Personalising Learning with Dynamic Prediction and Adaptation to Learning Styles in a Conversational Intelligent Tutoring System

ANNABEL MARIE LATHAM

A thesis submitted in partial fulfilment of the requirements of the Manchester Metropolitan University for the degree of Doctor of Philosophy

School of Computing, Mathematics and Digital Technology
the Manchester Metropolitan University

December 2011

Abstract

This thesis presents research that combines the benefits of intelligent tutoring systems (ITS), conversational agents (CA) and learning styles theory by constructing a novel conversational intelligent tutoring system (CITS) called Oscar. Oscar CITS aims to imitate a human tutor by implicitly predicting individuals' learning style preferences and adapting its tutoring style to suit them during a tutoring conversation.

ITS are computerised learning systems that intelligently personalise tutoring based on learner characteristics such as existing knowledge and learning style. ITS are traditionally student-led, hyperlink-based learning systems that adapt the presentation of learning resources by reordering or hiding links. Research suggests that students learn more effectively when instruction matches their learning style, which is typically modelled explicitly using questionnaires or implicitly based on behaviour. Learning is a social process and natural language interfaces to ITS, such as CAs, allow students to construct knowledge through discussion. Existing CITS adapt tutoring according to student knowledge, emotions and mood, however no CITS adapts to learning styles.

Oscar CITS models a human tutor by directing a tutoring conversation and automatically detecting and adapting to an individual's learning styles. Original methodologies and architectures were developed for constructing an Oscar Predictive CITS and an Oscar Adaptive CITS. Oscar Predictive CITS uses knowledge captured from a learning styles model to dynamically predict learning styles from an individual's tutoring dialogue. Oscar Adaptive CITS applies a novel adaptation algorithm to select the best tutoring style for each tutorial question. The Oscar CITS methodologies and architectures are independent of the learning styles model and subject domain. Empirical studies involving real students have validated the prediction and adaptation of learning styles in a real-world teaching/learning environment. The results show that learning styles can be successfully predicted from a natural language tutoring dialogue, and that adapting the tutoring style significantly improves learning performance.

Acknowledgements

This PhD research was funded by the Engineering and Physical Science Research Council (EPSRC). I would like to thank ConvAgent Limited for allowing me the use of their InfoChat conversational agent.

Many people supported me during my PhD research, and I am grateful for their encouragement and understanding.

Firstly, I would like to thank my project supervisors, Dr. Keeley Crockett, Dr. David McLean and Dr. Bruce Edmonds for their direction, advice and support. A special thank you to my Director of Studies, Keeley, for getting me back on track when my energy and motivation left me.

Most of all, I wish to thank my family who have shared with me the ups and downs of my PhD studies. Thank you to my parents, for giving me the confidence and opportunities to achieve my goals and for having faith in me. I'm sad that my mum didn't live to see me complete my PhD, but she always believed I would finish it. Thank you to John for his understanding, encouragement and belief in me, for cheering me up when life got me down and for taking care of our children when I could not be there. Finally, thank you to my wonderful children Oscar and Juliet for not complaining (too much) about my long working hours, for highlighting the triviality of my problems and for adding such infectious joy and enthusiasm into my life.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

ACITS	Adaptive Conversational Intelligent Tutoring System
CA	Conversational Agent
CITS	Conversational Intelligent Tutoring System
FS	Felder-Silverman learning styles model
ILS	Index of Learning Styles
ITS	Intelligent Tutoring Systems
PCITS	Predictive Conversational Intelligent Tutoring System

Chapter 1 Introduction

This thesis presents research investigating whether an automated conversational agent tutoring system can mimic a human tutor by picking up cues from individual learners and adapting its tutoring style to suit them during a tutoring conversation. The research involves the capture of knowledge from a learning styles model to develop an approach for predicting learning styles from an individual's tutoring dialogue and a strategy for adapting the tutoring style to match. The Oscar Conversational Intelligent Tutoring System (CITS) was designed to model a human tutor by directing a tutoring conversation and automatically detecting and adapting to an individual's learning preferences. This chapter provides the context and motivation of this research, a summary of contributions and a thesis outline.

1 Background and Motivation

The design of the Oscar Conversational Intelligent Tutoring System (CITS) presented in this thesis adopted principles from three main areas of research, namely intelligent tutoring systems, learning styles and conversational agents.

The increasing complexity of computer systems and their interfaces has led to a move towards creating intuitive, human-centred interfaces (O'Shea et al., 2011). Conversational agents (CAs) are computer systems that allow us to communicate with computers intuitively using natural language (O'Shea et al., 2011). However, traditionally conversational agents can only mimic human conversation, but unlike humans do not pick up and react to social cues during a conversation (Becker et al., 2007). This lack of social intelligence can have a negative, demotivating effect on users, damaging their confidence in the intelligence of the computer system (D'Mello et al., 2009). If conversational agents could detect and react to conversational cues in social situations, they could provide an improved, more human-like tutoring solution.

In the context of this thesis, the social situation being modelled is a learning environment. Learning is inherently a social process (Moreno et al., 2001), and during face-to-face tutorials human tutors automatically pick up behavioural cues from students about their understanding and learning preferences, and adapt their teaching style to aid learning. Learning styles describe the way that groups of people prefer to receive and process information, for example by viewing diagrams rather than textual descriptions, or by active experimentation rather than observation (Honey and Mumford, 1992). Matching teaching styles to individual differences such as learning styles has been shown to improve learning (Felder and Silverman, 1988), but there are many models of learning styles and the choice of model is fundamental (Coffield et al., 2004a).

Intelligent tutoring systems (ITS) extend the traditional computerised learning systems 'one-size-fits-all' approach by capturing and modelling individual traits used to personalise the instruction (Brusilovsky and Peylo, 2003). This involves presenting learning material in a style and order to suit the learner (e.g. by presenting learning material matched to preferred learning styles), and also proactively helping learners, e.g. by giving intelligent feedback on incomplete or erroneous solutions and guidance to assist learners in constructing solutions to problems. Most ITS are student-led with menus or hyperlinks to topics, sometimes designed specifically to capture individual traits (Cha et al., 2006). A few ITS mimic a human tutorial by including CA interfaces which enable learners to construct their own knowledge through discussion (Graesser et al., 1999; Rahati and Kabanza, 2010; Dzikovska et al., 2010). Conversational ITS (CITS) have been extended to detect and react to learners' emotions (D'Mello et al., 2010b), however no attempt has been made to detect and react to learning style during a tutoring conversation.

The motivation for this research came from the need for an Intelligent Tutoring System that could mimic a human tutor by directing a tutoring conversation, dynamically detecting and adapting to student learning styles during the conversation. Two main research questions were identified:

1. Is it possible to predict a student's learning style from a two-way tutoring discourse with a conversational agent tutor?
2. Does adapting to a student's learning style during a two-way tutoring discourse with a conversational agent tutor improve learning?

This thesis describes how the novel Oscar CITS addresses these challenges, including how:

- Oscar uses an example learning styles model as the basis for capturing learner behaviour, modelling learning style and designing an adaptive tutorial conversation;

- Oscar's innovative predictor module mimics a human tutor by using detected learner characteristics to dynamically predict learning style during a tutoring conversation;
- the flexible design of Oscar's adaptation algorithm incorporates both student preferences and the adaptive capability of individual tutorial questions;
- Oscar mimics a human tutor by adopting a tutor-led, conversational approach incorporating intelligent adaptation, problem solving support and solution analysis;
- the generic methodologies and architectures for developing Oscar CITS allow the free choice of learning styles model and subject domain;
- empirical studies involving real students have validated both the prediction and adaptation of learning styles in a real-world teaching/learning environment.

2 Research Goal and Objectives

The research goal of this thesis is to develop a conversational intelligent tutoring system that dynamically predicts an individual's learning styles from their tutoring conversation and adapts its tutoring style to suit the learner's preferences. There are currently no CITS that can mimic a human tutor by dynamically personalising a tutoring conversation based on learning styles.

The objectives of this research are:

- To review learning styles models and the methods used to model and adapt to learning styles in ITS.
- To develop a methodology for creating a Predictive CITS (PCITS) that can dynamically model student learning styles during a two-way tutoring discourse.
- To design an architecture for a PCITS that dynamically predicts student learning styles.
- To design an adaptation algorithm for an Adaptive CITS (ACITS) that can dynamically adapt tutoring to student learning styles.
- To develop a methodology for creating an ACITS that can dynamically adapt tutoring to student learning styles during a two-way tutoring dialogue.
- To design an architecture for an ACITS that dynamically adapts to student learning styles.

- To validate both methodologies and architectures by implementing two prototypes and evaluating their success by conducting studies.
- To improve the reuse of the architectures and methodologies by removing any dependence on a particular learning styles model or subject domain.

3 Contributions

The most significant contributions of this work are:

- Proof of the concept that it is possible to predict student learning styles from a two-way natural language tutoring dialogue with a CA.
- The generic methodology for creating an Oscar conversational intelligent tutoring system that can dynamically predict learning styles from a natural language tutoring dialogue.
- Generic tools to aid development of a predictive CITS, including:
    - Logic rules that match behaviour captured during a natural language tutoring dialogue to learning styles.
    - Question styles and templates that can aid in the development of conversational tutoring scenarios to predict learning styles.
    - 3-level model of a tutorial conversation.
- The architecture for a conversational intelligent tutoring system that can dynamically predict learning styles from a natural language tutoring dialogue.
- The generic methodology for creating a conversational intelligent tutoring system that can adapt tutoring to suit individuals' learning styles during a natural language tutoring dialogue.
- The general dynamic adaptation algorithm that combines both the strength of learning style and the strength of adaptation available for individual tutoring questions to produce the best fitting adaptation per question.
- The architecture for a conversational intelligent tutoring system that adapts its tutoring to suit individuals' learning styles during a natural language tutoring dialogue.
- Two prototype CITS and results from empirical studies that validate the generic architectures and methodologies for a predictive and adaptive CITS that personalises tutoring to individuals' learning styles.

The generic tools, methodologies and architectures listed are independent of the learning styles model and subject domain. These contributions are expected to be of value to researchers and practitioners in the fields of learning styles and conversational intelligent tutoring systems (CITS). Researchers can use these contributions as a starting point for future projects and practitioners can follow the methodologies and architectures to create CITS that can predict and adapt to learning styles models selected for their curriculum.

4 Thesis Outline

The thesis is organised into twelve chapters. The nature of the research and design of the Oscar Conversational Intelligent Tutoring System (CITS) led to a considerable amount of experimental work, resulting in a large amount of documentation. To aid the reader, at the end of each chapter there is a summary list of chapter highlights.

As three substantial areas of research are brought together for this project, the background review of literature and existing work is described over three separate chapters. Chapter 2 introduces theories of learning styles, outlining current debates on the nature of learning styles and describing several common learning styles models. The use of models in computerised learning systems, criticisms and challenges in learning styles research are discussed. Chapter 3 describes conversational agents and different approaches to implementing text-based CAs, followed by a review of two successful pattern-matching text-based CAs and the challenges in CA research. In Chapter 4, Intelligent Tutoring Systems and methods of introducing 'intelligence' are described. Particular interest is paid to intelligent personalisation, with a detailed review of adaptation to learning styles including methods of modelling student learning styles and ways of introducing adaptation. Next there is an appraisal of the use of CAs to deliver tutoring and a review of the most sophisticated conversational ITS (CITS), followed by a discussion of the research challenges in the field of ITS.

Following the background review, the aspects of existing methods for incorporating learning styles into ITS which applied to a conversational ITS were analysed. Chapter 5 presents the investigations into the first research question, namely how to identify learning styles from a natural language tutoring dialogue with a CA. A generic methodology and architecture for creating a CITS to predict learning styles (Oscar Predictive CITS) is described in Chapter 6, and its implementation is presented in Chapter 7. Chapter 8 describes in detail the empirical validation of the generic methodology and architecture, presenting the design and results of three studies investigating the success of Oscar Predictive CITS in terms of predicting learning styles, learning gain and student feedback.

Next, the question of how to introduce adaptation to student learning styles into a conversational ITS was considered. Chapter 9 presents a generic methodology and architecture for creating a CITS that adapts to individual learning styles (Oscar Adaptive CITS). The implementation of Oscar Adaptive CITS and the methodology and results of the empirical evaluation are described in Chapter 10. The second research question is tested here: does adapting to student learning style during a natural language tutoring discourse with a CA increase learning gain.

In Chapter 11, the generic nature of the Oscar CITS is demonstrated by summarising the steps required to create a predictive CITS and adaptive CITS for a different learning styles model and subject domain.

Chapter 12 concludes the thesis by highlighting its contributions and describing its limitations and future direction.

Chapter 2 Learning Styles

1 Introduction

There has been much research over the past 50 years devoted to individual differences and learning. As the process of teaching has moved from traditional teacher-led instruction towards a more co-operative inquiry-based approach, the question of how best to support and encourage learners must be considered. Learning styles were selected for this research as they represent a significant aspect of the complex process of learning that could be applied to add human-like social intelligence to computerised learning systems.

This chapter introduces some of the proposed theories of learning styles and illustrates the breadth, complexity and challenges of research in this area. Also considered is the use of learning styles theories in practice, and how they can be applied to add a social element and enhance learning in computerised learning systems.

2 Learning Styles

In simple terms, learning styles describe the way in which groups of people prefer to learn. However, there are several contradictory theories about learning, and therefore no single agreed definition. Some learning theories are based on psychological theories such as personality traits and intellect whereas others focus on brain functioning or the learning environment. This broad range of research has produced many conflicting models of learning styles. The Coffield review of ‘Learning styles and pedagogy in post-16 learning’ (Coffield et al., 2004a) aimed to appraise the overwhelming amount of theoretical and empirical research on learning styles in an educational context. Coffield et al. critically reviewed 13 popular learning styles models, having identified the existence of 71 different learning styles models, and 3800 related articles.

There have been various attempts to classify different research approaches, with Curry (1983) developing three categories relating to a student’s instructional preferences, information processing style and cognitive style. Coffield et al. (2004a) organised learning style theories into five ‘families’ depending on whether learning styles are thought to be fixed (e.g. inherited or personality traits) or open to change (e.g. affected by the environment or context):

- *Constitutionally-based learning styles and preferences* – learning styles are fixed genetically (e.g. Dunn and Dunn, 1974).
- *Cognitive structure* – learning styles are fixed habits linked to personality (e.g. Gardner, 1983).
- *Stable personality type* – learning styles are one part of relatively stable personality type (e.g. Myers, 1962).
- '*Flexibly stable' learning preferences* – learning styles are stable but different by situation (e.g. Honey and Mumford, 1992).
- *Learning approaches and strategies* – personal, environmental and curriculum factors influence learning strategies (e.g. Entwistle, 1998).

The next section will summarise six commonly used learning styles models across the five families, selected based on Coffield's review (2004a) and their application in technology enhanced learning.

3 Common Learning Style Models

3.1 Dunn & Dunn Learning Style Model

Like several of the popular learning styles models, the Dunn and Dunn model (Dunn and Dunn, 1974; Dunn and Griggs, 2003) has changed from its initial version in 1974 following additional research. Coffield et al. (2004a) placed the model in the family of theorists who believe that learning styles are based on inherited traits, and although Dunn and Dunn acknowledge external factors like the environment, they believe that learning styles are fixed. The Dunn and Dunn model is popular in the USA, being used in a large number of primary schools, as it distinguishes between children and adults. The model was adopted for iWeaver (Wolf, 2002), an adaptive computerised learning environment that teaches Java programming. iWeaver matches learning material to learner preferences for two aspects of the model: perceptual (part of physiological) and psychological.

The model describes learning styles over five aspects called *stimuli*, each with several factors, as follows:

- *Environmental* includes preferences for sound, light, temperature and seating/furniture design.

- *Emotional* incorporates learner levels of motivation, persistence, responsibility and need for structure.
- *Sociological* describes preferences for learning alone, in pairs, with peers, as a team, with an authority or in varied approaches (and for children, motivation from teachers and parents).
- *Physiological* describes perception inclinations (visual, auditory, kinaesthetic or tactile), time of day energy levels, the need for food and drink and for mobility.
- *Psychological* (which was added in later versions of the model) characterises preferred information processing as global or analytic and impulsive or reflective.

The Dunn and Dunn learning styles model is assessed using a questionnaire that results in a high or low preference for each factor in the model. There are three different age levels of the Learning Styles Inventory for children (Dunn et al., 1996) with 104 questions answered using a 3-choice or 5-choice Likert scale. The Building Excellence Inventory for adults (Rundle and Dunn, 2000) has 118 questions answered on a 5-choice Likert scale.

The Dunn and Dunn model is easily understandable and incorporates motivation, social interaction and physiological and environmental factors. The model may also be applied widely to children and adults. However, the simplicity of the model's connections between brain function and psychological/physiological preferences has been questioned (Coffield et al., 2004a) and the model describes instructional preferences rather than learning.

3.2 Gardner's Multiple Intelligences Theory

In his theory of Multiple Intelligences, Gardner (1983, 1993) proposed that there is more to intelligence than the widely accepted traditional definition. Gardner's theory of multiple intelligences expands the traditional notion of intelligence (based on IQ testing) to describe eight different aspects of intelligence, as follows:

- *Visual/Spatial* – known as 'picture smart', spatial intelligence describes the ability to visualise spaces internally in the mind, e.g. for navigation or playing chess.
- *Linguistic/Verbal* – known as 'word smart', linguistic intelligence describes the ability to use words to express ideas and understand other people.

- *Logical/Mathematical* – known as 'number smart', logical/mathematical intelligence describes the ability to reason and understand causal systems or manipulate numbers.
- *Bodily/Kinesthetic* – known as 'body smart', bodily/kinaesthetic intelligence describes the capacity to use ones body skilfully.
- *Musical/Rhythmic* – known as 'music smart', musical intelligence is the capacity to think in music, hearing, recognising and repeating patterns.
- *Interpersonal* – known as 'people smart', interpersonal intelligence is the ability to understand other people.
- *Intrapersonal* – known as 'self smart', intrapersonal intelligence refers to an introspective and reflective understanding of oneself, ones abilities, desires, reactions and weaknesses.
- *Naturalistic* – known as 'nature smart', naturalistic intelligence describes the ability to nurture and relate information to the environment.

Although not specifically related to learning, Gardner proposes that teaching should broaden its traditional linguistic and logical focus to incorporate different activities that better serve students with strengths in different intelligences. Gardner has not defined a test to assess an individual's Multiple Intelligences, as he believes it to be "more of an artistic judgement than of a scientific assessment" (Gardner, 1983). The EDUCE adaptive computerised educational system (Kelly and Tangney, 2006) successfully uses Gardner's Multiple Intelligences theory as a basis for dynamically modelling learning characteristics and delivering adaptive learning material. However, the model has been criticised as it does not redefine intelligence, but rather describes different abilities and skills.

3.3 Myers-Briggs Personality Types

The Myers-Briggs Type Indicator (MBTI) (Myers, 1962) categorises an individual's personality type and their approach to relationships. Although MBTI is not a learning styles model, Coffield et al. (2004a) reviewed it as part of the family of theories proposing that learning styles is one observable aspect of personality. The scope of MBTI includes learning and it is widely used in consultancy and training as a career development and managerial tool (Furnham and Medhurst, 1995). The MBTI model has also been used in computerised learning, for example El Bachari et al. (2010) who designed an adaptive e-learning system based on learner personality.

MBTI classifies a person's type according to four dichotomies (The Myers and Briggs Foundation):

- *Extroversion/Introversion* describes the preferred focus of an individual as on the outer world of people and things (extravert) or inner world of thoughts and ideas (introvert).
- *Sensing/Intuition* describes the way individuals perceive information – from their five senses (sensing) or from patterns and possibilities in the information (intuition).
- *Thinking/Feeling* categorises the way individuals evaluate information – based on logical judgements such as true or false (thinking) or on subjective evaluations such as better or worse (feeling).
- *Judging/Perceiving* describes how individuals live their outer life – preferring a structured and decided (judging) or flexible and adaptive (perceiving) lifestyle.

The MBTI is evaluated using Form M (Myers and McCaulley, 1998), a 93-question forced-choice questionnaire resulting in one of sixteen MBTI types (based on combinations of the dichotomies, e.g. ISTJ). The dichotomies are not independent, as each MBTI type represents a set of complex relationships between dichotomies known as type dynamics, and described by positive and negative traits.

3.4 Honey & Mumford's Learning Styles Model

The Honey and Mumford (1992, 2006) learning styles model was based on the influential Learning Style Inventory (Kolb, 1976), and is widely used in the UK for personal and organisational development. The model has also been used in a number of technology-enhanced learning systems, for example INSPIRE (Papanikolaou et al., 2003) which adapts its teaching of computer architecture to suit learner preferences. Honey and Mumford believe that under-utilised learning styles can be strengthened by following the actions given in the model. Honey and Mumford define four learning styles, as follows:

- *Activists* like to take direct action, try things out, participate and be the centre of attention. They are enthusiastic and welcome new challenges. Activists are more interested in the present than the past or the broader context.
- *Reflectors* are thoughtful, preferring to think things through before taking action. They are good listeners, are happy to repeat learning and like to keep a low profile.

- *Theorists* are logical, analytical and objective and prefer a sequential approach. They pay attention to detail and like to see how things fit together overall.
- *Pragmatists* are practical, enjoy experimenting with new ideas and like to solve problems by seeing how things work in practice.

Learning styles are assessed using the Learning Styles Questionnaire (LSQ). There are two versions of the LSQ, the 40-item and 80-item versions, which use a number of yes/no questions to assess the strengths of the four learning styles. The four resulting scores are then compared to a list of group norms that categorise the learning style preference as very strong, strong, moderate, low or very low.

The Honey and Mumford model describes the learning cycle in four stages: experiencing, reviewing, concluding and planning. The descriptions of preferred learning styles are designed to help learners maximise their learning over each stage of the learning cycle, and advice is given on how to strengthen under-utilised styles. The wide adoption of the model in business reflects its ease of use and practical advice, however no specific advice is given for educators so it may not be appropriate in higher education.

3.5 Felder-Silverman Learning Styles Model

The Felder-Silverman (FS) learning styles model (Felder and Silverman, 1988) was developed to describe the learning styles in engineering education and suggest different teaching styles to address learners' needs. The FS model defined five dimensions of preferred learning style: perception (sensory-intuitive), input (visual-auditory), organisation (inductive-deductive), processing (active-reflective) and understanding (sequential-global). Following further research and additional sampling, Richard Felder removed the organisation dimension from the model and altered the input dimension to visual-verbal (thus *verbal* includes both written and spoken words) (Felder and Henriques, 1995).

The four dimensions of learning style defined in the FS model each relate to a stage in the process of receiving and processing information as follows:

- *Perception* – learners are *sensory* or *intuitive* depending on the type of information they prefer to perceive, e.g. external or internal. Sensing learners prefer facts and experimentation, are patient with detail, comfortable with symbols (e.g. words) and careful but slow. Intuitive learners prefer principles and theories, are bored by detail, uncomfortable with symbols and quick but careless.

- *Input* – learners are *visual* or *verbal* according to the way they prefer to receive external information, e.g. diagrams or explanations. Visual learners remember what they see, like pictures and diagrams and prefer visual demonstration. Verbal learners remember what they hear, like discussion and prefer verbal explanation.
- *Processing* – learners are *active* or *reflective* according to the way information is converted into knowledge, e.g. discussion or introspective consideration. Active learners like to do something with information (discuss or test), they are experimentalists and process information by testing an idea. Reflective learners like to examine and manipulate information internally, are theoreticians and process information by postulating explanations and drawing analogies.
- *Understanding* – learners are *sequential* or *global* depending on their progression towards understanding, e.g. continual steps or large jumps. Sequential learners like to follow a linear reasoning process, can work with partially understood material and prefer information presented in a steady progression of complexity. Global learners make intuitive leaps, have difficulty working with material they have not understood and prefer to jump directly to complex material.

There are 16 (2^4) learning styles overall (an example being sensory/visual/ active/sequential). Each learning style dimension may be thought of as an axis with the opposite learning styles at each end (as illustrated in Figure 2.1), and learners are placed on each axis according to the strength of their preferred learning style. By defining independent dimensions the FS model describes the detail of learner tendencies, including the strength of preference as well as the nature of learning styles.

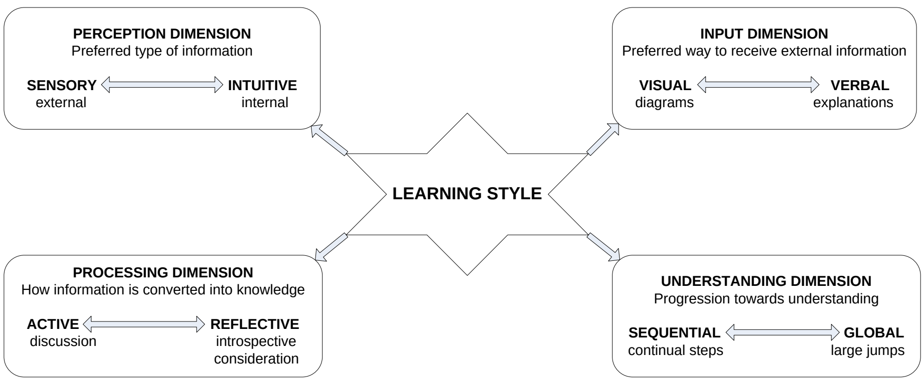


Figure 2.1. FS Model Dimensions

The Index of Learning Styles (ILS) (Felder and Soloman, 2008) is an instrument to assess Felder-Silverman learning styles. The ILS is a 44-question forced-choice self-assessment questionnaire with 11 questions per learning style (see Appendix 1). For each dimension, answers are compared and result in a learning style and a score from 1 to 11 (in steps of 2). Scores of 1 or 3 place the learner at the centre of the axis and indicate a low preference for that learning style, which Felder and Silverman call *neutral* preferences.

The FS model is frequently adopted by technology-enhanced learning systems (Dag and Gecer, 2009), e.g. CS383 (Carver et al., 1996) teaches computer systems, CooTutor (Wang et al., 2006) teaches spatial geometric transformation and Li et al. (2010) report an adaptive course on web technologies. Reasons for this popularity include:

- it is more feasible to implement a small number of dimensions (like the FS model);
- FS dimensions are distinct and independent;
- the FS model contains detail of typical behaviour for each learning style;
- there are descriptions of teaching styles that match preferred learning styles;
- the model was specifically designed for engineering students, and many computerised learning systems are applied to teaching computer science courses.

The FS model was developed with the purpose of improving teaching styles in engineering education. Its strengths lie in the detailed descriptions of learner behaviour tendencies and suggestions for teaching styles, and the fact that it describes both learning style preference and the strength of that preference. However it is recognised by the model that, as with most generalisations, the actual behaviour of students will not always conform to the tendencies described in the model.

3.6 Entwistle's Approaches and Study Skills Inventory for Students (ASSIST)

Entwistle's research (Entwistle, 1981, 1998) focuses on students' strategies for learning, proposing that learning styles are not fixed by inherited characteristics, but are affected by the learning environment. The model describes students' approach to learning and intellectual development and applies to students within higher education. Entwistle's model differentiates between a learning *style* (a student's preferred way of approaching learning) and a learning *strategy* (a student's approach to a specific task based on the perceived requirements).

Three main approaches to learning are described by the model (Entwistle et al., 2001) as follows:

- *Deep approach* – this describes students who intend to understand ideas for themselves, taking an active interest and personal engagement in learning.
- *Surface approach* – these students intend to cope with the course requirements, memorising facts and studying without reflecting on purpose or meaning.
- *Strategic approach* – this approach describes students who intend to achieve the highest possible grades by gearing the work to particular lecturers and being alert to assessment requirements.

