Berlin / Heidelberg, 2001.

[156] Ryen W. White, Peter Bailey, and Liwei Chen. Predicting user interests from contextual information. In James Allan, Javed A. Aslam, Mark Sanderson, ChengXiang Zhai, and Justin Zobel, editors, *SIGIR*, pages 363–370. ACM, 2009.

[157] Hongjing Wu, de Kort, Erik, and De Bra, Paul. Design issues for general-purpose adaptive hypermedia systems. In *Proceedings of the Twelfth ACM Conference on Hypertext*, 4a – Adaptive Hypertext, pages 141–150, 2001.

[158] Michalis Xenos. Prediction and assessment of student behaviour in open and distance education in computers using bayesian networks. *Computers & Education*, 43(4):345–359, 2004.

[159] R. R. Yager and D. P. Filev. Generation of fuzzy rules by mountain clustering. *Journal of Intelligent and Fuzzy Systems*, 2:209–219, 1994.

[160] Stephen J. H. Yang and Irene Y. L. Chen. Providing context aware learning services to learners with portable devices. In *ICALT*, pages 840–842. IEEE Computer Society, 2006.

[161] Ramin Yasdi. A literature survey on applications of neural networks for human-computer interaction. *Neural Computing and Applications*, 9(4):245–258, 2000.

[162] Jane Yau and Mike Joy. Architecture of a context-aware and adaptive learning schedule for learning java. In J. Michael Spector, Demetrios G. Sampson, Toshio Okamoto, Kinshuk, Stefano A. Cerri, Maomi Ueno, and Akihiro Kashihara, editors, *ICALT*, pages 252–256. IEEE Computer Society, 2007.

[163] Shiou-Wen Yeh and Jia-Jiunn Lo. Assessing metacognitive knowledge in web-based CALL: a neural network approach. *Computers & Education*, 44(2):97–113, 2005.

[164] L. Zadeh. Fuzzy sets. *Information and Control*, 8:338–353, 1965.

[165] Telmo Zarraonandia, Camino Fernández, Paloma Díaz, and Jorge Torres. On the way of an ideal learning system adaptive to the learner and her context. In *ICALT*, pages 641–643. IEEE Computer Society, 2005.

[166] Jie Zhang and Ali A. Ghorbani. GUMSAWS: A generic user modeling server for adaptive web systems. In *CNSR*, pages 117–124. IEEE Computer Society, 2007.