The ASSIST inventory (Entwistle, 1997) aims to measure undergraduate students' approaches to learning and their perceptions about the impact of course organisation and teaching. The inventory has 66 questions answered using a 5-choice Likert scale over three sections: what is learning, approaches to studying and preferences for different types of course and teaching.

ASSIST is intended to be used as a diagnostic tool for lecturers, students and course teams aiming to promote an environment that encourages the deep approach to learning. The strength of the model is its aim to describe strategies and approaches to learning and the attitude toward development of intellectual skills in higher education. However, the model is complex and not easy for non-specialists to apply, and has not been adopted by any computer-enhanced learning systems.

4 Challenges for Theories of Learning Styles

As has been described, there has been a large amount of research into learning styles leading to much controversy about the nature of learning. The lack of firm conclusions leaves a number of challenges and open questions, as follows:

- There is no holistic model of learning styles. A large number of learning style models exist, with inconsistent findings on their stability, similarity, differences and appropriateness to education. The major review undertaken by Coffield et al. (2004a) attempted to clarify the position.
- There is much disagreement on the nature of learning styles, in particular over whether they are fixed or change over time, environment, subject or history of learning experience. Coffield grouped learning styles models based on the

biggest area of difference – the argument over whether learning styles are god-given traits or skills and strategies.

- Which model is most appropriate for the situation? There is a lot of overlap, and some models are too complex to understand and implement by teachers, e.g. Entwistle.
- Does matching to learning styles in education work? Do mismatched learning styles stimulate motivation or does this depend on personality? If learning styles change over time, is it more helpful not to match to learning styles? Some approaches believe that matching to learning styles improves learning (Hsieh et al., 2011), whereas others think that learners benefit from mismatching (Felder and Brent, 2005; Smith et al., 2002).
- How feasible is it to match different teaching styles? In practice, do teachers have time to plan different formats of learning material, or adapt teaching based on learner feedback? Matching to learning styles in computerised learning systems requires a lot more development time and effort, extra resources, etc. so do the gains justify this?
- How accurately can learning styles be measured? How is this done in practice, and how often (if they change)? Are questionnaires completed properly by learners, who find them time-consuming? Coffield et al. (2004a) checked the validity of questionnaires, but if they are not completed accurately the results will be affected.

Despite the variety of opinions in this field, it is widely accepted that individual differences exist (Felder and Brent, 2005) and that learning preferences reflect individual traits (Entwistle, 1988; Honey and Mumford, 1992). Coffield concluded that the choice of model matters fundamentally in post-16 learning.

5 Learning Styles in Practice

In practice, learning styles are used in different ways, often reflected by the choice of model. For example, Honey and Mumford’s model (1992) is used in business for personal and organisational development (e.g. in management training for staff coaching and development (Avon and Somerset Constabulary)), whereas Entwistle’s (1997) ASSIST is used by educators wishing to promote an environment for deeper learning. In education, the formal assessment questionnaires are most often used to improve student self-awareness, but it is not common for lecturers to use the results of a formal tool when planning to teach a course. Instead, a lecturer will typically use their knowledge and experience of different groups of learners to incorporate different types of material and activities (Felder and Brent, 2005). During tutorials, lecturers will intuitively pick up informal behavioural cues from students that indicate their level of understanding and their preferred learning style, and use these observations to adapt their teaching style accordingly.

For example, in the FS model (Felder and Silverman, 1988) typical learner behaviours and associated teaching styles are described for each learning style. This information is useful to lecturers when informally grouping types of learners.

Although opinions on the usefulness of learning styles in education vary, their application to experimental research in computer-based education systems has shown that matching learning material to students' preferred learning styles can enhance their learning (Walters et al., 2000; Paredes and Rodriguez, 2004; Rasmussen, 1998; Riding and Grimley, 1999; Graff, 2003; Allinson and Hayes, 1996; Felder and Brent, 2005).

6 Conclusion

This chapter has introduced the broad area of research on learning styles and contrasted some of the many opposing theories. The Coffield (2004a) review organised learning styles theories into five families on a continuum, according to their biggest difference: the belief of whether learning styles are fixed by genetics or change according to the environment. Six common learning styles models across this continuum were summarised. The controversies in the debate on learning styles has resulted in a number of open questions and a lack of firm conclusions. Coffield (2004a) concluded that the choice of learning styles model in post-16 learning is fundamental.

In practice, learning styles are used mostly in education by learners for self-understanding. Workload and time constraints mean that formal assessment is rarely used by teachers during course design. Instead, teachers are more likely to use their experience and knowledge of learning preferences to informally group learners and adapt their teaching. Computer-based education systems are increasingly used to augment the student learning experience as well as for distance learning. By including knowledge of learning styles and delivering suitable learning material, some social intelligence may be added to computerised learning systems to improve the learning experience. A number of studies (Felder and Brent, 2005; Cooper, 2002; Smith et al., 2002) have shown that matching learning material with learning styles in computerised learning systems has improved learning outcomes. A computerised learning system that could automatically determine and adapt to an individual's learning styles could offer enhanced learning experiences without the time cost to human tutors.

7 Chapter Highlights

- Learning styles describe the way groups of people prefer to learn.
- There are numerous theories of learning styles with a wide range of views on their stability.
- Coffield et al. (2004a) organised and critically reviewed learning styles research in the context of post-16 education.
- There are many challenges and open questions in learning styles research that remain under debate.
- The choice of learning styles model is fundamental, but overlap and inconsistent descriptions make it a difficult choice.
- In practice, teachers rarely use formal assessment of learning styles when designing courses.
- Computerised learning systems that personalise teaching according to learning styles can enhance learning.

Chapter 3 Conversational Agents

1 Introduction

Communicating with a computer using natural language has been a goal in artificial intelligence for many decades, stimulated by the ‘Turing Test’ (Turing, 1950). Turing’s test was proposed to move the research community on from debating whether machines can think, to developing machines that could imitate humans well enough to fool a judge. Early attempts to pass the Turing test involved using computer programs (conversational agents) called chatterbots that used tricks during a conversation to create an illusion of intelligence (Weizenbaum, 1966).

Conversational agents (CAs) enable people to communicate with computers using natural language. The term CA has been used broadly to describe textual, spoken or embodied conversational systems (O’Shea et al., 2011), but they all share the key challenge of understanding the user input and responding appropriately. In expert systems (such as tutoring, Graesser et al., 2008b) CA interfaces can engage users in discussion by replicating human communication (e.g. classroom learning), helping users to build motivation and confidence by drawing on their own experience.

This chapter will introduce different approaches to implementing CAs, and then describe pattern-matching text-based CAs. Two successful text-based CAs will be reviewed and finally the challenges in developing more human-like CAs discussed. Despite the difficulties in developing CAs, natural language interfaces are desirable as an intuitive and familiar method of communicating with computer systems. Therefore CAs are an ideal solution for communicating with intelligent tutoring systems as they allow students to construct knowledge by asking questions and discussing problems as with a human tutor.

2 Conversational Agents

Conversational agents (CAs) allow people to interact with computer systems intuitively using natural language dialogues (O’Shea et al., 2011). CA interfaces are intuitive to use and have much to offer businesses with an online presence, adding a ‘friendly face’ to websites by offering users additional support and advice (Lee et al., 2001), for example ANNA, a CA guide to IKEA’s website (Inter IKEA Systems BV, 2004)). In addition, CAs can act as cost-reducing and profit-enhancing tools, managing emails and SMS messages, supporting sales and aiding customer retention by detecting and reporting dissatisfaction from their conversations (VirtuOz, 2007). As well as adding natural language ability to computer interfaces (Owda et al., 2011), CAs offer the traditional benefits of computer systems: they present consistent advice, do not require rest and are available for use at all times of the day. CAs have been used effectively in many applications, such as web-based guidance (Latham et al., 2010), database interfaces (Owda et al., 2011) and computerised learning (D'Mello et al., 2010a).

There are three main types of CA:

- The first *text-based CAs* were designed with the sole aim of holding a conversation, and are known as *chatbots* (Carpenter, 2007). More complex text-based CAs, such as goal-oriented textual CAs, are designed to address specific problems in a well defined subject domain. Goal-oriented textual CAs follow a goal-oriented methodology such as ConvAgent's InfoChat (Convagent Ltd., 2005).
- *Spoken dialogue systems* are also usually goal driven but use a spoken rather than textual interface (Sadek, 1999).
- *Embodied CAs* additionally mimic human gesture and body language during a conversation (Cassell, 2000).

In the context of this thesis, the term CAs refers to text-based CAs. It is relatively straightforward to add voice capability to a textual CA, which widens access and gives the appearance of a more human-like interface. By adding a voice capability to a textual CA intelligent tutoring system, D'Mello et al. (2010a) found that students progressed more quickly through the curriculum.

CAs conduct a conversation by accepting natural language user input and producing an appropriate response. Responses usually consist of predefined 'canned text' that can be changed to reflect the conversation context using variables. For example, variables can be used to include a name, e.g. 'How are you feeling today, Bob'. However, there are a number of different approaches to understanding user input in CAs:

- The natural language processing (NLP) approach (Khoury et al., 2008) seeks to understand the user input by studying the constructs and meaning of natural language and by applying rules to process important parts of sentences. Whilst sophisticated, NLP requires a lot of computational power that impacts on speed and scalability for real-time use over the Internet. Another problem with NLP is that user utterances are expected to be grammatically correct, which is often not the case.

- Pattern matching systems (Wallace, 2009; Convagent Ltd., 2005) use an algorithm to match key words and phrases within an utterance to pattern-based stimulus-response pairs rather than attempting to understand the input. Although limited to existing stimulus patterns, the pattern matching approach does not require grammatically correct or complete input. However, developing a set of stimulus-response pairs (known as a script) is a complex and time-consuming task.
- The AI method (O'Shea et al., 2010) compares the semantic similarity of phrases (Li et al., 2004) to decide on the meaning of the input. Research into semantic similarity measures is in its infancy and despite aiming to reduce the development time and effort of scripting CAs, the benefits are not yet fully realised.

CAs are ideally suited to simple question-answering systems as they are intuitive to use and allow users direct, non-linear access to information of interest. However, for systems requiring lengthy dialogue (such as tutoring), the time and expertise required to develop sophisticated CA scripts that adequately mimic human conversation is a challenge rarely tested.

3 Pattern-matching Text-based CAs

Most text-based CAs adopt the pattern matching approach as it is currently the one that works best for extended dialogues (O'Shea et al., 2011). The pattern matching approach requires the development of conversation scripts, a similar idea to call centre scripts, which match key input words and phrases to suitable responses. Scripts usually contain numerous patterns, leading to many hundreds of stimulus-response pairs in the CA's knowledge base, which demonstrates the complexity and time required to script a CA. Scripts are initially developed by anticipating user utterances and writing stimulus-response pairs to match them. CA scripts require considerable maintenance, needing continuous improvement by reviewing incorrect CA responses from conversation histories and modifying or adding stimulus-response pairs to address the problem. This requires considerable language expertise, and is labour intensive and time consuming.

A CA script is made up of a set of pattern-based stimulus-response pairs (hereafter called rules) containing a set of stimulus patterns, the rule's current status and a response pattern. Wildcards are used within patterns to match any number of words, broadening the rules to match utterances containing specific key phrases. As pattern matching CAs match key words within an utterance, they do not require grammatically correct or complete input. One of the biggest challenges is understanding the context of non-specific user utterances such as "What does it mean" and "Yes, do you?". Different topic groups and conversation histories are used to help find appropriate matches, for example, the meaning of a user utterance "Yes, please show me" can only be understood in relation to the current context and previous utterances of the conversation.

An algorithm decides the best fitting rule to fire, thus producing a CA response. CA scripts may be grouped into topics and may be linked together in a tree or graph structure (Sammut, 2001), sometimes over various levels (e.g. a script filter to capture abusive language). The organisation of the scripts and the efficiency of the matching algorithm have a direct impact on the real-time use of CAs as interfaces.

Pattern-matching textual CAs can be applied to general (chatbots) or goal-based conversations (e.g. gathering information to provide guidance (Crockett et al., 2009)), depending on the development methodology adopted. Two successful pattern-matching text-based CAs will now be reviewed.

3.1 Artificial Linguistic Internet Computer Entity (ALICE)

ALICE (Wallace, 2009) is a freely available text-based chatbot that was ranked the 'most human computer', winning the Loebner Prize (an annual Turing test competition (Loebner, 2011)) three times. ALICE implements pattern matching using an XML-type scripting language called Artificial Intelligence Markup Language (AIML). By making ALICE and AIML available free, the distributed development of around 41,000 rules (called *categories*) that can be matched to user utterances has been possible. The categories are held in a tree structure by an object called the GraphMaster that implements a pattern storing and matching algorithm that works in real time. The knowledge base (or ALICE's 'brain') can be downloaded along with the specifications for AIML and ALICE software from ALICE AI Foundation (2007).

Table 3.1 shows an example of an AIML category, which consists of a *pattern* and a *template* (response). The pattern consists of a key phrase that contains a wildcard (*) to match any number of words. The words matched to the wildcard can be retrieved for use in the response using <star/>. In the example template, a variable (called predicates in AIML) value is retrieved to tailor the response to the dialogue partner. The predicate *name* will have been set previously in the dialogue using the markup <set name=”name”>Dylan</set>. Variables (called *predicates*) allow information about the conversation to be stored, for example predicates are commonly used to bind pronouns (such as *he*) to subjects (such as *Einstein*).

Table 3.1. Example AIML Category

```
<category>
   <pattern>CAN I PLAY * TURING TEST</pattern>
   <template>
      We are already playing the Turing Game,
      <get name="name"/>
      .  Now it's your turn.
   </template>
</category>
```

Source: ALICE AI Foundation, 2007

The AIML recursion operator acts like a ‘goto’ command, recursively matching categories to divide up utterances or spot keywords. The conversation context is managed by storing the last utterance, and also by grouping categories into *topics* that are treated like ordinary words added to an utterance. AIML scripts may be written using a Knowledge Wizard tool that generates linguistic variations for simply phrased questions, such as “Who is Turing?”. However, anticipating user utterances and writing patterns to match these is a skilful and time-consuming job.

The use of ALICE and AIML is widespread as the chatbot is provided freely as open source. This allows the distributed development of the knowledge base with new patterns being added by many users. The set of patterns also evolves to improve incorrect responses noted from conversation histories by making patterns more specific. However, this strategy creates the problem of managing the knowledge base, particularly with regard to duplicate categories that must be removed by the ALICE AI Foundation.

AIML is a fairly simple scripting language, and patterns may only contain alphanumeric characters which may be insufficient for some applications. The restriction of only one pattern per category means that scripts are long and difficult to maintain, as there are many synonyms, and AIML's use of recursion to address synonyms does not lead to shorter scripts. The dialogue context is managed very simply by appending topic names to categories, and there is no mechanism for grouping or navigating categories to capture utterances on a particular subject.

ALICE and AIML rely on a very large knowledge base to manage general chat, however for many practical applications a goal-oriented CA is better suited.

3.2 InfoChat

InfoChat is a commercially available goal-based CA marketed by ConvAgent Ltd (2005). InfoChat has been successfully used in online advice and guidance systems, such as Adam, the Student Debt Advisor (Crockett et al., 2009) and the Bullying and Harassment Advisor (Latham et al., 2010). InfoChat implements pattern matching using a sophisticated scripting language, PatternScript (Michie and Sammut, 2001). Scripts are made up of rules that consist of stimulus patterns and responses, where each pattern matched to an input generates a response. PatternScript includes more features than AIML, such as shorthand features like *macros* (discussed below) and the ability to have numerous patterns within a single rule. This leads to shorter and better organised scripts which makes development and maintenance more efficient. PatternScript also allows scripts of rules to be organised into contexts that manage particular parts of a conversation.

Table 3.2. Example PatternScript Rule

```
<What-is-Bullying>
a:0.01
p:50  *<explain-0> * bullying*
p:50  *bullying *<explain-0>*
p:50  *<remind-0> * bullying*
p:50  *bullying *<remind-0>*
p:50  *<explain-0>* a bully*
p:50  *a bully*<explain-0>*
 r:   Bullying is persistent, threatening, abusive, malicious,
      intimidating or insulting behaviour, directed against an
      individual or series of individuals, or a group of people.
      *<set BullyDef true>
```

Source: Latham, Crockett & Bandar 2010

Table 3.2 shows an example PatternScript rule that consist of an *activation* level, a number of *patterns* with associated strengths and a *response*. In the rule, *<What-is-Bullying>* is the rule name; *a* is the activation level used for conflict resolution (Michie, 2001); *p* is the pattern strength followed by the pattern that is matched against the user utterance and *r* is the CA response. As each rule can contain many patterns, each matched individually, PatternScript scripts are shorter and easier to maintain than AIML scripts.

In the example (Table 3.2), patterns contain the wildcard (*) that matches any number of words, which can later be retrieved for use in the response. Patterns also contain macros (in the example *<explain-0>* and *<remind-0>*) that contain a number of standard patterns for synonyms which are each matched separately. The macro facility reduces the number of patterns needed in rules and simplifies maintenance, as synonyms need only be added once but are used by all scripts. Variables are set (and their values may be recalled) as part of the response; in the example the variable *BullyDef* is set to ‘true’ using the *<set> command.

PatternScript’s naming of rules allows the incorporation of features for controlling a conversation, such as the promotion, demotion or killing off of rules within the scripts. The setting of variables can also be used to restrict rules. These features along with the setting of activation levels for rules and patterns (strengths) enable scripts to be fine tuned and the priority of certain key words and phrases to be represented.

PatternScript allows the scripts to be developed modularly by grouping rules into sets called *contexts* (Sammut, 2001). Structuring scripts into contexts adds contextual information to user utterances to aid their understanding. There is always a current context representing the current state of the conversation. Rule responses may push control to a new context, and this recursive system is used to move the conversation towards its goal. Levels of contexts may also be used for *filtering* (e.g. responding to abusive language) or *backup* scripts (when no suitable pattern matches the input).

PatternScript has a number of sophisticated features for matching patterns and for controlling the dialog by promoting or demoting rules. PatternScript also allows scripts to be clearly organised into contexts, promoting modular development. The ability to associate numerous patterns with a rule and to define macros for common expressions leads to smaller and more maintainable scripts. However, as well as adding control, the use of PatternScript’s complex features adds the necessity for considerable expertise in developing and maintaining CA scripts. With PatternScript’s features for managing conversations and organising scripts, InfoChat lends itself to goal-oriented implementations. InfoChat has been successfully employed as a goal-based CA where the conversation domain is tightly defined (O'Shea et al., 2011).

4 Challenges for Text-Based CAs

Despite many decades of research, the success of CAs as intuitive computer interfaces is limited. A number of challenges remain that influence development and maintenance costs, including:

- The development of CA scripts is a time consuming and labour intensive process, which has an impact on development costs.
- Development of CA scripts is anticipatory (what will users say) and backward-looking (correcting incorrect responses by making more specific patterns) leading to a lengthy development time.
- The biggest challenge of scripting CAs is the coverage of all possible user utterances. Although this is more of a problem for chatbots that require general knowledge, goal-oriented CAs must also employ mechanisms to manage unexpected utterances in a way that appears intelligent.
- Skills in developing dialogues are crucial, as CA responses must be carefully written to maintain the conversation and, in goal-oriented CAs, to steer the conversation towards its goal.
- Developers of CA scripts must be highly skilled in selecting patterns of key words to match the required user utterances and give a sensible response.
- The maintenance of CA scripts is a complex task as rules interact and compete with each other and changes to even one rule can destabilise a CA. It is therefore essential that scripts and rules are organised coherently, for example using contexts, to minimise maintenance effort.
- The use of CA interfaces over the Internet requires systems to cope with large numbers of conversations simultaneously in real time. Response time must be fast enough to maintain a consistent dialogue, and there are limitations in scalability with some sophisticated CA approaches, such as NLP (O'Shea et al., 2011).
- When applied to extended conversations rather than answering direct questions, e.g. about products, CAs lack the social intelligence of humans. To genuinely mimic human behaviour, CAs additionally need to be able to pick up and react to

user affect, such as mood, personality, boredom, confusion or frustration (Becker et al., 2007).

Despite these challenges, CAs are able to communicate adequately with users in clearly defined domains. The recent move toward incorporating human-like social behaviour into CAs aims to improve the user experience by making CAs seem more natural (and less robotic). Some research has been done in detecting and responding to human social behaviour in CAs. Kumar et al. (2010) incorporated social conversational skills based on Bales (1950) Socio-Emotional Interaction Categories into a CA tutor for a collaborative learning environment. Graesser et al. (2008a) investigated the relations between emotions and learning with their CA tutor, AutoTutor, by correlating dialogue characteristics with emotional states recorded by learners. Mairesse et al. (2007) detected personality type from conversation and text using linguistic cues. Ma et al. (2005) estimated emotions from text-based conversation using keyword spotting. Ovesdotter et al. (2005) used machine learning to predict emotion from text, using children's fairy tales.

However, the goal of making conversations with computers seem human enough to pass the Turing test is still a long way off.

5 Conclusion

This chapter has introduced conversational agents, which allow the communication with computer systems using natural language. Text-based conversational agents were described. Two successful text-based CAs were contrasted, both of which adopt the pattern-matching approach which can cope with grammatically incomplete and incorrect utterances. The ALICE chatbot relies on a large knowledge base of rules for general conversations, but for goal-based situations such as tutoring, InfoChat is more powerful and the features of the PatternScript language offer more sophisticated scripting of a CA. The many challenges in developing CAs that can work in real time for extended dialogues were described. The main challenges include the complex, labour-intensive and time-consuming development and maintenance of CA scripts, the ability to respond in real time and the coverage of user utterances. Finally, the challenge of incorporating human-like social intelligence into CAs was introduced. Humans instinctively pick up and respond to verbal signals (e.g. "show me" instead of "tell me") that indicate different preferences for receiving information during a conversation. Developing CAs that can mimic this social intelligence will make them less robotic and easier to interact with. The application of social behaviour to CAs depends on the situation. For example, in computerised learning systems CAs need to detect and react to cues from learners indicating their preferred learning styles to make the learning experience more effective in promoting a deep understanding. A CA tutor that could actively discuss problems and solutions using natural language and react to social signals during the conversation could widen access to and help support face-to-face learning.

6 Chapter Highlights

- Conversational agents (CAs) enable natural language communication with computer systems.
- Most text-based CAs use a pattern-matching approach as it can cope with grammatical and spelling errors and is fast enough to respond in real time.
- Developing CA scripts is a complex, labour-intensive and time-consuming task.
- Once text-based conversation is working adequately, voice tools can be plugged in to enhance communication.
- ALICE and InfoChat are successful text-based pattern-matching CAs.
- A recent challenge for CAs is to incorporate human-like social intelligence.
- Computerised learning systems can benefit from CA interfaces that allow learners to discuss problems and construct knowledge.

Chapter 4 Intelligent Tutoring Systems

1 Introduction

Computerised learning systems were traditionally information-delivery systems developed by converting tutor or distance-learning material into a computerised format (Brooks et al., 2006). The popularity of the Internet has enhanced the opportunities for e-learning, however most online systems are still teacher-centred and take little account of individual learner needs (Spallek, 2003). Within the field of computerised learning systems, adaptive educational systems attempt to meet the needs of different students by offering individualised learning (Brusilovsky and Peylo, 2003). Intelligent Tutoring Systems (ITS) are adaptive systems that use intelligent technologies to personalise learning according to individual student characteristics, such as knowledge of the subject, mood and emotion (D'Mello et al., 2009; Graesser et al., 2008a) and learning style (Yannibelli et al., 2006).

There has been a wealth of research into adaptive computerised educational systems like ITS as they offer benefits such as being available at a time and place to suit the learner whilst offering individualised instruction, guidance and instant feedback. ITS can also help to widen access to education, and are cost effective compared to human one-to-one tutoring. This chapter aims to give an overview of the current state of ITS research and the challenges still to be overcome. The chapter brings together the three key areas of learning styles, conversational agents and intelligent tutoring systems, demonstrating the need for a socially intelligent conversational agent tutor and justifying this research.

This chapter will outline the current methods of including intelligence into ITS and consider the use of natural language interfaces such as conversational agents in ITS. Existing research methods of applying learning styles to enhance ITS will be discussed, along with the different approaches to modelling and adapting to learning styles. A number of unresolved challenges for future ITS research are then described.

2 Intelligent Tutoring Systems

Intelligent Tutoring Systems (ITS) are adaptive educational systems that employ intelligent technologies to provide individualised instruction (Graesser et al., 2005c). ITS build a model of the goals, preferences and knowledge of the student, and use this to adapt the teaching to the individual and to provide intelligent assistance. Adaptive Hypermedia Systems (AHS) are a type of adaptive educational system that create pages containing hyperlinks or menu items adapted to individual students. Although sometimes defined as distinct from ITS (Brusilovsky and Peylo, 2003), in the context of this thesis AHS are defined as a specific type of ITS that use hyperlinks and are concerned with just one aspect of intelligent tutoring: personalising learning material (known as 'curriculum sequencing'). Therefore in this thesis, the term ITS is used, which encompasses AHS.

There are three main approaches to intelligent tutoring: intelligent solution analysis, problem solving support and curriculum sequencing (Brusilovsky and Peylo, 2003).

Intelligent solution analysis adds intelligence to ITS by giving students detailed feedback on incomplete or erroneous solutions, helping them learn from their mistakes. In SQLTutor (Mitrovic, 2003) a technique called constraint based modelling is used to model the syntax and semantics of SQL. Student solutions are compared to the constraint model and intelligent feedback is given on errors so that students can learn from their mistakes.

Problem solving support techniques offer learners intelligent assistance to reach a solution. This approach adopts the constructivist style of teaching, as used by human tutors, to prompt learners to construct their own knowledge and encourage a deeper understanding of a topic. In ActiveMath (Melis, 2001) intelligent support is offered for mathematical theorem proving and in CIRCSIM-tutor (Woo Woo et al., 2006), hints help students diagnose physiology problems.

Curriculum sequencing systems introduce adaptation by presenting students with learning material in a sequence and style best suited to their needs. Curriculum sequencing is the technique most widely used by ITS and AHS (Brusilovsky and Peylo, 2003). Personalisation was traditionally based on existing knowledge, aiming to improve the learning experience by focussing the tutoring on topics that are unknown or require improvement. In ELM-ART (Weber and Brusilovsky, 2001) student knowledge is modelled and presentation is adapted with the annotation of learning resource links to indicate recommended resources. Recently personalisation has been extended to include other individual characteristics that might affect learning, such as the learner's emotions (Ammar et al., 2010; Graesser et al., 2006), personality (Leontidis and Halatsis, 2009) or learning style (Popescu, 2010). D'Mello et al. (2010b) mimicked human tutors in encouraging engagement by adapting to learner's emotions such as boredom or frustration. A review of ITS adaptation based on learning styles will be described further in section 3.

In order to personalise learning using curriculum sequencing, ITS capture and represent student characteristics in an individual student model and then use that information as a basis for adaptation. The student model therefore plays a central role in an ITS, and contains information about individual student characteristics (e.g. learning style) as well as subject knowledge. The gathering and updating of information in the student model ('student modelling') may be *static* (captured once at the start of the course, e.g. a learning styles questionnaire (Papanikolaou et al., 2003)) or *dynamic* (continuously or periodically during tutoring, e.g. preferred learning resource choices (Popescu, 2010)). Although dynamic student modelling offers the advantage of a more current student model, the gathering of reliable information is difficult (as it is often uncertain and imprecise) and sometimes results in a weak student model (Brusilovsky and Millan, 2007). There are two main approaches to student modelling (Brusilovsky, 1996):

- *Collaborative* modelling is where the student provides information and feedback on preferences, e.g. Papanikolau et al. (2003) gathered student learning styles preferences at the start of a tutoring session by asking students to complete a questionnaire. Students could then directly alter their learning preferences in the model.
- *Automatic* modelling is where information on student behaviour is gathered during a tutoring session rather than explicitly from the student, e.g. Garcia et al. (2007) used a history of student choices to model learning style and Ammar et al. (2010) used facial expression recognition to model learner emotion.

Adaptation is incorporated in ITS by two different approaches: *adaptive presentation*, which involves presenting different learning content, and *adaptive navigation*, which involves recommending or reordering learning content (Brusilovsky and Peylo, 2003). Adaptive navigation is normally only found in adaptive hypermedia systems (AHS), where students have the initiative to select learning material from a menu of links. In the context of this thesis, adaptation refers to the *adaptive presentation* of learning content matched to individual student characteristics.

Of the three intelligent techniques used in ITS, curriculum sequencing is the most commonly found. However, curriculum sequencing alone is little better than selecting chapters from a book. By combining all three intelligent approaches, an ITS can get close to offering the support available from a human tutor. Few ITS incorporate all three intelligent approaches (Graesser et al., 2005a; Woo Woo et al., 2006; Melis, 2001) as they are complex and time-consuming to develop. However, combining all three technologies adds benefits by offering a more effective learning experience and intelligent support that can help to build confidence and motivation (Graesser, 2008b).

The next section will outline the modelling and adaptation of individual learning styles in ITS.

3 ITS and Learning Styles

Most ITS personalise learning by adapting to a student's existing knowledge of a particular subject. The extension of ITS to adapt to other student characteristics, such as learning style, is a new area of research. As described in Chapter 2, learning styles describe the way groups of people prefer to learn. Research is divided on whether adapting to individual learning styles in computerised learning systems improves learning (Hsieh et al., 2011; Rasmussen, 1998; Riding and Grimley, 1999; Graff, 2003) or makes no difference (Ford and Chen, 2000; Shih and Gamon, 2002) to learning outcomes. However, learning styles have been widely used to enhance ITS by presenting learning material matched to individual students' preferred styles (Popescu, 2010; Villaverde et al., 2006; Garcia et al., 2007; Wang et al., 2006; Stash and De Brau, 2004; Spallek, 2003). When adapting to learning styles in ITS there are two main challenges:

- How will learning styles be modelled?
- How will the system adapt to individual learning styles?

Each of these key questions will now be discussed.

3.1 Modelling Learning Styles

As discussed in Chapter 2, there are conflicting opinions about whether learning styles are fixed traits or whether they change over time, by subject or the environment. Similarly, the way in which learning styles are assessed by ITS is under debate – whether the modelling of learning styles should be static (measured at the start of the learning) or dynamic (modelled periodically or continuously). There are two main methods for ITS to model learning styles: explicitly using the measuring instrument defined by the learning styles model, or implicitly using learner behaviour in the ITS.

3.1.1 Collaborative Modelling Using Questionnaires

The simplest way to measure learning styles is using the formal assessment described in the adopted learning styles model, normally a questionnaire (Wang et al., 2006; Spallek, 2003). However completing questionnaires is onerous for students, who do not always lend enough time or attention to complete them accurately. It is also difficult to avoid unintended influences in the questions, with some answers perceived as being better (Popescu, 2009). Also, as the modelling is static, if learning styles change over time or subject, the student model will not be accurate. These problems can lead to an unreliable student model (Yannibelli et al., 2006). The 'open model' approach allows students to modify their profile directly, and has been used to extend static modelling using questionnaires (Papanikolau et al., 2003). Whilst the open model approach gives increased learner control and feedback on the quality of the system model, it also increases the learner's workload and relies on the learner's understanding and knowledge of their preferred learning style.

Some examples of ITS that use questionnaires to model learning styles are:

- CS383 (Carver et al., 1999) was one of the first AHS to adapt to learning styles, modelling three dimensions of the Felder and Silverman model using the Index of Learning Styles (ILS) (Felder and Soloman, 1998).
- CooTutor (Wang et al., 2006) models the Felder and Silverman learning styles using the ILS questionnaire.
- INSPIRE (Papanikolau et al., 2003) uses the Learning Styles Questionnaire to model learning styles (Honey and Mumford, 1992). Learners can also directly modify the student model to reflect their preferences.
- AES-CS (Triantafillou et al., 2004) models the field dependence or independence (Witkin, 1962) of learners using a questionnaire. Learners may also alter the behaviour of AES-CS by changing options such as the amount of feedback given.
- iWeaver (Wolf, 2003) uses a questionnaire to initialise the model of the Dunn and Dunn (1974) learning styles. Learners are given an explanation of their

learning style and recommendations for style of resources, but may adjust the model by choosing other styles of resource. After each module, learners give feedback on the learning resources they use, with a ranked list which is used to adjust the student model.

In summary, although collaborative modelling of learning styles using questionnaires is the simplest method, the model will only be accurate if students lend enough time and attention to complete the questionnaire properly (Garcia et al., 2007).

3.1.2 Automatic Modelling Using Learner Behaviour

Implicit modelling of a student's learning style involves building and updating a student model automatically based on the student's behaviour and actions while they use an ITS for learning (Villaverde et al., 2006; Stash and De Brau, 2004; Garcia et al., 2007). Modelling learning style dynamically and continually updating the student model enables an ITS to adapt to changes in learning style over time or for different subjects. Implicit modelling removes any requirement for input by the student so they can concentrate on their learning task; however it is difficult to extract enough reliable information to build a robust student model. Some ITS have overcome this problem by designing interfaces with the goal of collecting data to model learning styles (e.g. Cha et al., 2006). However the main goal of an ITS is to intelligently help students to learn, so interface design should focus on promoting learning. Another way to overcome the problem of reliability is to adopt a mixed-modelling approach, initially modelling learning style using a questionnaire and then dynamically updating the model (e.g. Paredes and Rodriguez, 2004).

The types of learner behaviour used to model learning style include navigation and browsing patterns, the choice of resources (including time spent and frequency of access), the use of chat forums and test performance (Popescu, 2009).

There have been many different approaches to the automatic modelling of learning styles, including:

- Bayesian networks are probabilistic models that have been used to model the relationships between learning styles and behaviour factors. Garcia et al. (2007) used Bayesian networks to infer student learning styles from a history of their behaviour in using the ITS. Three dimensions of the Felder and Silverman (1988) model were modelled, with precisions of 58-77%. Enhancing the Bayesian model improved precisions to 66-80% (Garcia et al., 2008). EDUCE (Kelly and Tangney, 2004) offers different resources styled using four of the Gardner (1983) multiple intelligences, and uses Naïve Bayes to predict which resources students prefer, based on past choices.

- Artificial neural networks are computational models inspired by the neural structure of the brain, which have been used to classify student learning styles based on behaviour. Villaverde et al. (2006) used a neural network to determine student learning style for three dimensions of the Felder and Silverman (1988) model. The neural network analyses recent student behaviour in an ITS to automatically model learning style, achieving an accuracy of 69.3%. Hsu et al. (2010) used fuzzy inference rules to construct a neural network that identifies the relationship between learning activities and learning style. However, neural networks are less reliable for large amounts of input data (i.e. behaviour factors), (e.g. only ten behaviour factors (input neurons) were used by Villaverde et al., 2006) so this may not be enough to accurately model learning style. The opaque nature of neural networks also means that no information is learned about which behaviour factors are most significant in predicting a learning style.
- Genetic algorithms are adaptive heuristic search algorithms that mimic the process of evolution by natural selection. Yannibelli et al. (2006) adopted a genetic algorithm approach to model three dimensions of the Felder-Silverman (1988) learning styles model based on student behaviour in an ITS.
- Rule based methods involve modelling learning styles using rules that map patterns of behaviour extracted from learning styles models to learner behaviour. The DeLes tool (Graf et al., 2009) uses a rule-based method to infer student learning styles from their behaviour in a general Learning Management System (e.g. Moodle, 2011). DeLes models students using the Felder-Silverman (1988) model, and achieved precisions of 73-79%. WELSA (Popescu, 2009) also uses rule-based modelling (based on over 100 patterns of behaviour gathered from choice of learning resources, navigation and communication) to model learning styles using the Unified Learning Style Model (Popescu et al., 2007).
- Decision trees are models that can predict the value of a variable (e.g. learning style) based on a number of input variables (e.g. behaviour factors). Cha et al.

(2006) designed their ITS interface to capture behaviour related to the Felder and Silverman (1988) model, using decision trees and Hidden Markov Models to classify learning styles from learner choices and behaviour. Ozpolat and Akar (2009) used the NBTree algorithm to classify student learning styles from the content of learning objects rather than behaviour (using keywords matched in Internet search terms). Chen and Liu (2008) used decision trees and K-means clustering to automatically identify cognitive styles from learning patterns.

- Other research has involved plotting clusters of types of learners against behaviour factors, including a 'dead band' where the learning style is classified as 'unknown' (as in Sanders and Bergasa-Suso, 2010). Klasnja-Milicevic et al. (2011) clustered learners based on their learning style and then used the AprioriAll pattern mining algorithm to extract behavioural patterns from log files. By comparing learner behaviours to each cluster, learning style was identified and learning material recommended.

Central to all of these automatic modelling methods is the capture of behaviour characteristics during use of the ITS. It is difficult to decide on the most appropriate behaviour to model and selecting typical behaviours that discriminate between learning styles requires a detailed analysis of the chosen learning styles model. Even then, students do not always behave stereotypically as suggested by learning styles models (Coffield et al., 2004a; Garcia et al., 2007). There are clear differences in the number of behaviour characteristics used by ITS to model learning styles – e.g. Garcia et al. (2007) capture 11, Cha et al. (2006) capture 58 and Popescu (2010) captures over 100. This does not always lead to different levels of precision in modelling learning styles, as different modelling methods and learning styles models have different requirements.

This section has described the main methods of modelling learning styles, either from explicit learner information gathered from learning styles questionnaires or implicitly from learners' behaviour while using an ITS. The next section outlines how adaptation to individual learning styles is approached.

3.2 Adaptation to Learning Styles

ITS normally adapt by presenting learning material matched to student preferences with the aim of making learning easier and more effective. Opinion is divided about whether learning styles are static or dynamic (see Chapter 2) and some researchers (Felder and Spurlin, 2005) argue that learners should be presented with mismatched content to strengthen their weaker learning styles. This longer term goal has not yet been incorporated into ITS, which are concerned with providing an effective learning experience for the current topic.

Having modelled the student learning style, the method of adapting the tutoring to suit individual preferences must be considered. Learning style models normally describe several aspects (or dimensions) of student preference relating to stages of the learning process (see Chapter 2). To adapt to all learning styles in a model would require creating multiple copies of learning material, which is a large and time consuming development task. For example, in the Felder and Silverman (1988) model there are 32 different learning styles. To overcome this problem, some ITS only adapt to a subset of learning style aspects. For example, iLessons (Sanders and Bergasa-Suso, 2010) and LSAS (Bajraktarevic et al., 2003) adapt to a single dimension of the Felder and Silverman (1988) model. However the problem is that it is difficult to decide which aspects of learning style to model, and also only part of a student's learning style is being addressed. EDUCE (Kelly and Tangney, 2006) models half of Gardner's (1983) multiple intelligences, but must still provide four versions of learning resources. Also, by selecting the most preferred learning style, students are only presented with one type of learning resource. The literature does not provide details of adaptation algorithms or conflict resolution when two aspects of learning styles are equally strongly preferred. Some ITS (Carver, 1999; Kelly and Tangney, 2006) have addressed this issue by relying on students' initiative to select different types of learning resource from a list of links. An adaptation algorithm that selects a variety of learning resources based on several aspects of a student learning style may provide a richer learning experience.

Some examples of ITS which adapt to learning styles are:

- CS383 (Carver, 1999) implements adaptive navigation according to three dimensions of the Felder and Silverman (1988) model by reordering a list of media elements ranked with the most suitable items for an individual's learning style at the top.
- MANIC (Stern and Woolf, 2000) provides adaptive material using the 'stretchtext' technique, where basic material is presented to all learners which can be enriched with supplementary adaptive material, such as different media or

instruction styles. MANIC does not adopt a particular learning styles model, but incorporates aspects of different models, e.g. Felder and Silverman (1988).

- EDUCE (Kelly and Tangney, 2006) adapts the presentation of learning material according to the Gardner (1983) theory of multiple intelligences. Adaptive presentation is implemented using page variants, which involves creating different versions of each page with different presentations of the content suited to multiple intelligences. Learners may go back and choose other versions of the learning material using links. Adaptive navigation is provided with direct guidance (using the 'next' button) and also link hiding (hiding links to unmatched resources). Adaptation is selected based on the strongest of four intelligences included. Learners can also choose to turn off the adaptivity, and take the initiative in choosing resources to view.
- INSPIRE (Papanikolau et al., 2003, 2006) allows learners to select a goal and adapts according to their level of knowledge, progress and learning style. Curriculum sequencing and adaptive navigation are based on learners' goals, progress and knowledge whereas adaptive presentation style is based on the Honey and Mumford (1992) learning styles.
- The iLessons AHS (Sanders and Bergasa-Suso, 2010) adapts links to material retrieved from the Internet based only on the Active/Reflective Felder and Silverman dimension.

As discussed above, no adaptation algorithm or details of conflict resolution for choosing between two equally preferred aspects of learning style have been published. For simplicity and to avoid the need for many versions of learning resources, adaptation is often restricted to a subset of learning style dimensions (e.g. a single dimension as in Sanders and Bergasa-Suso, 2010; Bajraktarevic et al., 2003). However, by only adapting to one dimension of their learning style, such ITS only partially address a student's learning preferences. The challenge of reducing development time of multiple copies of adaptive learning resources whilst still providing the best adaptive learning experience for students remains unresolved.

3.3 Summary of Learning Styles in ITS

Extending ITS to adapt tutoring based on individuals' preferred learning styles has been shown to enhance students' learning experiences (Sangineto et al., 2007;

Kelly and Tangney, 2005; Triantafillou et al., 2004). The inclusion of alternate approaches to learning and instruction based on individual preferences adds a human-like social intelligence to ITS. However, the decision of how best to model individual learning styles is a complex task, and many different approaches to automatic modelling have been tried. The choice of adaptation method is equally complex and dependent on the extent of student initiative within the ITS. ITS that adapt to individual learning styles offer advantages over classroom tutorials, where human tutors do not generally have time to assess and adapt to individual learning styles.

4 Conversational Interfaces to ITS

ITS are normally designed to be student-directed, with a system of menu choices or hyperlinks that are reordered or ranked to recommend a particular sequence to learners (Klasnja-Milicevic et al., 2011). Whilst this design simplifies the analysis of student behaviour, it does not truly teach the students but rather assists in self-learning, and is often little different to recommending chapters of a book. Although rarely employed, conversational interfaces allow a more natural, teacher-led learning experience which supports the construction of knowledge used by human tutors (Chi et al., 2001). Conversational agents (CA) are complex and time consuming to develop (see Chapter 2), however the benefits of constructivist styles of learning are widely accepted (Sharples, 2005; Graesser et al., 2008b). CAs add another dimension of intelligence to ITS, as they can mimic the main method of human tutor communication, i.e. conversation, allowing students to experience collaborative problem solving similar to with human tutors. *Conversational ITS* (CITS) can act as personal tutors that are available at any time and place, and which provide instant answers to questions, feedback, and support. Moreno et al. (2001:179) suggest that as learning is an “inherently social process”, using CITS “learners interpret their relation with the computer as a social one involving reciprocal communication”. To mimic a human tutor, ITS should support the construction of knowledge through discussion: “it seems necessary for future generations of ITSs to incorporate natural language capabilities.” (Chi et al., 2001:518).

Despite the development barriers, there has been some success in enhancing learning using CA interfaces, for example:

- The best-known and most-developed CITS is AutoTutor (Graesser et al., 1999), which teaches computer literacy and will be reviewed separately in section 4.1. Why2-Atlas (VanLehn et al., 2002) was developed alongside AutoTutor and helps students build understanding of conceptual physics problems. RMT (Arnott et al., 2008) is also a descendent of AutoTutor which includes an animated CA tutor to teach introductory research methods using natural language.
- CIRCSIM-tutor (Woo Woo et al., 2006) incorporates a CA to engage medical students in discussion to solve physiology problems, modelling the diagnosis of a patient and enabling students to practice causal reasoning and the application of knowledge. CIRCSIM-tutor asks questions and produces hints to help students construct knowledge, however its understanding of language is limited to only a few student initiatives, and is very much linked to the field of medical diagnosis.
- TeachMed (Rahati and Kabanza, 2010) is an adversarial CITS that enables students to interview virtual patients to diagnose medical problems. In TeachMed, the CITS attempts to persuade students about the rationale of tutoring hints by selecting an appropriate argument. However, the natural language capability of TeachMed is very limited (in fact, students are offered move choices and must use text templates to reply).
- Beetle II (Dzikovska et al., 2010) is a CITS for tutoring basic electronics. Students are shown fixed slides with questions, are able to experiment using the circuit simulator and then explain their answers in natural language. Although hints are given in natural language, the tutoring content is not adaptive.
- AVIS (Kumar et al., 2010) is a CA tutor with ‘social conversational skills’ that supports small teams of learners in a collaborative learning scenario. AVIS plays the part of a tutor in a team working on a computer-aided mechanical engineering design project by implementing social interaction strategies such as encouraging inactive members. AVIS annotates student utterances with semantic categories and identifies inactive students to trigger social interaction strategies. However, the social strategies described can be expected to be integrated into any CITS mimicking a human tutor (e.g. cheerfulness, reassurance) and so engaging inactive students appears to be the only novel feature.

As discussed in sections 2 and 3, adapting tutoring to individual differences can aid learning. Some CITS adapt to subject-independent learner characteristics such as emotions (Kumar et al., 2010, Graesser et al., 2008a), however none consider student learning styles during tutoring. As described in Chapter 2, learning styles research suggests that students process and represent knowledge in different ways, and so prefer different styles of teaching. Adapting a conversational tutorial style to match individual preferences could improve student confidence and motivation, enhancing the learning experience. Incorporating adaptation to individual learning styles into a CITS remains an opportunity for improvement.

Extending CITS to incorporate spoken interfaces would widen access to ITS further and is a relatively straightforward task, however there are technology limitations to overcome before this is practical (D'Mello et al., 2010a). The biggest challenge for CITS (as for CAs) is understanding user input: when a CA does not understand the learner, "such breakdowns in comprehension run the risk of eroding the learner's confidence in the intelligence of the agent." (Graesser et al., 2005a:162).

The best-known CITS, AutoTutor (Graesser et al., 1999), has been the focus of extensive research for over a decade and will now be reviewed separately.

4.1 AutoTutor Conversational ITS

AutoTutor (Graesser et al., 1999, 2004, 2005a) is a sophisticated CITS that allows students to construct knowledge about computer literacy and Newtonian physics through discussion. AutoTutor uses an animated CA to present tutoring questions and engage in mixed initiative dialogue whilst guiding the student towards constructing a solution. As well as showing AutoTutor's response textually on screen, the animated CA speaks the text and shows appropriate facial expressions and gestures, although it is the dialogue content rather than the animation and speech that influences learning (Graesser et al., 2003). All intelligent technologies described in section 2 are implemented. A detailed model of learner knowledge is created by comparing student answers to a database of expectations and misconceptions associated with the tutorial question (using the latent semantic analysis approach (Graesser et al., 2005b)). During the conversation AutoTutor coaches the student to cover expectations and corrects misconceptions using a set of dialog moves such as giving feedback, prompting for more information and correcting bad answers. Students can ask a broad range of questions (known as *inquiry learning*) which

AutoTutor interprets and then responds by retrieving a ‘canned text’ answer from an ebook. AutoTutor was shown to improve learning gain (Graesser et al., 2003), however studies were conducted in a controlled laboratory setting rather than in a natural learning environment, and so participants were less liable to distractions.

One of the main problems with CA interfaces is the time and complexity of developing conversation scripts. Lesson authoring tools have been created for AutoTutor to enable the tutoring domain to be changed while reusing the CA components of the system (Susarla et al., 2003).

Students communicate with AutoTutor by typing in their utterance into a text box, although recently a speech recognition facility has been included (D’Mello et al., 2010a). The speech facility did not improve learning gain when compared to typed conversation, but more content was covered (as speaking is more efficient than typing) although speech recognition errors affected student feedback.

New research has involved extending AutoTutor to adapt to learner emotions as well as their knowledge (D’Mello et al., 2009; Graesser et al., 2008a). Emotions such as boredom and frustration are modelled using sensors to detect facial expressions and body posture as well as dialogue patterns. AutoTutor responds by, for example, giving a hint to a frustrated student or giving a challenging problem to a bored student. Adapting to emotions improved deeper learning for students with low knowledge, however some students were irritated by the empathic AutoTutor (D’Mello et al., 2010b).

As described above, the extensive research on AutoTutor has contributed much to the CITS field. Despite promising results, experiments using AutoTutor were conducted in a controlled laboratory setting and not in a real-life learning environment, where learning is affected by other factors, like distractions. For example, the detection of boredom and confusion from dialogue patterns alone was poor (D’Mello et al., 2008). The necessity for sensors to detect emotion in a real-life learning environment has many difficulties as facial recognition is a complex task. As well as requiring cameras with sufficient resolution (which is expensive), successful recognition of emotions requires a consistent environment, for example adequate lighting and learners seated in a fixed position. These barriers constrain the widespread use of AutoTutor in real learning environments, and significantly restrict its accessibility to strictly controlled laboratory environments.

4.2 Summary of Conversational ITS

Conversational interfaces to ITS can more effectively mimic face-to-face tutorials as they are conducted through natural language dialogue. CITS can help to improve learner confidence and motivation as they are intuitive to use and learners can construct knowledge and solutions to problems through discussion. However, CA technology is in its infancy, and developing a convincing ‘intelligent’ CA interface is challenging and time consuming. Although social factors such as mood have been incorporated, the accuracy of detecting emotion from dialogue alone is poor. Learning styles represent a significant aspect of the complex process of learning which could enhance learning in CITS (see Chapter 2). Yet there are currently no Conversational ITS that can detect learning styles from tutoring dialogue and adapt the conversational tutoring to individual’s learning styles.

5 Challenges for ITS

ITS are not in widespread use, despite offering many advantages over traditional computerised learning systems. This may be due to the lengthy development time or the lack of interoperability and reusability between adaptive systems (the ‘open corpus problem’ (Brusilovsky and Henze, 2007)). There are a number of challenges and open questions in ITS research, as follows:

- Like most expert systems, ITS are complex and time consuming to develop. For example, Murray (1999) reported that one hour of ITS instruction requires an estimated 100 hours of development time. Aside from the challenges of replicating human intelligence in tutoring, such as solution analysis and problem solving support, deciding on the best factors to model and use for the personalisation of learning is not straightforward (as discussed in sections 2 and 3).
- The development and maintenance of adaptive tutorial presentations is a labour intensive process requiring the design of multiple copies of the same content in different styles. Authoring tools aim to speed up development time (e.g. Stash et al., 2004), but the time spent capturing adaptive material from human tutors is much more significant than that spent formatting learning resources. The question of how best to reduce development time of adaptive learning resources whilst still providing the best adaptive learning experience for students remains open.

- The design of learning resources for reuse has not been implemented widely for ITS, as it has in virtual learning platforms (SCORM, 2004). The reuse of learning resources could reduce development time and cost, but ITS are typically proprietary systems with their own learning resources. iLessons (Sanders and Bergasa-Suso, 2010) is an AHS that reuses learning resources retrieved from the Internet which have been annotated (with respect to teaching/learning style) by human tutors. However, reusing learning resources can lead to an incoherent learning experience as portability needs to be considered during the development of the resources (Boyle, 2003). Incorporating adaptable learning resources into a learning design framework (such as Boyle, 2010) could bring the benefits of reuse into ITS research. KOD Packager (Karagiannidis and Sampson, 2004) includes templates that extend the learning resource metadata to describe factors of individual learner characteristics, e.g. learning style.
- There is some debate about whether ITS are significantly better than existing learning management systems or than reading a book (Woo Woo et al., 2006; Graesser et al., 2004).
- Scalability is important. The use of ITS over the Internet requires systems to cope with large numbers of individual tutorials simultaneously in real time, and response times must be fast enough to maintain student motivation.
- It is a complex task to create a computer system that appears intelligent, e.g. Woolf et al. (2001) identified difficulties in generating believable, life-like responses in an instructional dialogue.
- There are open questions when introducing adaptation, such as which models of individual differences should be adopted, which behaviour characteristics best indicate and discriminate between learning styles, what type of adaptation to include and which intelligent techniques are best for dynamically modelling and adapting to learning preferences (Papanikolaou et al., 2006).

The main barrier to the prevalent adoption of ITS lies in their lengthy development, which requires a sophisticated method of adding ‘intelligence’ and the production of different styles of learning material. However, the many benefits offered by accessible, individualised tutoring in an environment of financial restrictions and increasing workloads indicate a promising future for ITS.

6 Conclusion

Learning is a complex process, and so the design of computerised learning systems should incorporate different approaches to tutoring to accommodate the variety of different learners. ITS have the opportunity to offer something more than information delivery – offering personalised one-to-one tutoring is rarely possible in higher education environments. Also human tutors rarely have time to diagnose and adapt to learning styles on an individual basis. Instead, they use knowledge of learning styles to informally group learners and adapt their teaching style to the general population of the classroom. ITS can give instant feedback and individual assistance in constructing solutions to problems, and promote a deeper understanding of the topic. Delivered via the Internet, ITS tutoring is available anywhere at any time, and learners can repeat the same topic until they understand. However, ITS are not widely used. There are a number of challenges for ITS, including the lengthy development time, the reusability of learning material and a lack of real social intelligence. Incorporating individual aspects such as mood and learning styles makes ITS more socially intelligent and can improve the learning experience.

Many ITS are student directed, however some students lack the initiative and maturity to take responsibility for their own learning. Conversational interfaces can add a social element to ITS, as learning is tutor-led and conducted through a natural language dialogue. Conversational ITS (CITS) aim to mimic a human tutor by supporting the construction of knowledge through discussion, and learners can draw on their experience of classroom tutorials, helping to increase confidence. Learners see the conversational ITS as a collaborative partner, which can improve engagement and motivation. “By mimicking the conversational interaction of a human tutor….. [CITS] is seen as providing the student with “a conversational partner” or a “simulated social presence” well-suited to the social and conversational processes of learning” (Friesen, 2009:103). Although CITS have been extended to adapt to mood and emotion, there are no CITS that adapt tutoring to individual learning style preferences. With respect to this research, the personalisation of a learning experience by tutoring in a style suited to an individual’s learning styles represents a human-like social intelligence. If it were possible to extend such automated personalisation to an ITS with a natural language interface, a more socially intelligent and human-like conversational ITS would result.

This chapter has given an overview of the current state of research into Intelligent Tutoring Systems. The methods of including intelligence were outlined, and a review of how ITS personalise tutoring to individual learners. Drawing on the overview of learning styles theories in Chapter 2, the methods employed by ITS to model and adapt to learning styles in order to improve learning effectiveness were discussed. No adaptation algorithms or methods for dealing with conflict when several learning styles are equally preferred have been published. The benefits of including conversational agents (described in Chapter 3) as interfaces to conduct tutoring in ITS were described, and the outstanding challenges for ITS research listed.

7 Chapter Highlights

- Intelligent Tutoring Systems (ITS) extend traditional content-delivery computerised learning systems by using intelligent technologies to personalise learning.
- ITS add intelligence by curriculum sequencing, problem solving support and intelligent solution analysis.
- ITS personalise learning based on domain-specific factors, i.e. existing knowledge, and learner-specific factors such as emotion, mood and learning style.
- Incorporating adaptation requires the building of a student model that is used to direct the adaptive navigation or presentation of the tutorial.
- Learning styles are modelled explicitly using formal evaluation (questionnaires) or implicitly from learner behaviour.
- Adaptive presentation requires multiple copies of learning material in different styles.
- Conversational interfaces are rare in ITS despite allowing learners to construct knowledge through discussion.
- There are no Conversational ITS that adapt to individual learning styles.
- Research into ITS is an active area with many challenges and open questions.

Chapter 5 Predicting Learning Styles from a Natural Language Dialogue

1 Introduction

In Chapter 3 the benefits of natural language interfaces in offering intuitive, human-like interaction through conversation were described. Chapter 4 introduced Intelligent Tutoring Systems (ITS) and the advantages of personalised tutoring in improving motivation and the effectiveness of the learning experience. Online ITS present flexible learning at any time and place at a fixed delivery cost and at a learner's own pace. Conversational Intelligent Tutoring Systems (CITS) offer a more human-like tutoring experience by employing natural language interfaces to tutor through discussion. Like human tutors, CITS can apply the constructivist approach to learning which enables students to construct their own knowledge about a subject through discussion, and leads to a deeper understanding of the topic. As discussed in Chapter 4, although CITS are rare as they are complex to develop, some CITS exist that can detect emotion as well as existing knowledge (D'Mello et al., 2010b) to personalise tutoring. Chapter 2 described how learning experiences can be made more effective when a range of teaching styles are used which take account of different learning styles. Although several ITS personalise learning according to learning style, there are no CITS that mimic a human tutor by dynamically predicting and adapting to learning styles during a tutoring conversation. If it were possible to mimic a human tutor by picking up cues from students about their learning styles during a tutoring conversation, a CITS could dynamically predict learning styles and adapt its tutoring style to individual learners.

This chapter will describe the development of an approach for predicting learning styles from a natural language tutoring dialogue with a CITS. As no existing CITS can predict learning styles from a natural language dialogue, a number of different prediction methods will be investigated which can then be assessed empirically. The combination of different strategies results in an original methodology for extracting knowledge from a learning styles model in order to dynamically predict learning styles from a two-way natural language tutoring conversation. The methodology is independent of a particular learning styles model, so an exemplar model (the Felder-Silverman (1988) model) has been chosen for illustration.

2 Detection of Learning Styles

As discussed in Chapter 4, ITS detect learning styles in two different ways – using the assessment built into the learning styles model (usually a questionnaire) or by analysing learner behaviour throughout tutoring. The problem with questionnaires is that learners find them time consuming and onerous to complete, so often do not complete them accurately, leading to incorrect assessment of learning styles (Popescu, 2009). There is also some debate in the literature about whether learning styles change over time or are different for different subjects (as discussed in Chapter 2). Therefore predicting learning styles dynamically throughout the tutoring session could address these problems. Menu-based ITS have analysed typical learner behaviours (associated with learning styles) to predict learning styles (Hsu et al., 2010; Garcia et al., 2008; Villaverde et al., 2006; Kelly and Tangney, 2004), sometimes being designed specifically to collect such data (Cha et al., 2006). However, none have attempted to predict learning styles from a natural language tutoring conversation.

In order to develop a strategy for predicting learning styles from a natural language tutoring dialogue it is necessary to adopt an example learning styles model. As described in Chapter 2 (section 3.5), the Felder-Silverman (FS) learning styles model (Felder and Silverman, 1988) describes the learning styles of engineering students over four dimensions that relate to the process of receiving and understanding information (see Figure 5.1).

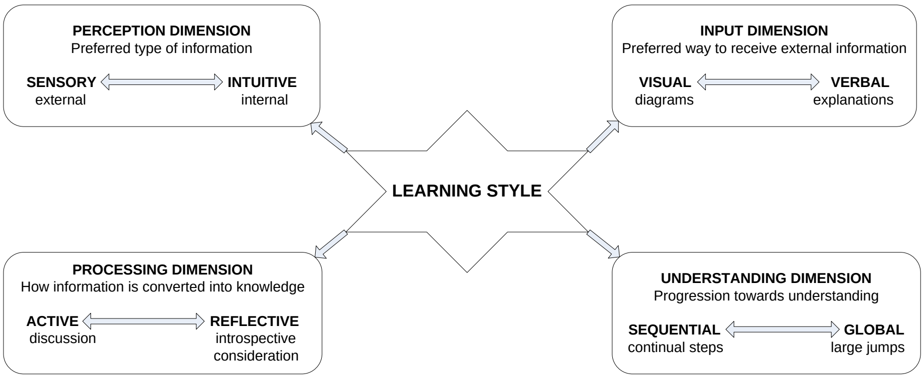


Figure 5.1. Dimensions of the Felder-Silverman Learning Styles Model

The FS model has been adopted by many of the Intelligent Tutoring Systems (ITS) that adapt to student learning styles (Dag and Gecer, 2009; Garcia et al., 2007; Cha et al., 2006; Wang et al., 2006; Sancho et al., 2005; Yannibelli et al., 2006;

Villaverde et al., 2006). As described in Chapter 2 (section 3.5) the FS model offers benefits to ITS as:

- it includes detailed descriptions of learning behaviour and teaching styles,
- it has a small number of dimensions that are distinct and independent,
- it describes engineering students who often make up experimental groups.

The FS model was selected as an exemplar learning styles model for the development of this generic methodology for these reasons as the participants of the initial experimental studies will be undergraduate engineering students.

A number of different methods of predicting learning styles from natural language tutoring dialogues will be considered, including:

- An evaluation of the FS model evaluation instrument to establish the most accurate predictor questions.
- An analysis of learning style knowledge in the FS model.
- Mapping of behaviour cues which discriminate between learning styles to a natural language tutoring dialogue.
- An analysis of words and phrases that may be indicative of learning style.
- The development of logic rules to encapsulate the learning styles knowledge extracted from the FS model.

3 Index of Learning Styles Study

The first step in creating a strategy to detect the learning styles of engineering students during a tutoring conversation was to consider the assessment used by the FS model. The Index of Learning Styles (ILS) (Felder and Soloman, 1997) is an instrument used to assess the FS learning styles. The ILS is a self-assessment questionnaire (see Appendix 1) containing 44 questions, 11 questions for each of the four learning style dimensions. For each question there are two possible answers (*a* and *b*), and learners must choose the answer that applies most of the time. After completing all questions, the total number of *a* and *b* answers for each FS dimension are compared, and the higher total represents the learning style. For example, for the Active/Reflective FS dimension, if the questionnaire totals are *a*=3 and *b*=8, the overall learning style is Reflective as the total of *b* answers is higher.

Incorporating all 44 questions from the ILS questionnaire into a single tutorial in order to implicitly detect learning styles is not practical. Therefore, if assessment questions from the ILS questionnaire were to be incorporated into a tutorial, it would be necessary to reduce the number of questions. The ILS questionnaire has been analysed to add semantic information (Graf et al., 2007), but if only a subset of questions are selected from the ILS questionnaire, criteria will be needed to select the best questions. For the purpose of the research presented in this thesis, a study was designed to analyse the ILS questionnaire with the aim of discovering which questions best predict an individual's learning style. The set of 'best predictor' questions could then be considered for inclusion in a conversational tutorial to assist in predicting learning styles. This study, reported in Latham et al. (2009) will now be described.

3.1 Experimental Design

The aim of the study was to extract the subset of questions from the ILS questionnaire whose answers most often agreed with the overall learning style, i.e. they are best at predicting the overall result. The sample size was determined using the non-probability quota sampling approach (Walonick, 2010), with a target sample size of 100 randomly selected adults aged over 18. This approach was most appropriate because the population of adults over 18 is unknown and the ILS questions will be examined for relationships with learning styles rather than to draw conclusions about the population. The actual sample group size for the study was 108 participants. All participants were adults who were studying (or had previously studied) on a computing graduate or post-graduate qualification. Participants for this study were selected from this group because the participants in the initial experimental studies will be undergraduate computing students.

3.2 Methodology

Each participant was asked to complete the ILS self-assessment questionnaire, either online, on paper or using an emailed electronic form. The ILS questionnaire and its results were then analysed. Each question of the ILS relates to one of the four dimensions of the model. For each complete questionnaire, the answer to each question was compared to the overall result for the related learning style dimension. Each question was assigned a score as follows:

- where the question answer matched the overall result for the learning style dimension, a score of 1 was assigned;

- where the question answer did not match the overall result for the learning style dimension, a score of 0 was assigned.

Once the results of all participant questionnaires were compiled, each question was associated with an overall score, which was then converted into a percentage. The percentage indicates each question's accuracy in predicting the overall result of the learning style dimension. The questions were then organised by learning style dimension and sorted according to the prediction accuracy. The accuracy level for best predictor questions was empirically set at 70% as this level gave the best accuracy and distribution of questions across the learning style dimensions. Those questions with a prediction accuracy of at least 70% were included in the subset of 'best predictor' questions.

3.3 Results and Discussion

Of the 108 participants, 5 questionnaires were incomplete as not all questions had been answered – these were excluded, leaving 103 completed questionnaires. Table 5.1 shows the distribution of learning styles across the sample group of 103. Most FS dimensions have an approximately equal split, except for the Visual/Verbal dimension, where there are twice as many Visual learners as Verbal. This was expected, as the FS model states that there are more Visual learners than Verbal.

Table 5.1. ILS Study Distribution of Learning Styles

LEARNING STYLE	n	%
Active	52	50%
Reflective	51	50%
Sequential	53	51%
Global	50	49%
Sensory	59	57%
Intuitive	44	43%
Visual	69	67%
Verbal	34	33%

Table 5.2 shows the resulting best ILS predictor questions for each learning style dimension, using a boundary of 70% accuracy. Of the 44 questions on the ILS questionnaire, 26 questions (spread over all four dimensions) could predict the overall learning style with an accuracy of at least 70%, and 17 questions with at least 75% (denoted by * in Table 5.2). The three best predictor questions had an accuracy of 84%, and all related to the Visual/Verbal dimension.

Table 5.2. Best Predictor Questions in the ILS Questionnaire

FS DIMENSION	PREDICTION ACCURACY	ILS QUESTION	
Active/ Reflective	77%	*Q17	When I start a homework problem, I am more likely to (a) start working on the solution immediately (b) try to fully understand the problem first
	77%	*Q25	I would rather first (a) try things out (b) think about how I'm going to do it
	74%	Q33	When I have to work on a group project, I first want to (a) have "group brainstorming" where everyone contributes ideas (b) brainstorm individually and then come together as a group to compare ideas
	73%	Q5	When I am learning something new, it helps me to (a) talk about it (b) think about it
	71%	Q9	In a study group working on difficult material, I am more likely to (a) jump in and contribute ideas (b) sit back and listen
Sequential/ Global	78%	*Q44	When solving problems in a group, I would be more likely to (a) think of the steps in the solution process (b) think of possible consequences or applications of the solution in a wide range of areas
	76%	*Q24	I learn (a) at a fairly regular pace. If I study hard, I'll "get it" (b) in fits and starts. I'll be totally confused and then suddenly it all "clicks"
	75%	*Q36	When I am learning a new subject, I prefer to (a) stay focused on that subject, learning as much about it as I can (b) try to make connections between that subject and related subjects
	71%	Q28	When considering a body of information, I am more likely to (a) focus on details and miss the big picture (b) try to understand the big picture before getting into the details
	70%	Q12	When I solve math problems (a) I usually work my way to the solutions one step at a time (b) I often just see the solutions but then have to struggle to figure out the steps to get to them
Sensory/ Intuitive	83%	*Q10	I find it easier (a) to learn facts (b) to learn concepts
	79%	*Q38	I prefer courses that emphasize (a) concrete material (facts, data) (b) abstract material (concepts, theories)
	78%	*Q6	If I were a teacher, I would rather teach a course (a) that deals with facts and real life situations (b) that deals with ideas and theories
	78%	*Q22	I am more likely to be considered (a) careful about the details of my work (b) creative about how to do my work
	77%	*Q2	I would rather be considered (a) realistic (b) innovative
	77%	*Q18	I prefer the idea of (a) certainty (b) theory
	71%	Q42	When I am doing long calculations (a) I tend to repeat all my steps and check my work carefully (b) I find checking my work tiresome and have to force myself to do it
	70%	Q30	When I have to perform a task, I prefer to (a) master one way of doing it (b) come up with new ways of doing it

FS DIMENSION	PREDICTION ACCURACY	ILS QUESTION	
Visual/Verbal	84%	*Q7	I prefer to get new information in (a) pictures, diagrams, graphs, or maps (b) written directions or verbal information
	84%	*Q11	In a book with lots of pictures and charts, I am likely to (a) look over the pictures and charts carefully (b) focus on the written text
	84%	*Q31	When someone is showing me data, I prefer (a) charts or graphs (b) text summarizing the results
	83%	*Q27	When I see a diagram or sketch in class, I am most likely to remember (a) the picture (b) what the instructor said about it
	78%	*Q3	When I think about what I did yesterday, I am most likely to get (a) a picture (b) words
	76%	*Q23	When I get directions to a new place, I prefer (a) a map (b) written instructions
	72%	Q19	I remember best (a) what I see (b) what I hear
	72%	Q43	I tend to picture places I have been (a) easily and fairly accurately (b) with difficulty and without much detail

3.4 Conclusion

The results showed that for the sample some questions in the ILS questionnaire are more accurate than others in predicting the overall learning style. If incorporating a limited number of ILS questions into a CITS to predict learning styles, it is best to select those questions that most accurately predict the overall learning style. It is recognised that all questions in the ILS are significant for assessing both the overall learning style and its strength. However, if replacing the completion of the ILS questionnaire with an implicit prediction of learning styles, the study results suggest a starting point of questions that best indicate overall learning style tendencies. The resulting subset of the best ILS predictor questions for each learning style dimension can now be considered when developing a strategy for predicting learning styles from a natural language dialogue.

4 Analysis of Learning Style Behaviour Traits

Human tutors use their knowledge of learning styles and behaviour in order to pick up cues from students about their learning preferences. Knowledge of the behaviour traits associated with learning styles is essential for predicting learning styles from a natural language dialogue. Felder and Silverman (1988) described dominant learner behaviours for each learning style in their model. Several ITS infer learning styles automatically from learner behaviour (Garcia et al., 2007; Cha et al., 2006; Wang et al., 2006), however not all of the behaviour characteristics used can be captured during a tutoring conversation. For example, student navigation and menu choices can be captured in an AHS but do not apply to a tutor-led conversation. This section describes a study undertaken to consider the implications of behaviour characteristics described in the FS model for predicting learning styles from a natural language dialogue.

4.1 Aim

The aim of the study was to extract from the FS model a list of behaviour characteristics for each FS dimension which could be used to predict learning styles from a natural language tutoring conversation.

4.2 Methodology

The FS model describes typical behaviour characteristics of each learning style. For clarity and ease of analysis, all behaviour information described in the FS model for each learning style was extracted and summarised in a table of common learner behaviour.

Next, each behaviour trait in the table of common learner behaviour was assessed using the following criteria:

1. Is it possible to map the behaviour trait onto a two-way online conversational tutorial?
2. How could the behaviour trait be used to implicitly predict learning styles?

Where it was found that a behaviour trait could be mapped onto a tutorial conversation and used to predict learning styles, the trait was included in a summary table along with a description of how it could be used to predict learning styles.

4.3 Results and Discussion

Table 5.3 shows a summary of common learner behaviour for each learning style. Where possible, the table contrasts related learning characteristics which distinguish the different dichotomies of each FS dimension.

Table 5.4 lists the subset of behaviour characteristics extracted from the FS model that can be mapped onto a two-way tutorial dialogue to predict learning styles. It was not possible to map all behaviour traits in Table 5.3 onto a two-way conversational tutorial, for example

- 'work well in groups' (Active) and 'work better alone' (Reflective) and
- 'strong in convergent thinking and analysis' (Sequential) and 'divergent thinking and synthesis' (Global)

  could not be assessed by a two-way conversation.

Table 5.3. Typical Learner Behaviour Characteristics extracted from the FS model

SENSOR	INTUITOR
Prefer facts, data, experimentation	Prefer principles and theories
Prefer solving problems using standard methods	Prefer innovation
Dislike surprises	Dislike repetition
Patient with detail	Bored by detail
Do not like complications	Welcome complications
Good at memorising facts	Good at grasping new concepts
Careful but slow	Quick but careless
Comfortable with symbols (eg. words)	Uncomfortable with symbols
VISUAL	VERBAL
Remember what they see	Remember what they hear, or what they hear then say
Like pictures, diagrams, flow charts, time lines, films	Like discussion
Prefer visual demonstration	Prefer verbal explanation
	Learn by explaining to others
ACTIVE	REFLECTIVE
Do something with information – discuss/explain/test	Examine and manipulate information introspectively
Active experimentation	Reflective observation
Do not learn much in passive situations (lectures)	Do not learn much if no chance to think (lectures)
Work well in groups	Work better alone
Experimentalists	Theoreticians
Process information by setting up an experiment to test an idea, or try out on a colleague	Process information by postulating explanations/interpretations, drawing analogies, formulating models
SEQUENTIAL	GLOBAL
Follow linear reasoning processes	Make intuitive leaps
Can work with material they have only partially or superficially understood	Difficulty working with material not understood
Strong in convergent thinking and analysis	Divergent thinking and synthesis
Learn best when information is presented in a steady progression of complexity and difficulty	Sometimes better to jump directly to more complex and difficult material

Table 5.4. Aspects of Learner Behaviour for Predicting Learning Styles from a Natural Language Tutorial Dialogue

BEHAVIOUR BY LEARNING STYLE	IMPLICATION FOR LEARNING STYLE PREDICTION
Sensor	
Prefer facts, data, experimentation	Perform better in questions with facts, examples and results
Dislike surprises	Prefer introductions, overviews and working in a sequential predictable order
Careful but slow	Consider timing interactions and number of errors
Comfortable with symbols (e.g. words)	Consider amount of discussion with the tutor
Intuitor	
Prefer principles and theories	Perform better in theory questions
Dislike repetition	Present information usually only once
Bored by detail	Perform better where information is summarised
Quick but careless	Consider timing interactions and number of errors
Uncomfortable with symbols	Consider amount of discussion with the tutor
Visual	
Remember what they see	Perform better in questions with diagrams, pictures, movies
Like pictures, diagrams, flow charts, time lines, films	Perform better in questions with pictures, diagrams, flow charts, time lines, films
Prefer visual demonstration	Perform better in questions with visual walkthroughs rather than textual explanation
Verbal	
Remember what they hear, or what they hear then say	Perform better in questions with movies and sound clips
Like discussion	Consider amount of discussion with the tutor
Prefer verbal explanation	Perform better in questions with movies, sound clips and tutor explanations
Learn by explaining to others	Consider amount of discussion with the tutor
Active	
Do something with information – discuss/explain/test	Consider amount of discussion with the tutor; perform better in questions with practical exercises
Experimentalists	Perform better in practical questions
Process information by setting up an experiment to test an idea, or try out on a colleague	Consider amount of discussion with the tutor; perform better in questions with practical exercises
Reflective	
Examine and manipulate information introspectively	Consider amount of discussion with the tutor
Theoreticians	Perform better in theoretical questions
Sequential	
Follow linear reasoning processes	Perform better when information presented in a steady progression of complexity and difficulty
Learn best when information is presented in a steady progression of complexity and difficulty	Perform better when information presented in a steady progression of complexity and difficulty
Global	
Sometimes better to jump directly to more complex and difficult material	Perform better where information is summarised and when they can attempt problems in one go.

4.4 Conclusion

The analysis of the FS model produced a list of behaviour traits (Table 5.4) for each FS dimension that could be used to predict learning styles from a two-way natural language tutoring conversation. Those behaviour traits not included in the final subset may be considered for inclusion in future work, for example by adding a group discussion facility.

5 Mapping of Behaviour Cues

The FS model describes typical behaviours for each different learning style. An examination of the ILS questionnaire (Appendix 1) and its best predictor questions (Table 5.2) illustrates that the questions have been designed to test the aspects of behaviour summarised in Table 5.3 and Table 5.4. When considering the extracted subset of FS learner characteristics for detecting learning style (Table 5.4) it became clear that each learning style could be categorised by a small number of behaviours that differentiated it from its opposite. For example, as seen in Table 5.4, Sequential learners are more successful when information is presented step-by-step in increasing difficulty, whereas Global learners learn more effectively when information is summarised and they can attempt difficult problems straight away. By applying this behaviour to a tutorial conversation, e.g. by incorporating questions that allow learners to choose to attempt a solution immediately or be guided through the steps of a solution, it may be possible to predict the learning style on the Sequential/Global dimension.

In order to map the behaviour characteristics in Table 5.4 to a tutoring conversation to predict learning styles, it was first necessary to decide which aspects of behaviour need to be captured. Each aspect of behaviour in Table 5.4 was studied in turn and the list was reorganised according to behaviour, with similar behaviours grouped together. For example, as both Verbal and Active learners like discussion, they were grouped together under the 'like discussion' behaviour category. Next, this list of behaviours was reduced further by considering the behaviour that would need to be captured from a natural language conversation. For example, the 'like discussion' category now became the 'discussion' category and included also the Sensor (like discussion), Intuitor (do not like discussion) and Reflective (do not like discussion) learning styles. The result of this analysis was a list of behaviour cues to be captured during the conversational tutorial which could be used to predict learning style. Table 5.5 lists the behaviour to be captured during a tutorial conversation in order to predict learning styles, and relates each behaviour variable to the learning styles it may be used to predict.

Table 5.5. Learner Behaviour Cues to be Captured During Tutoring

BEHAVIOUR VARIABLE TO BE CAPTURED	LEARNING STYLE
Number of discourse interactions	Sensor, Intuitor, Verbal, Active, Reflective
Number of questions asked	Sensor, Intuitor, Verbal, Active, Reflective
Tutorial duration	Sensor, Intuitor,
Reading time	Sensor, Intuitor,Visual, Verbal
Number of errors due to not reading the question	Sensor, Intuitor
Right answer after seeing an image	Visual
Right answer after seeing a movie/walkthrough	Visual, Verbal, Active
Right answer after an explanation of theory	Intuitor
Right answer after seeing an example	Sensor
Choose to be guided through the steps of solving a problem	Sensor, Sequential
Choose to solve a problem straight away	Intuitor, Global
Score for practical questions	Active, Sensor
Score for theoretical questions	Reflective, Intuitor

6 Use of Language

Mairesse et al. (2007) found that it was possible to determine an individual's personality type by looking at the type of vocabulary they used. As learning style is linked to personality (see Chapter 2), it may be possible that the type of vocabulary used can indicate an individual's learning style. Özpolat and Akar (2009) mapped a short list of key words to FS learning styles, and analysed student search terms to successfully predict learning styles for three of the four FS dimensions.

It was decided that a list of key words and phrases would be drawn up for each learning style so that the learner's discourse could be examined to see if learning style could be inferred by the use of these words. For example, the key word *show* (e.g. "Can you show me an example") may indicate a Visual learning style, whereas the keyword *tell* (e.g. "Can you tell me what to do") may indicate a Verbal learning style.

A starting point for drawing up a list of key words was the FS model, which used a number of particular words in its description of the dominant behaviour preferences for each learning style. The key words list in Ozpolat and Akar (2009) only produced results for Visual learners, so it was extended by analysing the FS model and extracting a list of indicative words and phrases mapped to each learning style. This initial key words list was then expanded using a thesaurus to produce an initial set of key words and phrases that were indicative of learning style (Table 5.6).

The process of discovering associations between key words and particular learning styles requires experimentation and analysis of tutoring dialogues, so the content of the list will be tested and expanded by analysing actual tutoring discourse once the Oscar CITS has been developed for a particular domain.

Table 5.6. Key Words and Phrases which may be Indicative of Learning Style

LEARNING STYLE	KEY WORDS/PHRASES
Sensory	detail, overview, outline, review, introduction, basic principles, essentials, intro, sequential, consecutive, continuous, persistent, regular, serial, steady, subsequent, successive, example, case, practical, exemplar, illustration, lesson, prototype, sample, specimen, stereotype, real world
Intuitive	summary, summarise, breakdown, brief, condense(d), succinct, digest, rundown, overview, outline, review, principle, basis, fundamental, theory, theoretical, approach, argument, assumption, basis, concept, foundation, idea, in principle, ideology, outlook, philosophy, position, premise, proposal, rationale, supposition, theorem
Visual	see, illustrate, inspect, look, observe, scan, view, watch, picture, depiction, drawing, illustration, image, impression, portrayal, representation, diagram, chart, graph, figure, layout, table, show, display, demonstrate, present(ation), simulation, video
Verbal	discuss, confer, consider, consult with, debate, discourse, go into, review, explain, analyse, break down, clarify, clear up, define, describe, get across, go into detail, spell out, tell, advise, declare, divulge, express, fill in, impart, inform, instruct, let know, mention, report, say, speak, state, forum
Active	practical, realistic, applied, empirical, experimental, working, exercise, interactive, action, activity, drill, examination, operation, problem, study, task, test, discuss, confer, consider, consult with, debate, discourse, go into, review, explain, analyse, break down, clarify, clear up, define, describe, get across, go into detail, spell out, example, case, exemplar, illustration, lesson, prototype, sample, specimen, stereotype
Reflective	principle, basis, fundamental, theory, approach, argument, assumption, basis, concept, foundation, idea, ideology, outlook, philosophy, position, premise, proposal, rationale, supposition, theorem
Sequential	example, case, exemplar, illustration, lesson, prototype, sample, specimen, stereotype, in steps, sequential, consecutive, continuous, persistent, regular, serial, steady, subsequent, successive, subsections, parts, detail
Global	summary, summarise, breakdown, brief, condense(d), succinct, digest, rundown, overview, outline, review, overall, whole

7 Logic Rules

In order for a CITS to dynamically predict learning styles it will be necessary to convert the captured knowledge of FS behaviour factors and keywords into logic rules. The set of logic rules will continually increment student learning style values as the tutoring conversation takes place.

The analysis of the FS model described in this chapter produced a list of learner behaviour cues to be captured during tutoring and linked to associated FS learning styles (Table 5.5). This knowledge of behaviour events must be converted into logic rules. Some of the behaviour cues can be directly converted to logic rules, for example:

```
IF    learner shown image/diagram
AND   learner gives correct answer
THEN  increase VISUAL;
```

This example logic rule is generated from the behaviour cue ‘Right answer after seeing an image’, linked to the Visual learning style (Table 5.5). If a learner does not know the answer, is shown an image and then gets the answer right, this visual presentation has helped their understanding so the Visual learning style value should be incremented.

Other behaviour cues that rely on comparisons rather than the occurrence of events can also be converted into logic rules, for example:

```
IF    learner_interaction_count > average_interaction_count
THEN  increase SENSOR
      AND increase VERBAL
      AND increase ACTIVE;

IF    learner_interaction_count < average_interaction_count
THEN  increase INTUITOR
      AND increase REFLECTIVE;
```

These example logic rules are generated from the behaviour cue ‘Number of discourse interactions’, linked to the Sensor, Intuitor, Verbal, Active and Reflective learning styles (Table 5.5). If, during a set number of questions, a learner’s conversation has an above average number of interactions, then it can be inferred that the learner enjoys discussion and therefore the Sensor, Verbal and Active learning style values should be incremented. If, however, a learner’s conversation has a below average number of interactions, this indicates they do not enjoy discussion and so the Intuitor and Reflective learning style values should be incremented.

Additionally, the knowledge of key words and phrases linked to learning styles can be converted into logic rules, for example:

```
IF learner-word IN {see, show, picture, diagram}
THEN increase VISUAL;
```

This example logic rule was generated using some of the key words linked to the Visual learning style (Table 5.6). (Note that the word set linked to Visual learners has been reduced in this example for clarity, but would actually contain all associated key words and phrases.) If a learner utterance contains a word in the key word set, the Visual learning style value should be incremented.

The full set of 29 logic rules devised for the FS model are given in Appendix 2. As the set of logic rules is based on learner behaviour and language during a conversational tutorial, the logic rules may be mapped to other learning styles models which associate the behaviour with different groups of learners. For example, the logic rules for the Visual FS learning style shown above may be adapted to match the Visual-Spatial intelligence from Gardner's Theory of Multiple Intelligences (1983), as follows:

```
IF    learner shown image/diagram
AND   learner gives correct answer
THEN  increase VISUAL-SPATIAL;

IF learner-word IN {see, show, picture, diagram}
THEN increase VISUAL-SPATIAL;
```

Similarly, the logic rules related to the 'Number of discourse interactions' behaviour may be applied to the Learning Styles Questionnaire (Honey and Mumford, 2006) Activist and Reflector learning styles, as follows:

```
IF    learner_interaction_count > average_interaction_count
THEN  increase ACTIVIST;

IF    learner_interaction_count < average_interaction_count
THEN  increase REFLECTOR;
```

Therefore, the set of logic rules produced is generic as it can be used to predict learning styles for other models. It is anticipated that the set of generic logic rules will be expanded to include additional learner behaviour as other learning styles models are adopted.

8 Conclusion

Natural language interfaces are intuitive for humans, as we use conversation to communicate in real life. A more intuitive and familiar interface for a tutoring system can help learners feel more confident, which can improve their motivation and the effectiveness of their learning. Conversational interfaces can also adopt the constructivist style of tutoring used by human tutors, and thought to aid a deeper understanding of a subject. Implicitly predicting learning style during a tutoring conversation removes the requirement for learners to complete onerous and time-consuming questionnaires. By continually predicting learning style during tutoring, any changes to learning style or differences in learning style for different subjects can be automatically picked up and adapted to.

This chapter has outlined the methodology and strategies for predicting learning styles implicitly from a two-way tutoring conversation with a CITS. In order to demonstrate the methodology for developing prediction strategies, an example learning styles model was adopted – the Felder-Silverman (FS) model. Knowledge of learning styles was extracted from the FS model and applied to a natural language tutoring dialogue. The first analysis considered the FS model assessment questionnaire, the ILS. A study was undertaken whose results show that some questions are more accurate than others in predicting the overall learning style, and so the number of questions incorporated into a tutorial can be reduced. 17 of the 44 questions were identified as having 75% or better accuracy in predicting the overall learning style, with the three best predictor questions predicting the overall learning style 84% of the time. Next, an analysis of the FS model was presented, with a summary of behaviour linked to learning styles from which a set of behaviour characteristics were extrapolated which could be used to predict learning styles. This subset was further analysed to produce a list of behaviour cues that could be captured from a tutoring dialogue and utilised to predict learning styles. Finally, the use of language during a tutorial conversation was considered, and a list of key words and phrases that could be indicative of learning style drawn up.

After analysing the FS model, the knowledge captured about learner behaviour and language was converted into a set of logic rules which increment learning style values following events during a tutoring conversation. The set of logic rules is generic as it is based on learner behaviour, which can be used to predict learning styles for other models.

The original methodology described in this chapter is generic, and so can be applied to other learning styles models to extract the knowledge required to predict learning styles from a natural language dialogue.

9 Chapter Highlights

- An original, generic methodology for automatically predicting learning styles from a two-way tutoring conversation with a CITS was described.
- An empirically derived subset of the best predictor ILS questions was presented.
- An analysis of learning style behaviour traits in the FS model, and their implication for predicting learning styles from a natural language dialogue was described.
- A set of learner behaviour cues to predict learning styles from a natural language tutoring dialogue was created.
- A list of key words and phrases which may be indicative of learning style was produced.
- Behaviour cues for implicitly predicting learning styles from a tutoring dialogue were encapsulated into logic rules.

Chapter 6 A Methodology and Architecture for Developing a CITS to Predict Learning Styles

1 Introduction

Computerised learning systems offer flexible learning at any time or place, via the Internet, at a fixed cost. Computerised learning systems allow students to learn at their own pace, and as well as being adopted for distance learning, they can support classroom courses. As described in Chapter 4, Intelligent Tutoring Systems (ITS) are computerised learning systems that adopt artificial intelligence (AI) techniques to personalise online tutoring, adding intelligence to make a learning experience more than just a computerised book. ITS most often personalise tutoring by adapting the learning content according to individual factors such as existing knowledge and learning style. Chapter 2 described how learning styles are important during the process of learning as they affect how a person receives and processes information. Human tutors try to include different styles of learning material and introduce different methods (e.g. tutorials, labs, lectures) to open up learning to all students. ITS that incorporate learning styles can improve the effectiveness of a learning experience (Walters et al., 2000; Paredes and Rodriguez, 2004). Conversational Intelligent Tutoring System (CITS) use natural language dialogue to conduct tutoring, and although conversation is a more human-like way of interacting than a menu, the complexities of developing conversational agent (CA) interfaces (see Chapter 3) means that they are not commonly adopted by ITS. There are currently no Conversational Intelligent Tutoring Systems (CITS) that mimic a human tutor by dynamically predicting and adapting to learning styles whilst directing a tutoring conversation.

Chapter 5 described a methodology for extracting knowledge from a learning styles model which could be used to predict learning styles from a natural language tutoring dialogue. However, a learning styles knowledge base is only one component required in a Conversational Intelligent Tutoring System (CITS) that can dynamically predict learning styles. Also required is the tutoring subject knowledge, and components with the ability to manage and conduct a tutoring conversation.

This chapter introduces a novel CITS called Oscar CITS which, during a tutoring conversation, can dynamically predict an individual's learning style and adapt its tutoring style accordingly. Throughout this thesis, the term *Oscar CITS* refers to the overall system that both predicts and adapts to learning styles. However, due to the complexity of the design, development and experimental analysis involved, the functions of prediction and adaptation have been separated. In this thesis, the research relating to the automatic prediction of learning styles during a tutoring conversation will be known as the *Oscar Predictive CITS (PCITS)*. The research relating to the adaptation of a tutoring conversation to an individual's learning styles will be known as the *Oscar Adaptive CITS (ACITS).*

In this chapter, section 2 gives an overview of the Oscar CITS. Next, an original methodology is proposed for developing Oscar Predictive CITS, a CITS that can automatically predict learning styles during a tutoring conversation. The methodology is independent of the learning styles model and tutoring domain as the PCITS construction is based around capturing learner behaviours during tutoring. A generic architecture for Oscar PCITS is proposed, which is modular for ease of development and maintenance and to integrate flexibility in the choice of learning styles model and tutoring domain.

2 Oscar Conversational Intelligent Tutoring System

The Oscar CITS is a novel conversational intelligent tutoring system that dynamically predicts a student's learning style during a tutoring conversation, and adapts its tutoring style to suit the preferred learning style. Oscar's pedagogical aim is to provide the learner with the most appropriate learning material for their learning style leading to a more effective learning experience and a deeper understanding of the topic. Rather than being designed with the purpose of picking up learning styles (such as Cha et al., 2006) the Oscar CITS attempts to mimic a human tutor by leading a two-way discussion and using cues from the student dialogue and behaviour to predict and adapt to their learning style. Oscar's natural language interface and classroom tutorial style are intuitive to learners, enabling them to draw on experience of face-to-face tutoring to feel more comfortable and confident in using the CITS. Oscar CITS is a personal tutor that can answer questions, provide hints and assistance using natural dialogue, and which favours learning material to suit each individual's learning style. The Oscar CITS offers 24-hour personalised learning support at a fixed cost.

As described in Chapter 4, Brusilovsky and Peylo (2003) identified three main strategies to including intelligence in Intelligent Tutoring Systems (ITS): curriculum sequencing, intelligent solution analysis and problem solving support. The Oscar CITS will combine all three intelligent technologies with a conversational interface, aiming to build the confidence of the learner and improve motivation and deep understanding of the subject. Oscar's intelligent approach includes presenting learning material in the sequence and style most suited to the individual's knowledge and learning style (curriculum sequencing), analysing and giving feedback on incomplete and erroneous solutions (intelligent solution analysis) and giving intelligent hints and discussing questions (problem solving support).

In summary, the main features of the innovative Oscar CITS are:

- Oscar mimics a human tutor by adopting a tutor-led, conversational approach.
- Oscar's natural language interface is intuitive to use and enables learners to actively discuss problems and solutions.
- Like human tutors, Oscar supports constructivist learning by incorporating problem solving support and intelligent solution analysis techniques.
- Oscar implicitly predicts an individual's learning style by capturing and modelling learner behaviour during a tutoring conversation.
- Oscar aims to improve the learning experience by intelligently adapting its tutoring style to match individual's learning styles.
- Oscar CITS is generic, allowing the free choice of learning styles model and subject domain.

The Oscar Predictive CITS (PCITS) is a CITS that incorporates the automatic prediction of learning styles during a tutoring conversation. The construction of Oscar PCITS will now be described. First, an original generic methodology for developing Oscar PCITS will be proposed, followed by a generic architecture for constructing Oscar PCITS.

3 A Generic Methodology for Creating Oscar PCITS

CITS are complex and time-consuming to develop, requiring expertise in knowledge engineering (the capture and formatting of expert knowledge (O'Shea et al., 2011), i.e. tutoring, learning styles, domain knowledge) and CA scripting.

Formalising the development of a CITS which can be applied to different learning styles models and tutoring domains will help to speed up the development.

The proposed generic methodology for creating an Oscar PCITS consists of three phases as shown in Table 6.1. The first phase of the methodology relates to the creation of the learning styles predictor module and the second phase to the tutorial subject domain. The third phase incorporates the learning styles predictor and tutorial conversation into a PCITS architecture. Each phase will now be described.

Table 6.1. 3-Phase Methodology for Creating Oscar PCITS.

Phase 1: Create the Learning Styles Predictor Module

1.1. Select a Learning Styles Model
    a. Reduce the learning styles model if necessary
    b. Extract the behaviour characteristics
1.2. Map learning style behaviour to the conversational tutoring style
1.3. Analyse the learning styles model for language traits
1.4. Adapt the generic logic rules to predict learning styles

Phase 2: Design a Tutorial Conversation

2.1. Capture the tutorial scenario and questions (including movies, voice, images, examples, etc.) from human tutors in a specific domain
2.2. Determine the conversational structure/style
2.3. Map tutorial questions onto the generic question styles and templates
2.4. Script Conversational Agent natural language dialogue for each tutorial question using the 3-level model
2.5. Link tutorial dialogue to logic rules through Conversational Agent variables

Phase 3: Construct the PCITS Architecture

3.1 Methodology Phase 1: Create the Learning Styles Predictor Module

Phase 1 of the Oscar PCITS Methodology involves the analysis of a learning styles model in order to create a Learning Styles Predictor module for the PCITS. The steps in this phase have been devised following the development of strategies for predicting learning styles from natural language described in Chapter 5.

3.1.1 Step 1.1: Select a Learning Styles Model

The first step in creating the learning styles predictor module requires the selection of a learning styles model (Felder and Silverman, 1988; Honey and Mumford, 1992). To illustrate Phase 1 of the methodology, the Felder-Silverman (FS) model (Felder and Silverman, 1988) was selected as the initial experimental group will be university engineering students (see Chapter 5, section 2).

Next, if the chosen learning styles model or its assessment tool is large, it may be possible to reduce the model to allow its implicit inclusion into a tutoring conversation. As described in Chapter 5, section 3, a study was undertaken to reduce the FS model by investigating the FS model's evaluation tool, the Index of Learning Styles (ILS) (Felder and Soloman, 1997). The experiment found that 17 questions predicted the overall learning style result in at least 75% of cases, with the top three questions predicting the result in 84% of cases (Table 5.2). The resulting subset of the best ILS predictor questions formed the basis of learning styles prediction by the Oscar PCITS.

The final part of Step 1.1 is to extract from the model the typical learner behaviour characteristics for each learning style. As described In Chapter 5, section 4, for clarity and ease of analysis the behaviour characteristics were extracted from the FS model and summarised in a table of common learner behaviour (Chapter 5, Table 5.3).

3.1.2 Step 1.2: Map Learning Style Behaviour to the Conversational Tutoring Style

To map learning style behaviour to the conversational tutoring style, each behaviour characteristic extracted in step 1.1b is assessed using the following criteria:

1. Is it possible to map the behaviour trait onto a two-way online conversational tutorial?
2. How could the behaviour trait be used to implicitly predict learning styles?

All behaviour traits that can be mapped onto a tutorial conversation and used to predict learning styles should be included in a summary table along with a description of how they could be used to predict learning styles. In Chapter 5, section 4 this subset of behaviour traits for the FS model is shown in Table 5.4.

Next, it is necessary to decide which aspects of behaviour need to be captured during a tutoring conversation. Each behaviour trait in the summary table (Table 5.4) needs to be studied in turn and the list reorganised according to behaviour, with similar behaviours grouped together. For example, as both Verbal and Active learners like discussion, they should be grouped together under the 'like discussion' behaviour category. Next, this list of behaviours should be reduced further by considering which behaviour needs to be captured from a natural language conversation. For example, the 'like discussion' category would now become the 'discussion' category and additionally include the Sensor (like discussion), Intuitor (do not like discussion) and Reflective (do not like discussion) learning styles. The result of this analysis is a list of behaviour cues to be captured during a conversational tutorial which can be used to predict learning style. In Chapter 5, Table 5.5 lists the behaviour to be captured during a tutorial conversation to predict FS learning styles.

3.1.3 Step 1.3: Analyse the Learning Styles Model for Language Traits

As described in Chapter 2, the vocabulary used in natural language dialogue was found to indicate personality type (Mairesse et al., 2007) and learning style (Ozpolat and Akar, 2009). Step 1.3 of the methodology involves analysing the learning styles model to extract any language traits that could be indicative of learning style. In Chapter 5 section 6, indicative words and phrases used to describe behaviour traits were extracted from the FS model and mapped to learning styles. This key words list should then be expanded using a thesaurus to produce an initial set of key words and phrases indicative of learning style (e.g. as seen in Table 5.6).

The process of discovering associations between key words and particular learning styles requires experimentation and analysis of tutoring dialogues, so the content of the list must be tested and expanded by analysing actual tutoring discourse once the Oscar PCITS has been developed for a particular domain (see Chapters 7 and 8).

3.1.4 Step 1.4: Adapt the Generic Logic Rules to Predict Learning Styles

The final step in phase 1 of the methodology is to convert the knowledge of the learning styles model (the captured behaviour and language traits gathered from steps 1.2 and 1.3) into a set of logic rules. The aim of the logic rules is to continually increment student learning style values following events during the tutoring conversation. The analysis of the FS model described in Chapter 5 produced a set of 29 logic rules based on learner behaviour and language to be captured (Appendix 2). As this set of logic rules relates to learner behaviour during a tutoring conversation, the rules are generic and can be adapted for different learning styles models (as demonstrated in Chapter 5, section 7). For different learning styles models, the set of generic logic rules should be adapted, and extended if required to include additional learner behaviour defined by the chosen model. Table 6.2 shows two examples of logic rules developed using the behaviour cues from step 2.2 (Chapter 5, Table 5.5) and mapped to the FS model. The first example, rule 1, is generated from the behaviour cue ‘Right answer after seeing an image’ and is linked to the Visual learning style. If a student does not know the answer, is shown an image and then gets the answer right, this visual presentation has helped their understanding so the Visual learning style value is incremented. Rule 2 is generated from the 'Number of errors due to not reading the question' behaviour, linked to the Intuitor and Visual learning styles. If the answer to a question is given in the explanation text and a student gets the answer wrong, this behaviour indicates they are careless and not comfortable with reading text, so the Intuitor and Visual learning style values are incremented.

Table 6.2. Example Logic Rules to Adjust Student Learning Style Values Based on Tutoring Conversation.

1. Example rule to test whether presenting information visually helps the student's information perception:

   IF student shown image/diagram
   AND student gives correct answer
   THEN increase VISUAL;

2. Example rule to test how comfortable the student is with words and with detail:

   IF answer is given in the explanation text
   AND student does not know the answer
   THEN increase INTUITOR
   AND increase VISUAL;

The full list of 29 logic rules developed for the FS model is given in Appendix 2. The logic rules will be applied during a tutoring conversation to dynamically predict learning styles. Some of the behaviour cues listed in Chapter 5, Table 5.5, such as duration and number of interactions, will be assessed at set times during a tutoring conversation rather than continually throughout the tutoring. The set of logic rules resulting from this step can be applied during a tutoring conversation to dynamically predict learning styles.

This section has described the steps in phase 1 of the generic Oscar PCITS methodology to create a Learning Styles Predictor module using the FS model as an example.

3.2 Methodology Phase 2: Design a Tutorial Conversation

Phase 2 of the methodology involves capturing the tutorial from expert human tutors and iteratively developing a tutorial conversation with input from the human tutors.

3.2.1 Step 2.1: Capture the Tutorial Scenario and Questions from Human Tutors in a Specific Domain

The first step in designing a tutorial conversation is to capture a tutorial scenario from human tutors. After deciding on the subject domain for the tutorial, interviews should be conducted with expert human tutors in the chosen domain to identify important concepts for the tutorial syllabus. Next, a number of tutorial questions and a multiple choice question (MCQ) test need to be devised which cover the agreed learning outcomes of the tutorial. To capture the tutorial scenario, a document (known as the '*tutorial conversation blueprint*') should be produced in consultation with tutors which contains a conversation script for each question, including a set of possible learner answers and tutor's responses to these. For each following learner response, a further tutor response should be written, and so on, until each question reaches a conclusion. The resulting tutorial conversation blueprint contains a number of different learning paths for each tutorial question, depending on individual learner knowledge and responses. Resources such as examples, movies, images etc. should be embedded into the tutorial conversation blueprint as appropriate. The MCQ test will be used to test learners' knowledge and understanding at the end of the tutoring session. Capturing human tutor expert knowledge when designing the tutorial conversation is an iterative process, however by planning and detailing the dialogue at this point, the development of the conversational agent will be more efficient.

3.2.2 Step 2.2: Determine the Conversational Structure/Style

A CITS that attempts to mimic a human tutor must be able to manage a tutoring conversation on a number of levels, each with a different goal. Step 2.2 of the methodology determines the structure of the CA tutorial conversation. Drawing on experience of classroom tutorials (Department for Education and Skills, 2004a, 2004b), three parts of a tutorial conversation with separate goals were distinguished and a three-level model of a tutorial conversation was designed for use in the PCITS (Figure 6.1). At the highest level (the 'social level'), Oscar PCITS needs the ability to maintain a natural language tutorial conversation, and like a human tutor must pick up cues if the learner is not engaging in the tutorial (e.g. use of bad language) and choose to end the session. At the main 'tutoring level', Oscar PCITS directs the tutorial, explains topics and asks questions, guiding the learner towards an understanding of the topic. This may involve Oscar PCITS giving feedback on erroneous or incomplete solutions (intelligent solution analysis), explaining the topic using different methods if required, such as practical examples (curriculum sequencing) and giving hints to help the learner construct a solution (problem solving support). During a tutorial, learners may wish to discuss a related topic to help their understanding, requiring a deeper ‘discussion level’ with the ability to discuss and explain a predefined set of Frequently Asked Questions related to the domain.

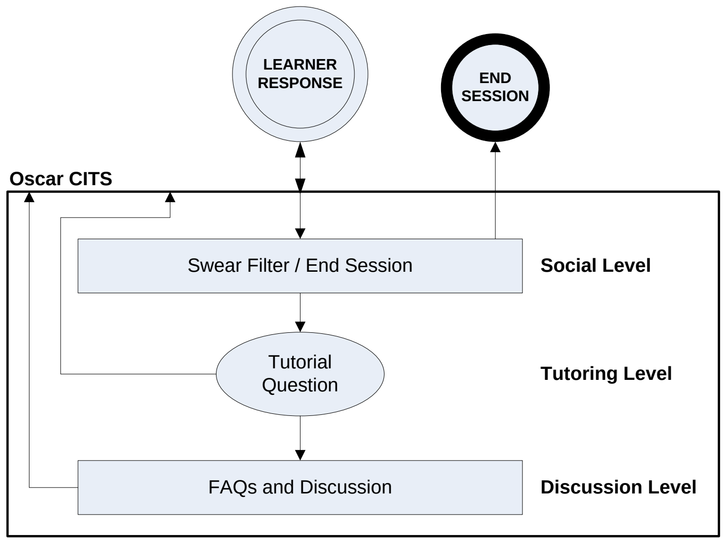


Figure 6.1. 3-Level Model of a Tutorial Conversation.

The implication of this structure of a natural language tutorial is that it is necessary to develop mechanisms and conversations that can work at all three levels when applying the Oscar PCITS to a learning styles model and subject domain. As part of this step, a list of FAQs and answers should be captured from the human tutors, scripted in natural language and added to the tutorial conversation blueprint.

3.2.3 Step 2.3: Map Tutorial Questions onto the Generic Question Styles and Templates

The third step in phase 2 of the methodology links the captured tutorial questions to the behaviour characteristics identified in phase 1 step 1.2. This is done by mapping tutorial questions to a set of generic question styles and templates.

During the development of the Learning Styles Predictor module (Phase 1 steps 1.1 and 1.2), questions and behaviour from the FS model were mapped to a conversational tutoring style. To gather the behavioural information identified in phase 1 as indicative of learning style, a set of generic styles and templates of tutorial questions was developed. The templates will speed up the development of Oscar PCITS tutorials for different domains whilst still offering the flexibility to include different styles of question as the domain dictates. The goal is to develop a toolkit by expanding the set of question styles and templates over time as the Oscar PCITS is applied to different learning styles models and domains requiring different approaches to tutoring.

Four generic styles of tutorial question were devised, as follows:

- *Practical Style:* practical problems and exercises that test the application of knowledge. These will indicate whether a learner performs better in practical questions.
- *Theoretical Style:* theoretical questions to test understanding and knowledge, which will indicate whether a learner performs better in theoretical topics.
- *Process Style:* questions where learners can ask to be guided through a process or choose to attempt to solve the question all at once. This will indicate a learner's preferred approach to tackling complex practical questions.
- *Trick Question Style:* 'trick questions' where part of the answer is given in the explanatory text, which will test the learner's attention to detail and reading skills.

Further to the generic styles of question described above, two generic question templates were represented diagrammatically to show the flow of the tutorial conversation and the use of different hints and explanations. Figure 6.2 shows a generic question template that could be applied to both practical and theoretical question styles. The generic question template is designed to offer hints to students about the question answers and also to gather information about the type of help which is most effective (which may indicate their learning style). In Figure 6.2, the question is asked in box 1 and if the learner responds with the correct answer at any point, they are given feedback and taken to the next question (response 2). If the learner does not know the answer or their answer is wrong, Oscar explains the concept and repeats the question (response 3). If the learner still does not know the answer or their answer is wrong, Oscar shows an example and repeats the question (response 4). If the learner still does not know the answer or their answer is wrong, Oscar shows a movie clip and asks the question again (response 5). If the learner is still unable to answer correctly, Oscar shows an image or diagram to explain the concept and repeats the question (response 6). Finally, if the learner still does not know the correct answer, Oscar tells them the answer, suggesting that they revise the topic (showing additional resource links), then asks if they wish to continue with the tutorial (response 7). If the learner wishes to continue, they are taken to the next question; if not the tutorial is ended.

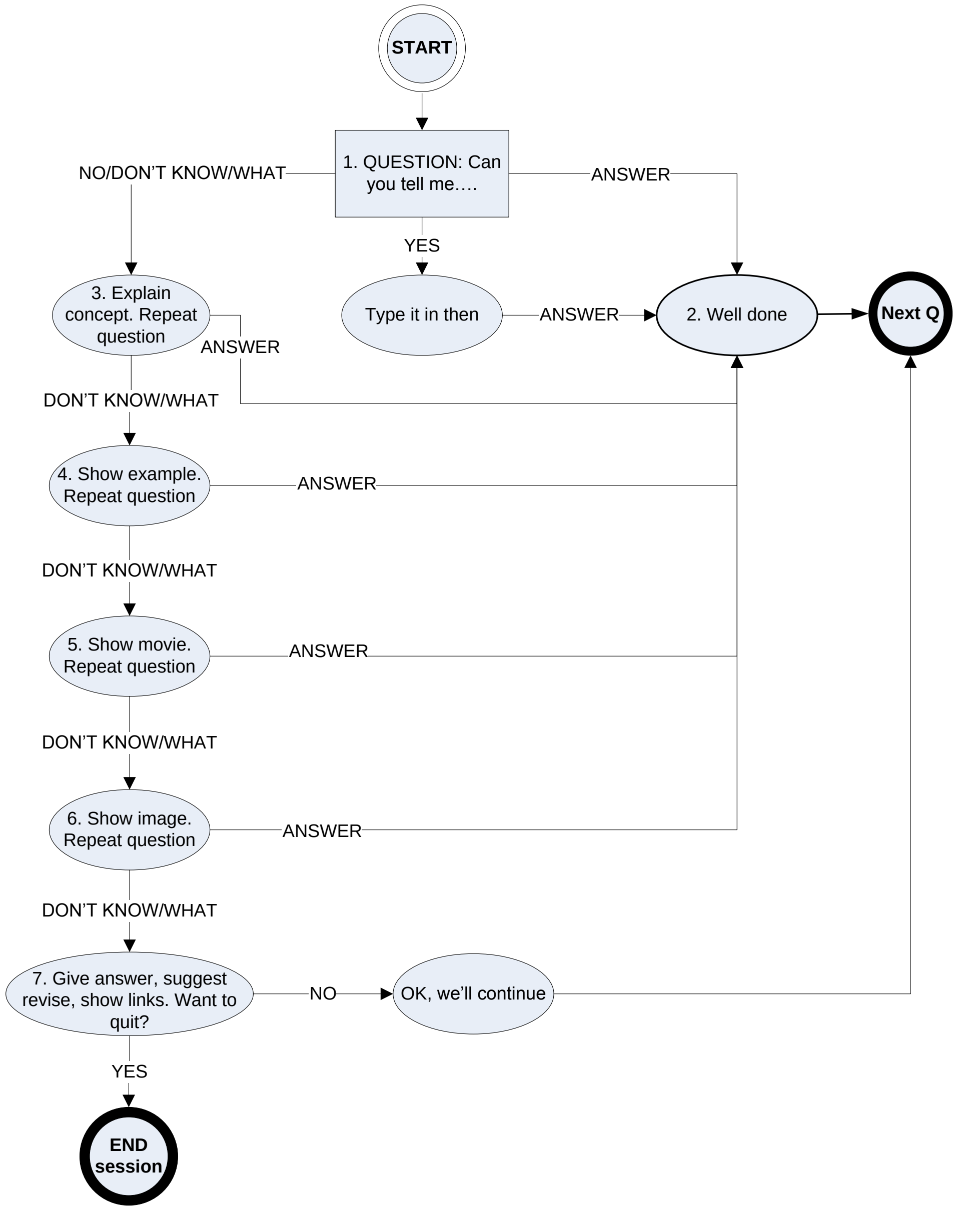


**Figure 6.2.	Generic Question Template with Hints**

The second generic question template, in Figure 6.3, adopts the *Process* generic question style, where a learner has a choice of approach to performing a complex task or following a process. In box 1 the question is asked, and the learner is asked whether they wish to be guided through the process or to attempt the solution in one step. The example process in the template in Figure 6.3 has three steps, but the template can be extended to fit any number of steps as required. If the learner wants to attempt the task in one go, they are asked to solve the problem (response 2) straight away. Following this selection, if the solution is correct, the learner is given feedback and taken to the next tutorial question. If there are errors or omissions, the learner is given feedback on the errors and given two more chances to solve the problem. If there are still errors or omissions after the third attempt (response 3), the learner is provided with step by step guidance, starting at the process step containing the problem. However, if after box 1 the learner asks to be guided through a process in steps, the Oscar PCITS gives an explanation of the first step and asks for a partial answer (response 4). If the learner gives a correct answer, they are given guidance for the next steps in turn. However, if the learner does not know the answer or gives an incorrect answer, they are shown extra resource(s) (e.g. a movie, example or image) and asked for the answer to step 1 again (response 5). If the learner still does not know or gives an incorrect answer, they are given the answer and moved on to the next step (response 6). This process is repeated for all steps until the learner completes the answer or has been given the answer for each step, when they are taken to the next tutorial question.

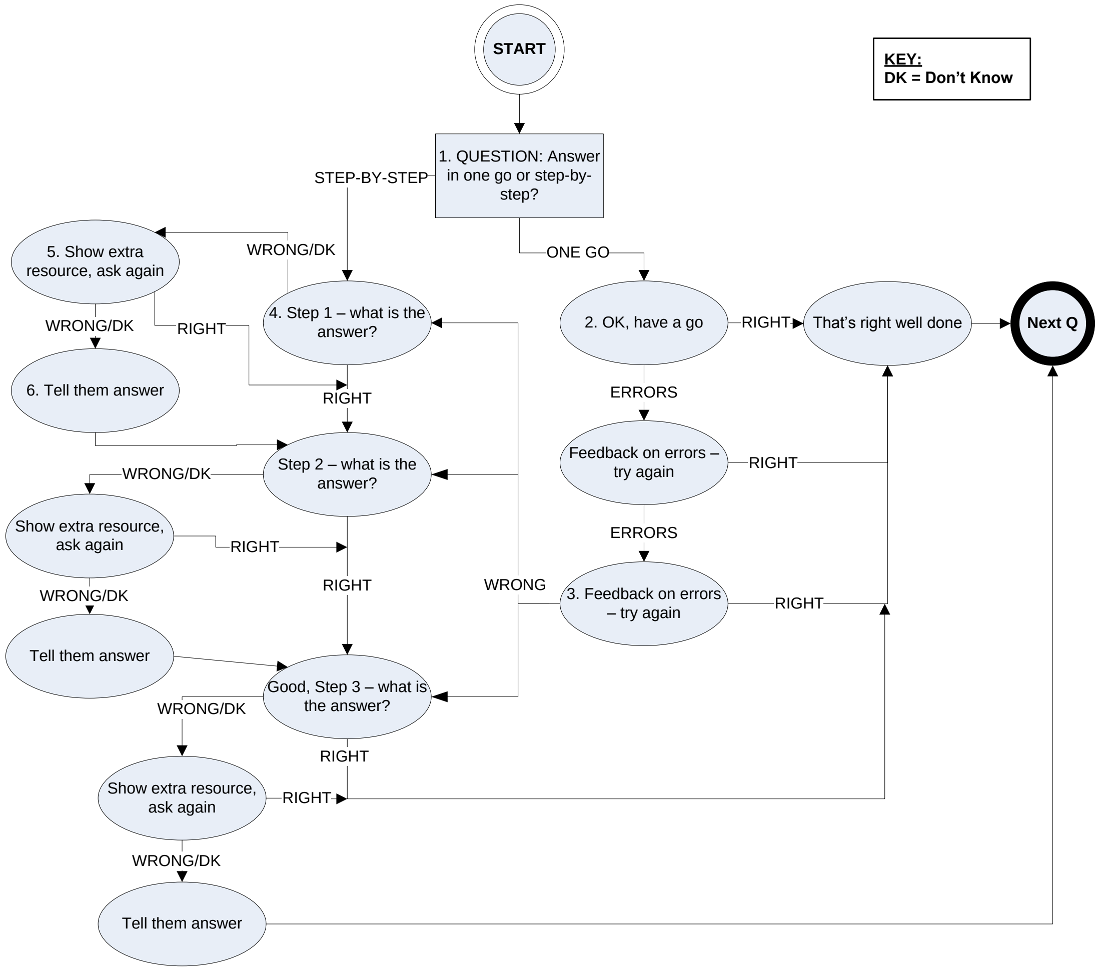


Figure 6.3. **Generic Question Template with Choice of Approach**

The generic templates and styles of tutorial question described in this section are intended as a proof of concept, and will be extended as requirements arise for different kinds of tutoring appropriate to different domains.

In this step of the methodology, tutorial questions are mapped onto the generic styles and templates (with extra resources included as required) and any resulting changes in dialogue are recorded in the tutorial conversation blueprint.

3.2.4 Step 2.4: Script Conversational Agent Natural Language Dialogue for each Tutorial Question using the 3-Level Model

Step 2.4 of the methodology involves creating Conversational Agent (CA) scripts to conduct the tutoring dialogue defined in steps 2.1, 2.2 and 2.3 (and recorded in the tutorial conversation blueprint). First, it is necessary to adopt a text-based CA that can capture and receive information using variables. Variables are used by the CA to collect information about behaviour and knowledge during the conversation and to receive information about learning styles and knowledge for adapting the conversation. As described in Chapter 2, there are different approaches to designing text-based CAs, but CAs will normally define an appropriate scripting language for developing conversation scripts. Text-based CA scripts are sometimes organised into modules called contexts that manage a particular part of a conversation (Sammut, 2001) and simplify the scripting of lengthy dialogues. The structure of the CA scripts is dependent on the CA and scripting language selected.

Before the scripting of the tutorial conversation can begin it is necessary to organise the CA scripts by applying the 3-level tutorial conversation model (Figure 6.1) as described in the tutorial conversation blueprint. Once the structure of the scripts has been defined, the CA dialogue needs to be scripted for each tutorial question across each level of the 3-level tutorial conversation model.

3.2.5 Step 2.5: Link Tutorial Dialogue to Logic Rules through Conversational Agent Variables

The final step in phase 2 of the methodology links the behaviour captured by the tutorial conversation to the set of logic rules (produced in phase 1) that predict learning styles.

1. First include the related learning styles in the tutorial conversation blueprint, which is the human-readable record of the tutoring conversation. Moving through the tutorial conversation blueprint, for each learner behaviour found, annotate the

document with the learning style defined in the associated logic rule. The logic rules from Phase 1 step 1.4 specify which learning styles are to be incremented when particular events occur (e.g. for the FS model, incrementing the Sensory learning style value after an example is shown).

2. Next, the CA scripts must be updated to capture the behaviour by setting variable values when particular rules fire.
3. Now that the tutorial conversation has been fully scripted for a CA it must be tested and verified by expert human tutors.

This section has described the steps of the generic methodology to design a tutoring conversation for a specific domain.

3.3 Methodology Phase 3: Construct the PCITS Architecture

Once the learning styles predictor module and the tutorial conversation have been designed, it is necessary to incorporate them into a PCITS architecture. The PCITS will require a number of components, including a CA, a Tutorial Knowledge Base, a Graphical User Interface (GUI) and a Student Model. The components will be described in section 4, which proposes a standard Oscar PCITS architecture that is generic and incorporates the required components.

4 Oscar PCITS Architecture

The Oscar PCITS is independent of a particular learning styles model and of the subject domain being taught. As such, it is important that the Oscar PCITS architecture allows for the learning styles model and tutoring domain to be changeable by keeping the system knowledge separate from the functionality. Additionally it is important for the adoption of the Oscar PCITS that the development time of tutorials be minimised. Therefore, a modular architecture is most appropriate as it allows individual modules to be reused and replaced as necessary.

Typical adaptive educational systems contain student, domain, pedagogical and presentation components (Wenger, 1987). A PCITS additionally requires a conversational agent component that can manage a natural language dialogue.

A general expert system architecture (Latham, Crockett & Bandar, 2010) was adapted for the Oscar PCITS. Figure 6.4 shows the proposed generic architecture of the Oscar PCITS, which has been designed in a modular structure with component reuse in mind. The generic architecture allows alternative tutorial knowledge bases and CA scripts developed following phase 2 of the methodology to be simply 'plugged in' to adapt the tutoring to new subjects. Similarly, different learning styles models may be applied by replacing the Learning Styles Predictor component (created following the methodology phase 1).

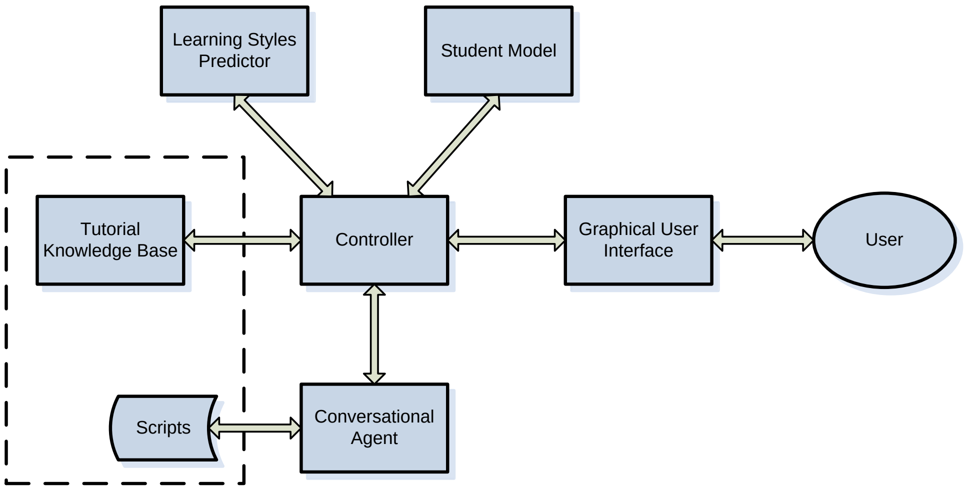


Figure 6.4. Generic Oscar PCITS Architecture

Controller

The controller is the central manager of the system, responsible for communicating with all components and managing the learner interaction. All communication and information passed between modules passes through the controller. In practice, this component may be combined with the GUI module.

Graphical User Interface (GUI)

The GUI is the user interface, responsible for display, managing events (such as clicking of buttons etc.) and sending communication to and from the user. The display consists of a webpage that provides instructions, displays questionnaires, tests, images, documents, interactive movies and a chat area which is used to communicate with the user. As Oscar PCITS directs the tutoring conversation, no navigation buttons are included as there is no menu system.

The modular nature of the architecture means that the GUI component could be changed to reflect the application needs, such as being deployed as an application on a smartphone or including a speech and voice recognition module.

Student Model

The student model component is responsible for knowing all information about individual students, such as their identifier and password, level of knowledge, topics visited, test scores and learning styles. The student model component receives and sends information from and to the controller about the student. The student model is recorded in a student model database.

Conversational Agent (CA)

The conversational agent component is responsible for accepting natural language text and information about topic and learning style from the GUI, tutorial knowledge base and learning styles predictor components via the controller, and generating a natural language response. The CA accesses a database of conversation scripts (related to but not linked to the tutorial knowledge base) in order to match the input to rules that generate a response. The CA records the dialogue in log files that can be accessed by the controller.

The CA selected must allow information to be passed in and out using variables.

Tutorial Knowledge Base

The tutorial knowledge base is responsible for managing course information, such as topics and their breakdowns, related tests and teaching material, which is accessed from a Tutor Material database. All tutor information is related to a particular tutorial module and will be categorised according to teaching style (related to learning style). The tutorial knowledge base will receive information and instructions from the GUI, learning styles predictor and CA components via the controller, and will send information to the GUI and CA via the controller.

Learning Styles Predictor

The learning styles predictor component is responsible for accessing information about learning styles and related teaching styles, held in a learning styles database. This component will receive information from the CA, GUI and student model to predict a student's learning style. Given learning style values from the student model and teaching material values from the knowledge base, this component will apply an adaptation algorithm to determine the most appropriate adaptation for an individual tutorial question. This module is developed by following phase 1 of the Oscar PCITS methodology.

An implementation of the Oscar PCITS architecture will be described in the next chapter, Chapter 7.

5 Conclusion

This chapter has proposed a novel conversational intelligent tutoring system called Oscar that can predict and adapt to an individual's learning styles whilst directing a tutoring conversation. Natural language interfaces are intuitive for humans, as we use conversation to communicate in real life. A more intuitive and familiar interface for a tutoring system can help learners feel more confident, which can improve their motivation and the effectiveness of their learning. Conversational interfaces can also adopt the constructivist style of tutoring used by human tutors, and thought to aid a deeper understanding of a subject. Oscar PCITS includes the three main intelligent technologies offered by ITS – curriculum sequencing based on individual learning style and existing knowledge, intelligent solution analysis and problem solving support. Implicitly predicting learning style during a tutoring conversation removes the requirement for learners to complete onerous and time-consuming questionnaires. By continually predicting learning style during tutoring, any changes to learning style or differences in learning style for different subjects can be automatically picked up and adapted to. Adapting the tutorial to an individual's preferred learning style aids the effectiveness of the learning experience.

An original, generic methodology was proposed to construct Oscar PCITS. The three-phase methodology is independent of the learning styles model and tutoring domains selected. Phase 1 of the methodology instructs the development of a Learning Styles Predictor module, detailing the extraction and analysis of knowledge from the selected learning styles model and the design of logic rules that apply this knowledge for learning styles prediction. A set of generic logic rules were extrapolated from the logic rules created for the exemplar FS model, which can be applied to other learning styles models as they are based on learner behaviour.

Phase 2 of the methodology directs the design of a tutorial conversation, from the capture of a tutoring scenario to the design of the conversation and scripting of the CA. A three-level model of a tutorial conversation was proposed and a set of generic tutorial question styles and templates developed to allow for quicker development and more consistent tutorial styles. Finally, phase 3 of the methodology requires the construction of the PCITS architecture.

A generic architecture was proposed for Oscar PCITS, which is modular to allow for component reuse and easier and quicker development and maintenance. The architecture is independent of the choice of learning styles model and the tutoring domain applied.

To validate the proposed generic methodology and architecture for Oscar PCITS, an experimental study is required. The next two chapters will describe an implementation of Oscar PCITS and experiments conducted to investigate Oscar PCITS' success in tutoring and predicting learning styles.

6 Chapter Highlights

- Oscar CITS is a novel conversational intelligent tutoring system that automatically predicts and adapts to learning style while directing a tutoring conversation.
- Oscar CITS is independent of the learning styles model selected and the tutoring domain.
- An original, generic methodology was proposed to construct an Oscar Predictive CITS (PCITS) which can automatically predict learning styles during a tutoring conversation.
- Oscar's innovative predictor module mimics a human tutor by using detected learner characteristics to dynamically predict learning style from a natural language tutoring dialogue.
- A set of generic logic rules to predict learning styles was created.
- A three-level model of a tutorial conversation was proposed.
- A set of generic tutorial question styles and templates were developed.
- A generic, modular architecture for assembling Oscar PCITS was proposed.

Chapter 7 Implementation of Oscar Predictive CITS

1 Introduction

The Oscar Predictive CITS (PCITS) proposed in Chapter 6 can dynamically predict a student's learning styles whilst directing a tutorial conversation. A generic methodology and architecture for developing the Oscar PCITS were proposed, which are independent of learning styles model and subject domain. Chapter 5 explored the strategy and methodology for predicting learning styles from a natural language dialogue, using Felder-Silverman (1988) as an example learning styles model.

This chapter will present the development of a prototype Oscar PCITS following the methodology and architecture, and utilising the generic designs, proposed in Chapter 6. The prototype Oscar PCITS delivers a conversational tutorial for the subject domain of the database language Sequential Query Language (SQL). The learning styles model adopted for the prototype Oscar PCITS is the Felder-Silverman (FS) model. In order to validate the methodology and architecture, the prototype Oscar PCITS will then be tested empirically, as described in Chapter 8.

2 Implementing Oscar PCITS

To validate the methodology and architecture proposed in Chapter 6, a prototype Oscar PCITS was implemented. The remainder of this chapter will describe the development of the prototype Oscar PCITS with reference to the 3-Phase Methodology proposed in Chapter 6 and repeated in Table 7.1.

3 Phase 1: Creating the Learning Styles Predictor Module

For the prototype Oscar PCITS, the FS model was adopted (Felder and Silverman, 1988) as it has a small number of dimensions (which is more feasible to implement) and it models the learning styles of engineering students, who will make up the initial experimental groups (see Chapter 5, section 2). Following steps 1.1, 1.2 and 1.3, the FS model was examined to extract the knowledge of behaviour and language traits indicative of learning styles. The resulting knowledge was further analysed, resulting in a table of behaviour characteristics (Table 5.5) and key words (Table 5.6) to be captured during a tutoring conversation. The final step in phase 1 (step 1.4) is to convert the FS knowledge into a set of logic rules that can continually increment learning style values during tutoring (see Appendix 2). Chapter 5 describes in detail these steps of the methodology and the studies conducted using the FS model.

Table 7.1. 3-Phase Methodology for Creating Oscar PCITS.

Phase 1: Create the Learning Styles Predictor Module

1.1. Select a Learning Styles Model
   a. Reduce the learning styles model if necessary
   b. Extract the behaviour characteristics
1.2. Map learning style behaviour to the conversational tutoring style
1.3. Analyse the learning styles model for language traits
1.4. Adapt the generic logic rules to predict learning styles

Phase 2: Design a Tutorial Conversation

2.1. Capture the tutorial scenario and questions (including movies, voice, images, examples, etc.) from human tutors in a specific domain
2.2. Determine the conversational structure/style
2.3. Map tutorial questions onto the generic question styles and templates
2.4. Script Conversational Agent natural language dialogue for each tutorial question using the 3-level model
2.5. Link tutorial dialogue to logic rules through Conversational Agent variables

Phase 3: Construct the PCITS Architecture

4 Phase 2: Designing the Tutorial Conversation

For the prototype Oscar PCITS the subject domain of the database Sequential Query Language (SQL) was selected. This subject was chosen because the target audience for the pilot study is undergraduate computing students, for whom a Databases course including SQL is compulsory. Additionally, SQL is widely taught in undergraduate computing courses and so the opportunity for reusing an SQL tutorial is high. The tutorial conversation was designed following phase 2 of the methodology, as described in the following sections.

4.1 Step 2.1: Capturing the Tutorial Scenario

The first step in designing a tutorial conversation was to capture knowledge and information from human tutors. The scope of the tutorial was considered first, in terms of its length, the syllabus and learning outcomes. Interviews were conducted with undergraduate level database course tutors to identify important SQL concepts for the tutorial syllabus. It was decided that a revision tutorial for basic SQL would be of most benefit to undergraduate students of several taught units. This decision was based on the fact that basic SQL is taught as part of several units, and included in examinations, however the use of SQL is a skill that may be forgotten if not used regularly. An online revision tutorial would offer students personal assistance in revising basic SQL at a time and place to suit them, such as during the examination period when formal teaching has ceased.

After taking into account the SQL taught in several unit syllabuses, a syllabus for the revision tutorial was developed which covered some basic SQL commands and concepts. Ten tutorial questions (Table 7.2) and a multiple choice question (MCQ) test (see Appendix 4) were devised to cover the learning outcomes of the tutorial. The syllabus was accepted by SQL lecturers based on their knowledge of student requirements. The MCQ test is required to assess existing knowledge before the start of the tutorial and the same MCQ test is then used to assess learning gain after completion of the tutorial. Twelve questions that test the knowledge and use of SQL covered in the tutorial syllabus were selected from an existing set of MCQ test questions used to support teaching of SQL in the department. The MCQ test was assessed by SQL lecturers who agreed it adequately assessed the learning outcomes from the tutorial syllabus. The syllabus was accepted by SQL lecturers based on their knowledge of student requirements.

To capture the tutorial scenario, a document was produced (called the '*tutorial conversation blueprint*') in consultation with lecturers which contained a conversation script for each question, including possible learner answers and tutor's responses to these. For each learner response, a further tutor response was written, and so on, until each question in the tutorial had a number of different paths depending on individual learner knowledge and responses. The design of the tutorial conversation was a time consuming and iterative process, however by planning and detailing the dialogue at this point, the development of the conversational agent was more efficient. Resources such as examples, movies, images etc. were embedded into the tutorial conversation as appropriate. The resources were gathered or created where necessary, to support the prediction of various learning styles. Appendix 3 shows an excerpt of the tutorial conversation blueprint document.

4.2 Step 2.2: Determining the Conversation Structure/Style

The three-level model of a tutorial conversation defined in the methodology (see Chapter 6, Figure 6.1) was applied to the tutorial conversation scenario. As the main 'tutoring level' dialogue was written during step 2.1, this involved adding the 'social level' and the 'discussion level' conversation to the tutorial conversation blueprint document.

For the social level, an existing set of common off-task user dialogue and responses developed for the ConvAgent Student Debt Advisor (ConvAgent Ltd, 2005) was adapted. This set of dialogue included swearing, general social comments and indicators of boredom, e.g. “How long is your hair”, “I am bored”, “Why should I answer”, “Are you real”, “Tell me again”. The dialogue set was adapted to remove the unrelated (i.e. pertinent to debt advice) comments, and to amend the responses to relate to the tutorial, e.g. by reminding students that they need to focus on their study. By analysing actual dialogue from tutoring sessions, it is anticipated that this dialogue set can be expanded and reused for future implementations.

The discussion level required a list of Frequently Asked Questions (FAQs) and answers to be captured from interviews with SQL lecturers and added to the tutorial conversation blueprint document, along with related resources such as examples and diagrams. Again, it is anticipated that this set of SQL-related FAQs be expanded on analysis of actual tutoring dialogues.

4.3 Step 2.3: Mapping Tutorial Questions to Generic Styles and Templates

In Chapter 6, a set of generic question styles and templates was described, which capture different learner behaviour characteristics (identified in phase 1 step 1.2). In this step, each tutorial question was considered alongside the generic question styles and templates, to see if a style or template could be applied. If so, the question was reorganised where necessary to follow the template, and additional resources (such as diagrams) were gathered or created where necessary. The tutorial dialogue was then updated in the tutorial conversation blueprint document. Table 7.2 maps each tutorial question to the generic question styles and templates.

Table 7.2. SQL Revision Tutorial Outline

Tutorial Question	Styles Applied	Templates Applied
Question 1 – DDL	Theoretical	Hints
Question 2 – DML	Theoretical	Hints
Question 3 – SELECT *	Practical	
Question 4 – Datatypes	Practical, Trick Question	Hints
Question 5 – Query with join	Practical, Process	Choice of Approach, Hints
Question 6 – Functions	Theoretical	Hints
Question 7 – Query with functions	Practical	Hints
Question 8 – GROUP BY	Theoretical	Hints
Question 9 – Query with range	Practical, Process	Choice of Approach, Hints
Question 10 – DISTINCT	Theoretical	Hints

Appendix 5 shows diagrams that are examples of applying the generic question templates to tutorial questions.

4.4 Step 2.4: Scripting the Conversational Agent Dialogue

This step of the development methodology is the complex and time-consuming task of developing the CA scripts for the tutoring dialogue. For the prototype, the Convagent Ltd (2005) InfoChat CA was selected as it is a text-based CA that allows information to be captured and received using variables. InfoChat adopts a pattern matching approach, which will be best suited to student dialogue as it is flexible enough to cope with bad grammar, misspellings and use of ‘text-chat’ language. InfoChat CA scripts must be scripted using the PatternScript language (Michie and Sammut, 2001). Scripts are made up of rules that consist of patterns and responses, where each pattern matched to an input generates a response. PatternScript allows the scripts to be developed modularly by grouping rules into sets called contexts (Sammut, 2001).

4.4.1 Organisation of CA Scripts

Before starting to develop the CA scripts, it was important to decide on a strategy for organising and navigating script contexts to manage the tutoring dialogue. As described in Chapter 2, Sammut (2001) described a way of managing conversations by grouping rules into sets, called *contexts*. Each context contains rules pertaining to a particular topic or goal of the conversation, thus adding contextual information to the current user utterance. There is always a current context representing the current state of the conversation. PatternScript allows control to be passed between contexts, which is useful in a structured conversation, such as a tutoring dialogue.

In Step 2.2 of the methodology (Section 2.2), the tutoring conversation was mapped to the three-level model. The organisation of contexts for this model adopts the *filter* and *backup context* approach proposed by Sammut (2001). Figure 7.1 shows the application of these special scripts to the tutoring conversation, which will be described below:

1. The *social level* of the conversation is managed using a *filter context*. All learner utterances will first be ‘filtered through’ this context to pick up any bad or abusive language, or any signs of boredom, and give an appropriate response (such as ending the session if swearing is detected,

or giving a motivating message to bored learners). The assumption is that most utterances will pass through the filter context and down to the current context which manages the current tutoring question (the *tutoring level*).

2. Within a tutoring question context, a mechanism is required to cope with unmatched utterances, as it is not possible to anticipate all of the possible utterances a learner may make. Each tutoring question context must contain a 'catch-all' rule whose purpose is to push all unmatched utterances down to a *backup context* (the *discussion level)*.
3. The backup context contains rules scripted to discuss FAQs. Where an utterance does not match any rule in the backup context, control will be returned to the tutoring question context where the CA should give a response which asks for more information and reminds the learner of the tutoring question, in the hope of getting the conversation back on track.

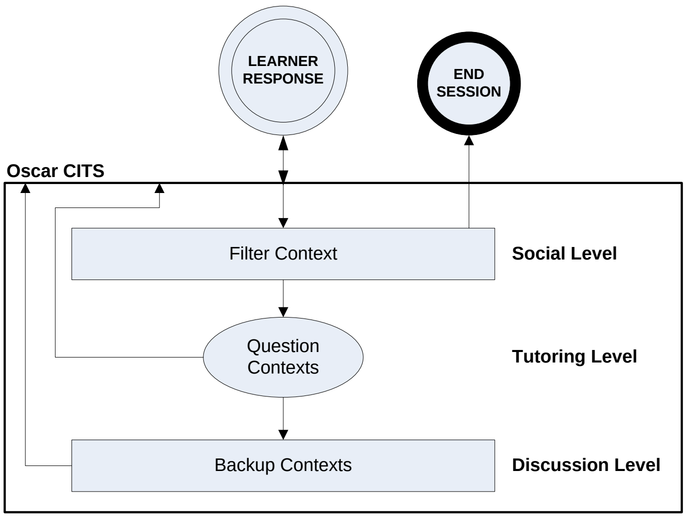


Figure 7.1. Management of Conversation Levels Using Different Types of Contexts

When organising the tutoring level CA scripts, it was initially planned to map a single context to each tutorial question. However during development it became apparent that this would not be sufficient for every question as some of the contexts contained too many rules which became too complex to manage. For questions applying a generic question template (see section 4.3) more than one context was required, as described below.

The *Hints* question template (Chapter 6, Figure 6.2) has two main paths:

1. The learner is asked the question, gives the answer and moves on.

2. The learner does not know the answer, or gives the wrong answer. The learner is then presented with different hints until they give the correct answer or the answer is explained.

Therefore, two contexts are required as follows:

1. A context to ask the question and determine if the learner knows the answer. If the learner knows the answer, control will be passed to the next question. If not, control will be passed to the related 'hints' context.
2. A context to give hints and help the learner remember the answer. Once the answer is given, control will be passed to the next question.

The *Choice of Approach* question template (Chapter 6, Figure 6.3) has a number of paths:

1. The learner is asked the question, and whether they wish to answer in one go or step by step.
2. Path 1: the learner answers in one go. They are given three attempts to get the right answer. If the learner gives the right answer, they move on to the next question. Otherwise, they are given help with the steps of the process where errors are apparent.
3. Path 2: the learner is given guidance on each step of the process, until they give the correct answer and move on to the next step. Finally, they move on to the next question.

Therefore, at least four contexts are required depending on the number of steps in the process, as follows:

1. A context to ask the question and determine how the learner wishes to answer – in one go or step by step.
2. A context to ask the question and determine if the learner knows the answer. If the learner knows the answer, control will be passed to the next question. If not, after three attempts, control will be passed to the context related to the step in the process containing the error.
3. A context to ask the question relating to the first step in the process, and determine if the learner knows the answer. If the learner knows the answer, control will be passed to the next step in the process. If not, help the learner remember the answer. Once the answer is given, control will be passed to the next step in the process.

4. A context to ask the question relating to the second step in the process, and determine if the learner knows the answer. If the learner knows the answer, control will be passed to the next step in the process (or next question). If not, help the learner remember the answer. Once the answer is given, control will be passed to the next step in the process (or next question).
5. Further contexts similar to 3 and 4 above, one for each step in the process.

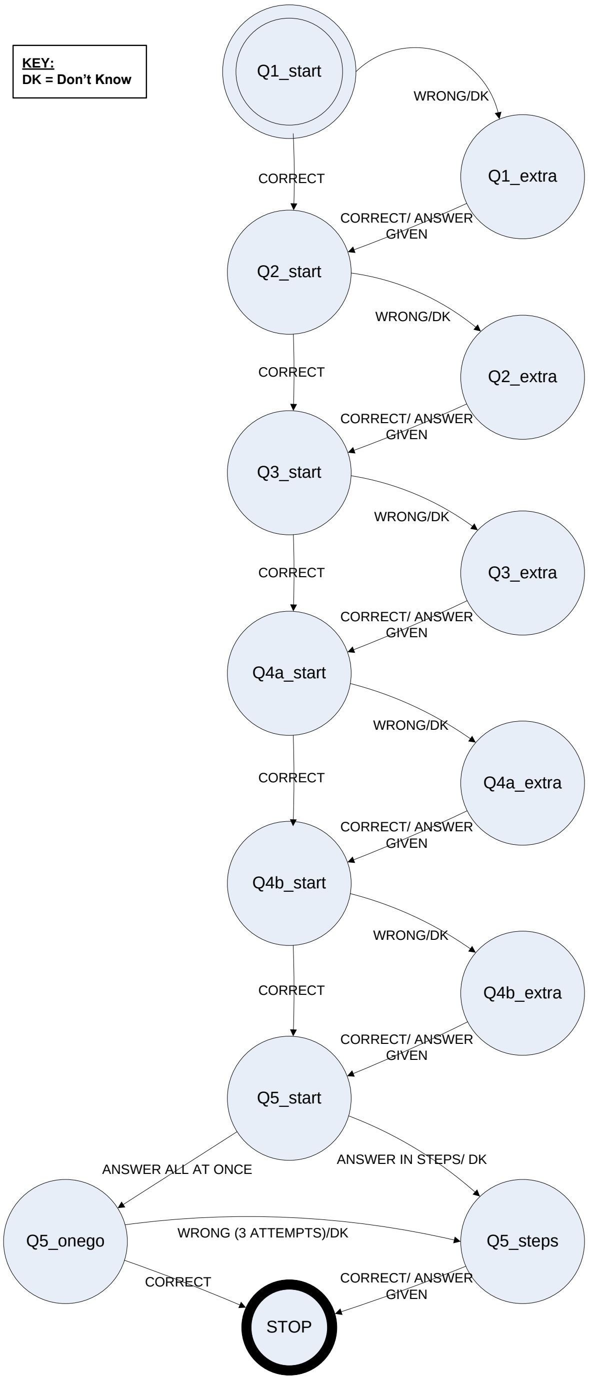


Figure 7.2. Tutoring Conversation Flow Through Contexts

Figure 7.2 shows the flow of control through contexts for the first five tutoring questions. In the diagram, for question 5 three of the four ‘steps’ contexts have been omitted for clarity. Navigation is managed by rules within the scripts, with control being pushed to new contexts specified within the rule response.

4.4.2 Scripting the Tutorial Conversation

Once the strategy for organising the scripts had been devised, the process of scripting the tutoring conversation could begin. The tutoring conversation blueprint document contained a walkthrough for each question’s dialogue, and so this dialogue needed to be translated into PatternScript. As mentioned previously, each rule in a PatternScript context contains a number of patterns that match the user input, and a response that forms the CA output. Each user utterance in the tutoring conversation blueprint document maps to at least one rule in the CA script. A simple example of an FAQ rule (named <FAQ-select1>) from one of the tutorial scripts is shown in Table 7.3. In the rule, *a* is the activation level used for conflict resolution (Michie, 2001); *p* is the pattern strength followed by the pattern that is matched against the user utterance. *r* is the CA response. Also seen in the example is the wildcard (*) and macros (<explain-0>) containing a number of standard patterns that are each matched separately. When the rule fires, the variable *FAQ* is set to ‘true’ by the *<set> command.

Table 7.3. Example CA script: FAQ rule.

```
<FAQ-select1>
a:0.01
p:50  *<explain-0> *select*
p:50  *select* <explain-0>*
p:50  *<remind-0> *select*
p:50  *select* <remind-0>*
p:50  *<confused-0> *select*
p:50  *select* <confused-0>*
r:    The SQL SELECT command is used to retrieve data from
      one or more database tables. *<set FAQ true>
```

The procedure followed for scripting each context was as follows:

1. Create a text file with a unique name for the context.
2. Create a starting rule that fires when the context is invoked.
3. Create a ‘catch-all’ rule that fires when no patterns match, and moves control to the backup context.
4. Script all CA rules for the context.

5. Test the individual context to check that rules fire when expected, and amend any patterns, pattern strengths or activation levels as necessary. Repeat until it works.

The procedure followed for scripting each rule was as follows:

1. Create a unique rule name for the user utterance.
2. Consider the user utterance. Extract the important words and create a pattern to match the utterance, using the wildcard to match unimportant words.
3. Consider different ways of phrasing the utterance, e.g. using words in a different order, and create patterns for each different phrase.
4. Consider words with the same meanings as the important words, and create macros that will substitute those words into the pattern. Update the pattern to refer to macros.
5. Create the tutor response. This may involve creating the output utterance, setting the value of variables and moving between contexts.

Using PatternScript it was possible to tightly control the conversation by applying logic that used the setting of variables by the CA script to communicate with the controller. An example was when the CA response required an image or movie to be shown – a variable was set within the script, its value being the image name, and the controller then used this variable to direct the GUI to show the right image along with the CA response.

Scripting languages are not powerful enough to check rigid expressions requiring case-sensitivity, non-alphanumeric symbols or the syntax of programming code, such as SQL queries. As some of the tutorial questions required SQL queries to be written, regular expressions were used to check the syntax of the code. Variable values were then set and passed to the CA to indicate whether the student utterance gave the correct answer, used the correct syntax, or contained a common error (e.g. incorrect case of a String). The CA could then use the variable value to issue an appropriate response. An example regular expression is shown in Table 7.4. The expression checks a WHERE clause condition against allowable SQL statements.

Table 7.4. Example Regular Expression

```
[Ww][Hh][Ee][Rr][Ee]\s+[Cc][Oo][Uu][Rr][Ss][Ee]\.[Cc][Rr][Ss]_[Tt][Ii][Tt]
[Ll][Ee]\s*=\s*'Database Systems'\s*;*
|[Ww][Hh][Ee][Rr][Ee]\s+[Cc][Rr][Ss]_[Tt][Ii][Tt][Ll][Ee]\s*=\s*'Database
Systems'\s*;*
|[Ww][Hh][Ee][Rr][Ee]\s+[A-Za-
z]{1,5}\.[Cc][Rr][Ss]_[Tt][Ii][Tt][Ll][Ee]\s*=\s*'Database Systems'\s*;*
```

Applying the regular expression in Table 7.4, all of the following allowable SQL statements would match and be accepted as correct:

```
WHERE course.crs_title='Database Systems'
WHERE  CRS_TITLE = 'Database Systems';
Where c.crs_title ='Database Systems';
```

However, the following incorrect statements do not match:

```
Where course_title='Database Systems';
Where c.crs_title = 'database systems';
Where crs_title = Database Systems
```

For the SQL revision tutorial, there were 38 contexts containing around 400 rules to manage the tutoring conversation, which demonstrates the complexity of the task.

4.5 Step 2.5: Linking the Tutorial Dialogue to Logic Rules

The final step in phase 2 of the methodology links the behaviour captured by the tutorial conversation to the set of logic rules from phase 1 that predict learning styles. The tutorial conversation blueprint document was updated for each learner behaviour event by noting the learning style(s) to be incremented as defined in the associated logic rule. The logic rules from Phase 1 (step 1.4) specify which learning styles are to be incremented when particular events occur (such as incrementing the Sensory learning style value after an example is shown).

Next, the CA scripts were amended to match the updated tutorial conversation blueprint document. For all CA scripts, each rule related to a behaviour event defined by a logic rule (as recorded in the tutorial blueprint document) was amended so that a variable would be set when that rule fired. Variables will be used to record behaviour information during the conversation.

Once the CA scripts had been updated they were fully tested using an InfoChat test engine to allow scripting errors to be diagnosed and corrected and to check that patterns and variable setting performed as expected. The validation of the scripted tutoring dialogue by tutors could not be done until the Oscar PCITS prototype architecture had been partially constructed to include the GUI and CA components. It was then possible to view resources linked to the dialogue during the tutoring conversation.

This step completes the design of the tutoring conversation, and the scripting of the CA to direct the tutoring. The next section will describe phase 3, the construction of the Oscar PCITS architecture.

5 Phase 3: Construct the Oscar PCITS Architecture

The Oscar PCITS architecture proposed in Chapter 6 and shown in Figure 7.3 was adopted. The prototype Oscar PCITS components were developed using Microsoft .net C# and ASP and built on a dedicated web server. The student model, tutorial knowledge base and learning styles predictor databases were developed using mySQL. The conversational agent used, by permission of ConvAgent Limited, was InfoChat and scripts were developed using its associated scripting language, PatternScript. The construction of the components will now be described.

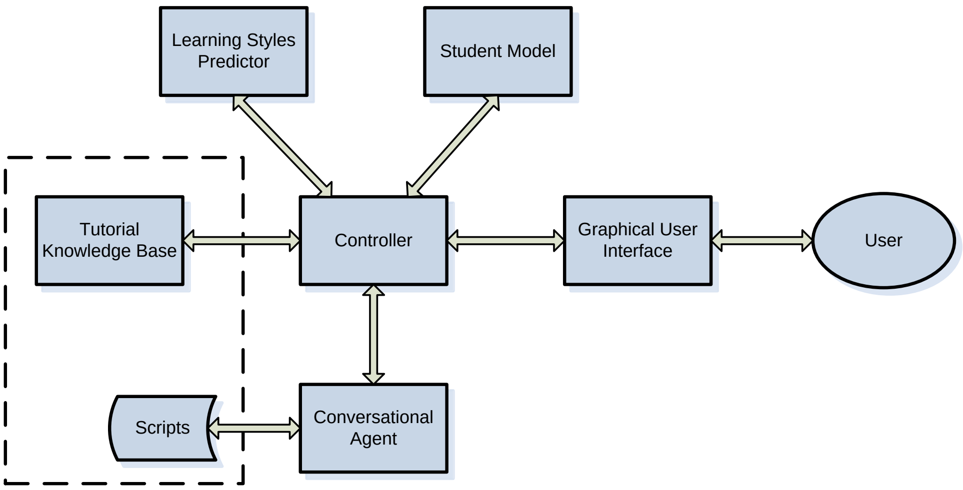


Figure 7.3. Oscar PCITS Architecture

Controller

The *controller* is responsible for managing the tutorial, and for communicating with all components. At the start of the first tutoring session, no initial learning style values will exist for a student. During the tutoring conversation, learning style values will be incremented depending on the learner's tutoring conversation. Periodically, the value pairs of each learning style dimension will be compared to reveal the student's overall learning style tendency for that dimension (i.e. the greater value). Learning style values depend on an individual's unique tutoring session, and if no evidence is gathered relating to a particular learning style dimension, that learning style will remain unclassified.

Graphical User Interface (GUI)

The user interfaces for most ITS show a number of windows on-screen at once, e.g. windows showing a menu of hyperlinks, explanation text, images or examples. Some ITS that predict learning styles have been designed with that end in mind (Cha et al., 2006) – i.e. their interface shows a menu of hyperlinks selected to capture user choices which are used to predict learning style. CITS like AutoTutor (Graesser et al. 1999) also have a screen divided into sections, one of which is a chat window where the tutoring conversation takes place.

The goal of Oscar PCITS is to mimic a human classroom tutorial. It was decided to keep the design of the GUI very simple but containing the elements that may be present in a face-to-face tutorial. Therefore the GUI contains a single chat box where the two-way tutoring conversation takes place. Relevant images, examples and exercises are displayed alongside the chat window when required, as in a classroom where a human tutor displays images or draws diagrams and examples on the whiteboard. Multimedia resources such as movies pop up in new windows of the browser, thus mimicking the display of such movies on a separate video screen in a classroom.

When designing the Oscar PCITS GUI, there were a number of considerations:

- Oscar PCITS is to be delivered online, and so must work in the standard Internet browsers (Mozilla Firefox, Internet Explorer, Safari).
- Oscar PCITS aims to mimic a human tutor by directing the tutoring conversation, therefore there will be no menu or user navigation. Users may ask to leave the session as part of the conversation, or close the browser.
- The conversation 'chatbox' will not display a history of the conversation as this is not available in a face-to-face tutorial, however learners could of course ask for reminders or repetitions as part of the conversation.

- As Oscar PCITS is intended to hold an individualised conversation, there will be no facility for group discussion amongst learners in the initial prototype. However discussion of topics will be possible with Oscar.

The prototype Oscar PCITS GUI is shown in Figure 7.4.

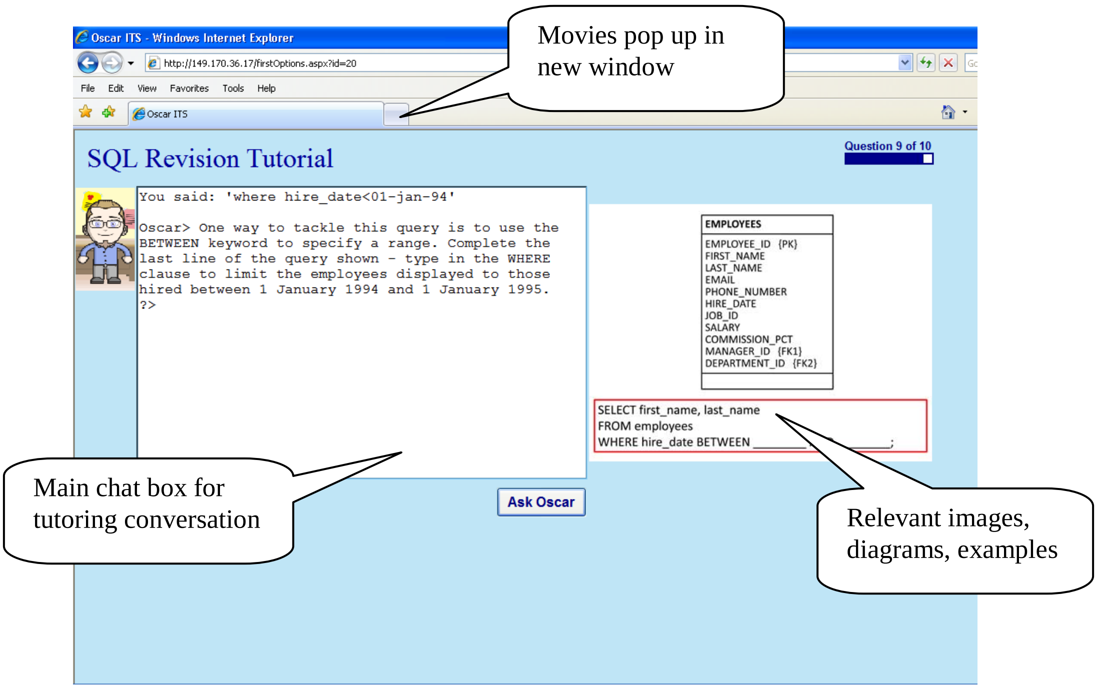


Figure 7.4. Oscar PCITS Learner Interface

Student Model

For the prototype Oscar PCITS, the student model contains three main database tables, as shown in Figure 7.5.

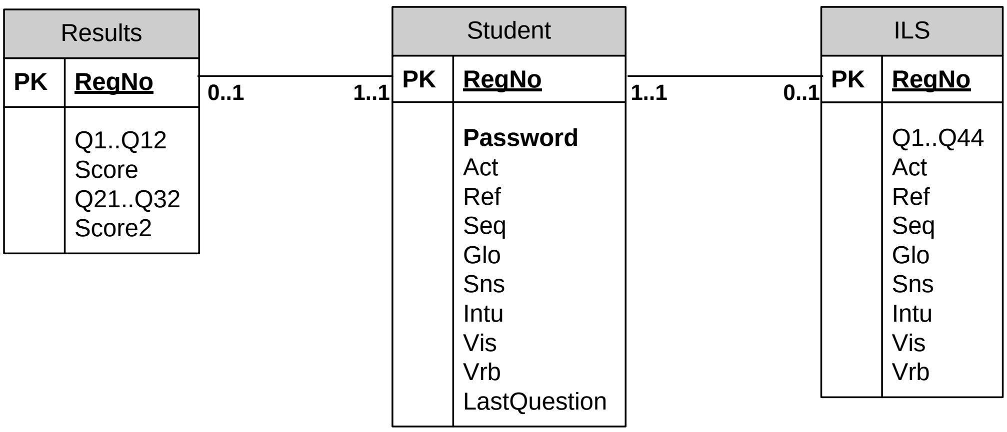


Figure 7.5. Student Model Class Diagram

- The *Student* table records the student registration number and password, a numerical value for each learning style (which has been predicted by Oscar

PCITS) and the last question the student attempted. Learning styles are held in eight values, one value representing each pole of the four dimensions, rather than one value per dimension. A student's overall learning style at any point in time is indicated by comparing the two values for each dimension, with the larger value representing their learning style. For a full Oscar PCITS with several tutorials, a separate table would be required to record the student's progress (i.e. last question) in each tutorial module.

- The *Results* table records the student answers and scores for associated MCQ tests. In Figure 7.5, for brevity Q1..Q12 indicates twelve attributes (as does and Q21..Q31), one for each test answer. In a full Oscar PCITS a student would have several results records, one for each tutorial module attempted.
- The *ILS* table was created for experimental reasons only, to record the student's answers to the Index of Learning Styles (ILS) questionnaire (Felder and Soloman, 1997), which measures FS learning styles, and the resulting scores for each learning style. In Figure 7.5, for brevity Q1..Q44 indicates 44 attributes, one for each ILS question. This table would not exist in a full Oscar PCITS, as it would not be necessary to compare Oscar PCITS learning style predictions with a formal questionnaire.

Conversational Agent (CA)

A CA module was written to interface with the adopted InfoChat CA which accessed the set of CA scripts. The design of the CA scripts followed steps 2.4 and 2.5 of phase 2 of the Oscar PCITS methodology, described in Section 4. The CA scripts are related to (but not linked to) the Tutorial Knowledge Base component.

The InfoChat CA records two log files of the conversation:

- A *log file* records the natural language text input, the scoring of all matched rules, the movement through contexts, the winning rule that fires and the resulting natural language text response. This log aids the testing of scripts, allowing adjustments to rules and patterns to ensure that the best fitting rule fires. The log file is only used during the development stage of the CA for debugging and improving CA scripts.
- A *state file* records the current context and variable values at the current point in the conversation. Whilst also useful for testing of scripts, variables are used

to pass information to and from the CA, and may be set by the controller or by the CA scripts.

A third log file for the CA component was included to record the timing of the tutoring conversation during experimentation, as follows:

- A *time log* records a narrative record of the conversation with date and time stamps for each interaction. This log file allows the tutoring dialogue to be reviewed separately from the technical detail, as it does not contain script information. It also enables the analysis of timings, word and interaction counts during experimentation.

Tutorial Knowledge Base

This module was developed by following steps 2.1 to 2.3 of phase 2 of the Oscar PCITS methodology, as described in Section 4. For the prototype Oscar PCITS, the tutorial knowledge base contains two main database tables, as shown in Figure 7.6.

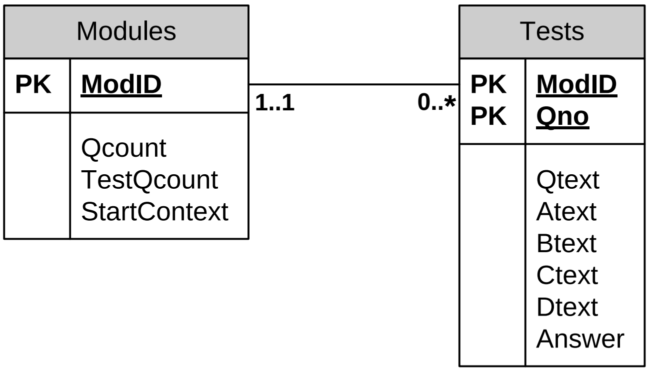


Figure 7.6. Tutorial Knowledge Base Class Diagram

- The *Modules* table has a record for each tutorial module, recording the number of tutorial and test questions and the starting context name of the CA script.
- The *Tests* table has a record for each module MCQ test question, recording the text of the question text and its four answer choices, and the correct answer.

The tutorial knowledge base is related to (but not linked to) the CA scripts that deliver the tutorial conversation.

Learning Styles Predictor

This module was developed by following phase 1 of the Oscar PCITS methodology, as described in Section 3. For the prototype Oscar PCITS, the learning styles predictor module implements the FS model logic rules. The logic rules increment learning style values in the student model based on learner behaviour during the conversation. The required behaviour information is extracted from the CA log files. Implementation of the adaptation of tutoring to learning styles will occur in a later prototype.

6 Example Learner Dialogue

Table 7.5 shows an example dialogue snippet from a tutoring conversation with Oscar PCITS that demonstrates problem solving support.

Table 7.5. Example Tutoring Dialogue Showing Problem Solving Support

Utterance	Dialogue
33. Oscar>	Now, in our books table suppose we want to store the price of the book where the maximum price of a book is 99.99. What would be the datatype?
34. Learner>	VARCHAR2
35. Oscar>	I'm sorry, that's wrong. Check your spelling and try again - type in the datatype for book price (with a maximum of 99.99).
36. Learner>	VARCHAR2 MAX 99.99
37. Oscar>	That's still wrong. Take a look at the table of common datatypes - can you now type in the datatype for book-price where the maximum price is 99.99?
38. Learner>	NUMBER(2,2)
39. Oscar>	That's correct - we need a datatype of NUMBER(2,2) in order to show the decimal point in book-price, with a maximum of 99.99.

In Table 7.5, the question adopts the ‘hints’ question template to prompt the learner to construct their knowledge and answer the question, rather than giving them the answer. In utterance 33, Oscar continues a question about datatypes. The learner gives the wrong datatype in utterance 34, and next Oscar indicates that the answer is wrong and gives the learner a second chance to answer in case of spelling or typing errors, repeating the question. In utterance 36 the learner tries again, still with the wrong answer. Oscar gives feedback that the answer is wrong (utterance 37), and helps by showing an image of the most common datatypes to see if the learner is able to pick out the required answer. In this action, Oscar is helping the learner by reminding them of the common datatypes and asking them to think about the correct datatype rather than simply telling them the answer. In utterance 38 the learner enters the correct datatype, extended to include the decimal places, and Oscar then confirms that they have the right answer and repeats the solution to reinforce the knowledge.

In Table 7.6, a second example dialogue snippet shows Oscar PCITS' intelligent solution analysis technique. The question requires learner to write a query, and adopts the 'choice of approach' template where learners can choose to answer in one go or in stages. In the example, the learner has chosen to write a query with step by step guidance from Oscar PCITS. Throughout this dialogue, a diagram showing the database tables and the partially written query is shown (like the example in Figure 7.4). In utterance 58, Oscar reminds the learner of the requirement of final stage of the query and asks the learner to type in a join condition. In utterance 59, the learner types in an SQL statement that is almost right, but is missing two apostrophes. Oscar analyses the statement and tells the learner that their statement would not generate the right results, as they have not used the correct syntax for a string (utterance 60). Oscar asks the learner for the symbol representing strings, and the learner types in " which is incorrect (utterance 61). Oscar then encourages the learner to try again (utterance 62), and the learner answers correctly (utterance 63). Finally, Oscar confirms that the answer is correct, displays the completed query in an image to reinforce the knowledge and gives positive encouraging feedback.

Table 7.6. Example Tutoring Dialogue Showing Intelligent Solution Analysis

Utterance	Dialogue
58. Oscar>	The final part of our query is a condition to choose all course titles that are called 'Database Systems'. As we want this condition to be included as well as the join, we need to use the AND operator. Can you type in the second join condition to complete the query?
59. Learner>	AND course.crs_title = Database Systems
60. Oscar>	That's nearly right, but this join statement would not match any of our course titles because we have not indicated that Database Systems is a string. Can you remember the symbol we use to indicate strings?
61. Learner>	"
62. Oscar>	You're nearly there. Have a think about VARCHAR2 SQL datatypes and type in the symbol we use to show that Database Systems is a string.
63. Learner>	'
64. Oscar>	That's right, so we have finished, and our final query is now shown in the picture. Well done.

7 Conclusion

This chapter has presented the practical implementation of the Oscar PCITS methodology and architecture proposed in chapter 6 for an example learning styles model, the Index of Learning Styles, and an example subject domain, SQL. The first phase of the methodology, relating to the learning styles knowledge, was described in detail in Chapter 5. The implementation of the second phase involved the development of a tutoring conversation for SQL revision designed to pick up learner behaviour relevant to predicting learning styles.

The implementation of each step in phase 2 was described, starting with the design of the tutorial by capturing the tutoring scenario from human tutors and structuring the conversation by applying the three-level model of a tutorial conversation. Next, the generic styles and templates given in the methodology were applied to the tutorial questions and their structure was updated. Once the tutorial conversation had been fully designed, it was scripted for the InfoChat CA using the PatternScript language. This complex task involved devising mechanisms for organising and navigating the CA scripts during the tutorial conversation for the three conversation levels. The process scripting of the CA was then described, and the logic used to manage the conversation using variables was explained. Finally, the tutorial conversation and CA scripts were linked to the logic rules for predicting learning styles using variables.

Phase 3 of the methodology involved constructing the architecture for Oscar PCITS proposed in Chapter 6. The implementation of each component was described.

The resulting prototype Oscar PCITS will be used to experimentally validate the methodology and architecture, and will be described in the next chapter.

8 Chapter Highlights

- Prototype Oscar PCITS developed following the generic methodology and architecture from Chapter 6.
- Oscar PCITS delivers an online SQL revision tutorial and implicitly predicts learning styles (FS model) during tutoring.

Chapter 8 Learning Styles Prediction Experiments

1 Introduction

The Oscar Predictive CITS (PCITS) proposed in Chapter 6 can dynamically predict a student's learning styles whilst directing a tutorial conversation. In order to validate the methodology and architecture proposed in Chapter 6, a prototype Oscar PCITS was developed. Chapter 7 presented the implementation of the prototype Oscar PCITS applied to the subject domain of the database language Sequential Query Language (SQL) and the Felder-Silverman (FS) learning styles model (Felder and Silverman, 1988).

In this chapter, three real-world experimental studies will be described, which were undertaken to investigate the success of Oscar PCITS in predicting learning styles during tutoring. There are seven main hypotheses identified relating to the prediction of learning styles, which are tested in 14 different experiments. Additionally, the Oscar PCITS' ability to tutor effectively was investigated by considering student learning gain and qualitative feedback from learners gathered from a user evaluation questionnaire. The results support six of the seven hypotheses, showing that it is possible to automatically predict learning styles from a tutorial conversation with a CITS in a real educational setting. On average, learners completing the Oscar PCITS tutorial achieved a positive learning gain and users valued the conversational tutoring style.

2 Experimental Design

In order to validate the methodology and architecture proposed in Chapter 6, an empirical study was undertaken that evaluated the success of Oscar PCITS in predicting learning styles. This section will outline the design of the experiment, describing the hypotheses made for predicting learning styles and the planned method of evaluation.

2.1 Hypotheses to be Tested

The main hypothesis for the experimental studies is:

> *H: It is possible to estimate a learner's learning style from a two-way tutoring discourse with a conversational agent (CA) tutor.*

This hypothesis requires a conversational agent tutor to be scripted to deliver a tutoring session to a single learner. The learner's behaviour and discourse during the tutoring session will be recorded and analysed to identify patterns of behaviour which may be indicative of particular learning styles. For example, can the number of words used by the learner indicate a particular learning style? Chapter 5 described the analysis of the FS model to extract typical behaviours for different learners. The hypothesis will be tested by comparing the results of the analysis to those of a formal learning style questionnaire completed by the learner. The main hypothesis may be broken down into several constituent parts each referring to separate aspects of learner behaviour, as described below.

- *H1: certain key phrases uttered during a CA tutoring conversation is indicative of learning style.*

Can any link be found between the sort of vocabulary used by the learner and their learning style? For example, if a learner asks "show me an example", does the use of the phrase "show me" (maybe as opposed to "give me") indicate a more visual learning style? Key words and phrases uttered by a learner will be compared to their FS learning style as assessed using the Index of Learning Styles (ILS) evaluation instrument (Felder and Soloman, 1997).

- *H2: the time taken for a particular tutor-learner interaction is indicative of learning style.*

The FS model suggests that time taken can be indicative of particular learning styles. For example, learners with a Sensory learning style are slow (as opposed to those with an Intuitive learning style who are quick). The tutorial duration will be calculated and compared to the overall group average to show whether the learner is the same as, slower than or quicker than the average. The results will then be compared to the learning style as assessed using the formal ILS questionnaire.

- *H3: the success of a learner in a particular style of tutoring question is indicative of learning style.*

As discussed in Chapter 5, some learners prefer more theoretical questions (representing a Reflective learning style) and some more practical questions (representing an Active learning style). The success of the learner in these different types of tutoring question will be compared by considering whether the learner answered the question correctly. The results will then be compared to the learning style as assessed using the formal ILS questionnaire.

- *H4: the amount of discussion a learner enters into with the tutor is indicative of learning style.*

  Some learners may prefer to discuss a question with the tutor (indicating a Verbal learning style) while others use more succinct language (indicating a more Visual learning style, for example). The total learner word count and the average word count per learner utterance will be compared to the overall group average to show whether the learner is the same as, more verbose than or less verbose than the average. The results will then be compared to the learning style as assessed using the formal ILS questionnaire.

- *H5: the success of a learner after experiencing a particular style of tutoring is indicative of learning style.*

  Some learners show preference for information presented in a particular way, for example by viewing a diagram or movie as opposed to a verbal explanation. The success of the learner in different types of information presentation will be compared by considering whether the learner answered the question correctly after a particular style. The results will then be compared to the learning style as assessed using the formal ILS questionnaire.

- *H6: a lack of attention to detail in answering questions is indicative of learning style.*

  Some learners may be patient with detail (suggesting a Sensory learning style) while others may be bored by detail and prone to making small errors (suggesting an Intuitive learning style). Some tutoring questions should be designed to test this attention to detail by presenting the answer in the explanatory tutor text. The conversation logs will be studied to see whether the learner makes mistakes even in these cases. The results will then be compared to the learning style as assessed using the formal ILS questionnaire.

- *H7: choosing to be guided through a process (or not) is indicative of learning style.*

Some learners prefer to follow a linear reasoning process (suggesting a Sequential learning style) while others prefer to make intuitive leaps (suggesting a Global learning style). Some tutoring questions should offer learners the choice of being guided through a complex task or of making an attempt at the answer straight away. The learner choices will be considered, and the results will then be compared to the learning style as assessed using the formal ILS questionnaire.

2.2 Evaluation Criteria

In addition to evaluating the Oscar PCITS prediction of learning styles by testing the hypotheses stated in section 2.1, Oscar PCITS' ability to tutor effectively will be investigated. Evaluation of the Oscar PCITS will therefore take place on three levels:

1. *Prediction:* Can Oscar PCITS predict learning styles dynamically from a two-way tutoring discourse? How successful is the prediction of learning styles?
2. *User evaluation:* How successful do learners feel the tutoring system is, and would they use Oscar PCITS in practice?
3. *Learning gain:* Does Oscar PCITS successfully tutor learners, i.e. do they learn anything?

2.2.1 Prediction of Learning Styles

This criterion evaluates the first research main question (stated in Chapter 1, Section 1), 'Is it possible to predict a student's learning style during a two-way tutoring discourse with a conversational agent tutor?'. As part of the anonymous registration with Oscar PCITS, each participant will be asked to complete the formal ILS questionnaire. The ILS questionnaire results will be recorded in the student model, and will be compared to the data gathered during the experiment to determine the accuracy of each particular variable in predicting a participant's learning style.

2.2.2 Qualitative User Evaluation

In addition to evaluating the prediction of learning styles, Oscar PCITS' ability to tutor effectively was investigated. A user evaluation feedback questionnaire was designed to gather qualitative user feedback following the Oscar PCITS tutorial. The feedback questionnaire consists of eight questions to be rated using a six-point LIKERT scale, four questions requiring a Yes/No answer, and three open questions, as shown in Table 8.1. A six-point LIKERT scale was selected rather than the more usual five-point LIKERT scale in order to force participants to express an opinion one way or another rather than selecting the centre rating.

Table 8.1. Oscar PCITS User Feedback Questionnaire

Please rate your experience of the following using the scale provided:	SCALE
1. Instructions	High 6/5/4/3/2/1 Low
2. Screen layout and design	High 6/5/4/3/2/1 Low
3. Tutoring	High 6/5/4/3/2/1 Low
4. How well did Oscar understand you?	High 6/5/4/3/2/1 Low
5. Did you find the tutoring helpful?	Yes 6/5/4/3/2/1 No
6. Was the conversation natural?	Yes 6/5/4/3/2/1 No
7. Was the conversation frustrating?	Yes 6/5/4/3/2/1 No
8. Do you feel Oscar helped you to revise?	Yes 6/5/4/3/2/1 No
9. Would you use a resource like Oscar:	
a. Instead of attending a face-to-face tutorial?	Yes/No
b. Instead of learning from a book?	Yes/No
c. As well as classroom tutoring?	Yes/No
d. Would you use the resource at all?	Yes/No
10. What else could Oscar have done to help you learn?	
11. Please state 3 positive points about using the Oscar computer tutor	
12. Please state 3 negative points about using the Oscar computer tutor	

2.2.3 Learning Gain

A frequently used measure of effective tutoring (Kelly and Tangney, 2006; Graesser et al., 2003; Lee et al., 2004) involves evaluating whether participants have increased their understanding of the topic at the end of the Oscar PCITS tutorial. Clearly success in implicitly predicting learning styles would be pointless unless some learning has also taken place during the tutorial. The pre-test/post-test approach was adopted (Kelly and Tangney, 2006; Graesser et al., 2003; Lee et al., 2004) whereby the same Multiple Choice Question (MCQ) test is completed before the tutoring conversation begins (*pre-test*) and after the tutoring ends (*post-test*). Learning gain will be evaluated by comparing the MCQ pre-test score to the MCQ post-test score to see whether test scores have improved, as follows:

Eq. 1. Learning gain = post-test score - pre-test score

3 Experimental Methodology

This section will describe the experimental methodology in testing the ability of Oscar PCITS to deliver an effective tutorial and to dynamically predict learning styles. For the experimental study, Oscar PCITS has been implemented to deliver a revision tutorial for SQL, based on the content of several undergraduate modules within the Department of Computing and Mathematics. However, by selecting basic SQL content and deciding on a revision tutorial, the Oscar PCITS SQL revision tutorial is suitable for any individual who has previous experience of SQL.

Initially, it was decided that a controlled pilot study of ten participants should take place to check that sufficient aspects of behaviour were being recorded during the tutorial, and to assess whether any areas of the tutorial or hypotheses required expansion. Following this small initial pilot study, a larger study was undertaken in a real teaching/learning environment, however there were difficulties in attracting sufficient numbers of participants who completed the full online tutorial. Finally, the Oscar PCITS SQL revision tutorial was integrated into a final year undergraduate module within the department, and a ‘natural learning environment’ study was undertaken. No participants in any study had previous experience using Oscar PCITS. As the experimental methodology was the same for all studies, the descriptions in this section relate to all three unless otherwise stated.

3.1 Description of Participants

Study 1

Ten participants were chosen whose first language was English and who had previous experience of an undergraduate ORACLE SQL course (but with various levels of expertise). This controlled study took place in an office setting where participants could be unobtrusively observed during their tutorial session with Oscar PCITS. Participants completed the tutorial individually in a single session.

Study 2

There were 43 participants in this study who had previous experience of an undergraduate SQL course and various levels of SQL expertise. A number of ‘SQL Revision Laboratories’ were timetabled to coincide with the revision period before the examinations took place. The laboratories were advertised to undergraduate and Masters students studying units that covered SQL. Additionally, several lecturers requested that their students complete the Oscar PCITS SQL Revision tutorial during timetabled laboratories for second year Information Systems units. The study took place in a ‘natural learning environment’: participants commenced the Oscar PCITS revision tutorial in the laboratories, and those who did not complete the tutorial in a single session continued the revision via the Internet at another time.

Study 3

There were 61 participants in this study, which also took place in a real teaching/learning environment. All participants were studying on the Advanced Database Applications unit, which is a final year unit for a computer science degree at Manchester Metropolitan University. The Oscar PCITS SQL revision tutorial was integrated into the first teaching week and during the timetabled laboratories, participants were asked to complete the revision tutorial. As basic SQL is a prerequisite, revision of SQL is important to the understanding of more complex SQL taught during the unit. In order to promote full completions of the tutorial, participants who completed the Oscar PCITS revision tutorial were awarded marks in recognition of their engagement. Participants commenced the Oscar PCITS revision tutorial in the laboratories, and those who did not complete the tutorial in a single session were able to continue the revision via the Internet at another time.

3.2 Methodology

For each study, participants were asked to revise their knowledge of SQL by completing the natural language Oscar PCITS SQL Revision tutorial. Each participant followed an individual learning path depending on their existing knowledge, behaviour and dialogue. The participant interaction with Oscar PCITS will be detailed in section 3.3. During the tutorial, each participant's current tutorial question was recorded in the student model to allow participants to end the tutoring session and restart the tutorial from the same question at a convenient time. All dialogue from the tutoring conversation was recorded in log files, along with values of the behaviour variables used to predict learning style described in Chapter 5. Following the study, the data gathered during participant interactions was analysed to explore the success of the tutorial in predicting participant learning styles. The experimental analysis will be described in section 3.4. Additionally, an evaluation of the success of the tutorial in terms of participant learning gain and the participant feedback was completed.

The next sections will describe the participant interaction and experimental analysis undertaken for all studies.

3.3 Participant Interaction

For all three experimental studies, participant interactions followed the same steps, as illustrated in Figure 8.1. After the anonymous registration, participants completed an online version of the formal ILS questionnaire which was recorded in the student model. Before starting the conversational tutorial, participants were presented with a pre-tutorial 12 question MCQ test, known as the *pre-test*, to assess their existing SQL knowledge. The pre-test results were stored in the student model. Next, Oscar PCITS directed a two-way conversational SQL revision tutorial that took on average 43 minutes, with each participant following an individual learning path depending on their existing knowledge and the dialogue. There were ten main tutorial questions. At the end of the tutorial, participants completed the same MCQ test (known as the *post-test*) to assess their learning gain, with the results being stored in the student model. Next, Oscar PCITS presented participants with a comparison of their test results (indicating their learning gain) and some feedback on their tutorial performance. Finally, participants were asked to complete a user feedback questionnaire.

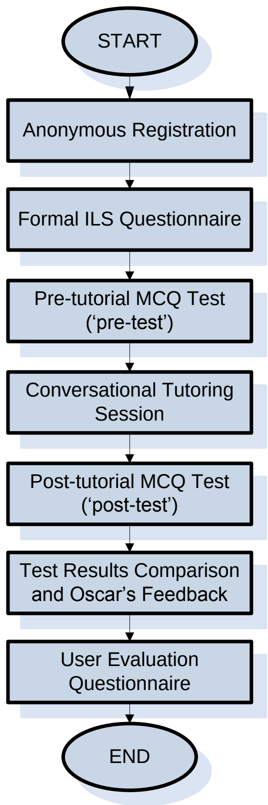


Figure 8.1. Experimental Oscar PCITS Tutorial Interaction

3.4 Experimental Analysis

The data gathered from the experiments was analysed to determine whether and how well the Oscar PCITS predicted learning styles. There were fifteen experiments designed to test the hypotheses in section 2.1. As described in Chapter 5, logic rules were incorporated into the Oscar PCITS which increment participant learning style values during the tutorial conversation. The logic rules incorporate several aspects of learner behaviour, some of which will also be considered separately in order to investigate whether they can be used to predict learning style within the Oscar PCITS. The analysis of results is different for each experiment, so will now be described separately.

Experiment 1: Logic Rules

In Chapter 5 the creation of logic rules by mapping various aspects of learner behaviour to the FS model was described. This experiment considers the learning styles scores resulting from the application of the logic rules to each participant's behaviour during the conversational tutorial. The logic rules increment associated learning style scores if behaviour is present. Resulting scores are held in eight learning style values in the student model, one for each possible learning style (i.e. Sensory, Intuitive, Visual, Verbal, Active, Reflective, Sequential, Global).

The learning style scores will be compared for each FS dimension to give a prediction of overall learning style tendency for that dimension. For example, for the processing dimension if the score for Active is higher than the score for Reflective, the participant is predicted to be Active. Where scores are equal, the learning style dimension is unclassified and will be excluded from the analysis. To assess the accuracy of learning style prediction, the predicted learning style for each dimension will be compared to the ILS questionnaire results. The number of correct predictions for each learning style is counted to produce an accuracy value that is the percentage of correct predictions for each learning style. This experiment tests the hypotheses H5, H6 and H7 defined in section 2.1 and generates prediction accuracies for all learning style dimensions.

Experiment 2: Extended Logic Rules

This experiment extends Experiment 1 by including additional criteria for some logic rules. A number of logic rules increment learning style values if a particular type of learning resource has been viewed by a participant, for example if an image has been viewed the Visual learning style score is incremented. Similar logic rules exist for other resources mapped to learning styles, such as examples and movies. In Experiment 2, for such logic rules the learning style value is only incremented if the participant is presented with the resource AND responds with a correct answer to the tutorial question directly following that resource.

As in Experiment 1, learning style scores will be compared to produce a prediction of learning style for each dimension, and the result compared to the ILS questionnaire result to generate a prediction accuracy percentage. This experiment tests the hypotheses H5, H6 and H7 defined in section 2.1 and generates prediction accuracies for all learning style dimensions.

Experiment 3: Tutorial Question Style

The FS model relates a learner's preference for practical or theoretical topics to their learning style. This experiment considers the style of tutorial questions where participants gave the correct answer, and counts the number of correct theoretical questions and the number of correct practical questions. The number of correct answers will be compared, taking into consideration the possible number of correct answers for theoretical and practical questions using the formula below:

Eq. 2. $$\frac{\text{Correct practical questions}}{\text{Total practical questions}} \quad \textit{compared to} \quad \frac{\text{Correct theoretical questions}}{\text{Total theoretical questions}}$$

Participants who have performed equally well in both styles of question are unclassified and will be excluded from the analysis. Where participants have performed better in practical questions, the Oscar PCITS predicts their learning style to be Active and Sensory. Participants who have performed better in theoretical style questions are predicted to be Reflective and Intuitive. The Oscar PCITS prediction will be compared to the ILS questionnaire results and the number of correct predictions counted for each learning style, to produce a prediction accuracy percentage. This experiment tests the hypothesis H3 defined in section 2.1 and generates prediction accuracies for the perception (Sensory/Intuitive) and processing (Active/Reflective) learning style dimensions.

Experiment 4: MCQ Question Style

Experiment 4 is similar to Experiment 3, but it considers improvements in MCQ test questions. Each MCQ test question is related to a tutorial question, and can therefore be related to a style of either theoretical or practical, as described for

Experiment 3 (see Appendix 6). This experiment investigates whether a preference in tutoring style has led to more improvements in related MCQ test questions. For each participant, the results of the post-test will be compared to those for the pre-test. Each MCQ test question where the answer for the pre-test was incorrect but for the post-test was correct (i.e. the participant has improved their answer following the tutorial) will be related to the tutoring style. The totals for practical and theoretical improvements, taking into account the opportunity for improvement, will be compared as follows:

Eq. 3. $$\frac{\text{Improved practical questions}}{\text{Incorrect practical questions (pre-test)}} \quad \textit{compared to} \quad \frac{\text{Improved theoretical questions}}{\text{Incorrect theoretical questions (pre-test)}}$$

Participants who have improved equally in both styles of question are unclassified and excluded from the analysis. Participants who have improved more in practical questions are predicted to be Active and Sensory, whereas participants who have improved more in theoretical style questions are predicted to be Reflective and Intuitive. The Oscar PCITS prediction will be compared to the ILS questionnaire results and the number of correct predictions counted for each learning style, to produce a prediction accuracy percentage. This experiment tests the hypothesis H3 defined in section 2.1 and generates prediction accuracies for the perception (Sensory/Intuitive) and processing (Active/Reflective) learning style dimensions.

Experiment 5: Word Count

The FS model states that Verbal learners learn better by discussing a subject. This experiment considers the total word count for participant utterances during the tutorial dialogue, and compares the total to the average word count for participant utterances across the sample. Where a participant has an above average word count, Oscar PCITS predicts they are Verbal learners, and where they have a below average word count, they are predicted to be Visual learners. The predicted learning style will be compared to the ILS questionnaire results, and the correct predictions counted for each learning style to produce a prediction accuracy percentage. This experiment tests the hypothesis H4 defined in section 2.1 and generates prediction accuracies for the input (Visual/Verbal) FS dimension.

Experiment 6: Number of Interactions

Similar to Experiment 5, this experiment considers the amount of discussion by counting the number of discourse interactions during the tutoring session. An interaction is defined as one participant response, which (like in a spoken face-to-face conversation) may consist of a single word or any number of sentences. Each participant's number of interactions will be compared to the average number of interactions across the sample. Where a participant has an above average number of interactions, Oscar PCITS predicts they are Verbal learners, and where they have a below average number of interactions, they are predicted to be Visual learners. The predicted learning style will be compared to the ILS questionnaire results, and the correct predictions counted for each learning style to produce a prediction accuracy percentage. This experiment tests the hypothesis H4 defined in section 2.1 and generates prediction accuracies for the input (Visual/Verbal) FS dimension.

Experiment 7: Word Count per Interaction

As in the previous two experiments, this experiment also measures a participant's preferred amount of discussion by considering the average number of words uttered by a participant per interaction. Each participant's 'word count per interaction' will be calculated by dividing the total participant word count by the number of interactions. Each participant's word count per interaction will be compared to the average word count per interaction across the sample. Where a participant has an above average word count per interaction, Oscar PCITS predicts they are Verbal learners, and where participants have a below average word count per interaction, they are predicted to be Visual learners. The predicted learning style will be compared to the ILS questionnaire results, and the correct predictions counted for each learning style to produce a prediction accuracy percentage. This experiment tests the hypothesis H4 defined in section 2.1 and generates prediction accuracies for the input (Visual/Verbal) FS dimension.

Experiment 8: FAQ Count

Experiment 8 considers a participant's preference for discussing a problem and asking questions by counting the number of Frequently Asked Questions (FAQs) they ask during the tutorial. As described in Chapter 6 the discussion layer of the tutorial conversation enables participants to ask questions related to the topic which have been scripted as FAQs. The number of FAQs asked will be counted for each participant and compared to the average number of FAQs asked across the sample. Where a participant asks an above average number of FAQs, Oscar PCITS predicts they are Verbal learners, and where they ask a below average number of FAQs they are predicted to be Visual learners. The predicted learning style will be compared to the ILS questionnaire results, and the correct predictions counted for each learning style to produce a prediction accuracy percentage. This experiment tests the hypothesis H4 defined in section 2.1 and generates prediction accuracies for the input (Visual/Verbal) FS dimension.

Experiment 9: Tutorial Duration

The FS model states that Sensory learners are careful but slow, whereas Intuitive learners are quick but careless. This experiment will consider the duration of the tutoring conversation. For each participant, the tutorial duration will be compared to the average duration across the sample. Where a participant has an above average tutorial duration, Oscar PCITS predicts they are Sensory learners, and where they have a below average tutorial duration they are predicted to be Intuitive learners. The predicted learning style will be compared to the ILS questionnaire results, and the correct predictions counted for each learning style to produce a prediction accuracy percentage. This experiment tests the hypothesis H2 defined in section 2.1 and generates prediction accuracies for the perception (Sensory/Intuitive) FS dimension.

Experiment 10: Time per Interaction

Similarly to Experiment 9, this experiment considers the average duration of a participant interaction. In this case, ‘time per interaction’ will be calculated by dividing the total duration by the number of interactions. Each participant’s time per interaction will be compared to the average time per interaction across the sample. Where a participant has an above average time per interaction, Oscar PCITS predicts they are Sensory learners, and where they have a below average time per interaction, they are predicted to be Intuitive learners. The predicted learning style will be compared to the ILS questionnaire results, and the correct predictions counted for each learning style to produce a prediction accuracy percentage. This experiment tests the hypothesis H2 defined in section 2.1 and generates prediction accuracies for the perception (Sensory/Intuitive) FS dimension.

Experiment 11: Reading Time

Experiment 11 considers a participant's aptitude with words by investigating their reading speed. The hypothesis is that the longer a student takes to read instructions (i.e. the less comfortable the student is with words), the more they tend towards the Sensory and Visual learning styles. The time taken to read ten Oscar words (known in this section as 'reading time') will be calculated for each participant, using the following formula:

Eq. 4. $$\frac{\text{Tutorial duration (seconds)}}{\text{Number of Oscar words}} \times 10$$

Each participant's reading time will be compared to the average reading time across the sample. Where a participant has an above average reading time, Oscar PCITS predicts they are Sensory and Visual learners, and where they have a below reading time, they are predicted to be Intuitive and Verbal learners. The predicted learning style will be compared to the ILS questionnaire results, and the correct predictions counted for each learning style to produce a prediction accuracy percentage. This experiment tests the hypothesis H2 defined in section 2.1 and generates prediction accuracies for the perception (Sensory/Intuitive) and the input (Visual/Verbal) FS dimensions.

Experiment 12: Approach to SQL Queries

In Experiment 12, the learner's approach to writing SQL queries will be considered. The FS model describes Sequential learners as preferring information to be presented in a steady progression of complexity and difficulty, whereas Global learners prefer to jump directly to more complex and difficult material. Following the generic 'choice of approach' question template in Chapter 6 (Figure 6.3), two questions in the tutorial (Q5 and Q9) were written with a choice of approach to writing SQL queries to solve a problem. For each question, participants who attempt the query straight away will be predicted to be Global learners whilst participants who ask for guidance will be predicted to be Sequential learners. Each participant has two predictions, one for each question. The predicted learning style will be compared to the ILS questionnaire results, and the correct predictions counted for each learning style to produce a prediction accuracy percentage. This experiment tests the hypothesis H7 defined in section 2.1 and generates prediction accuracies for the understanding (Sequential/Global) FS dimension.

Experiment 13: Attention to Detail

Following the generic question styles described in Chapter 6, one question in the tutorial has been written as a ‘trick question’, where the answer to the question is given in the explanatory text. This question (Q4) will test the participant’s attention to detail and reading skills. In this experiment, participants who do not answer the question correctly have not read the text carefully and so are predicted to be Visual and Intuitive learners. Participants who answer correctly are predicted to be Verbal and Sensory learners. The predicted learning style will be compared to the ILS questionnaire results, and the correct predictions counted for each learning style to produce a prediction accuracy percentage. This experiment tests the hypothesis H6 defined in section 2.1 and generates prediction accuracies for the perception (Sensory/Intuitive) and the input (Visual/Verbal) FS dimensions.

Experiment 14: Key Phrases

As described in Chapter 5, a list of key words and phrases that may be indicative of learning style has been drawn up. This experiment involves counting the occurrences of the key words and phrases (see Table 5.6) and comparing their occurrence to the related learning style. When a participant utters a key word or phrase, their learning style is predicted to be the learning style mapped to that key word or phrase. The predicted learning style will be compared to the ILS questionnaire results, and the correct predictions counted for each learning style to produce a prediction accuracy percentage. This experiment tests the hypothesis H1 defined in section 2.1 and generates prediction accuracies for all FS dimensions.

4 Results and Discussion

This section will present the collated results of all three studies, although each study was also analysed individually. The collated results represent a sufficiently large sample size so that sensible conclusions may be drawn. In cases where the results of individual studies showed different characteristics, they will also be discussed separately.

4.1 Overall Results

Not all participants from studies 2 and 3 completed the full tutorial session. Participants with only partially complete tutorial sessions were excluded from most of the analysis. Exceptions are Experiments 11 to 14, which require particular events to have taken place. Of those participants who completed the tutorial, 2 participants were excluded as they had not accurately completed the ILS questionnaire, answering 'a' for all questions. Therefore, for Experiments 1 to 10, 75 participant interactions were analysed.

Table 8.2. Learning Style Distribution

Learning Style	Study 1		Study 2		Study 3		All Studies	
	n	%	n	%	n	%	n	%
Sensory	2	20%	12	60%	31	69%	45	60%
Intuitive	8	80%	8	40%	14	31%	30	40%
Visual	8	80%	19	95%	38	84%	65	87%
Verbal	2	20%	1	5%	7	16%	10	13%
Active	4	40%	10	50%	29	64%	43	57%
Reflective	6	60%	10	50%	16	36%	32	43%
Sequential	3	30%	14	70%	28	62%	45	60%
Global	7	70%	6	30%	17	38%	30	40%

Table 8.2 shows the distribution of learning styles for the studies, assessed by the ILS questionnaire, and Figure 8.2 represents this data spread graphically. As expected, the split of learning styles assessed by the ILS questionnaire was not equal across the sample. Although on the whole three of the four FS dimensions were approximately equally spread, the Visual/Verbal dimension contained many more Visual than Verbal learners. This finding supports the observation in the FS model, which states that "most people of college age and older are visual". This has implications for the analysis of results for predicting the Visual/Verbal learning styles, as the dataset is so biased towards the Visual learning style.

Table 8.3 shows the participant learning gain results, calculated using Eq. 1. as described in section 2.2.3 (post-test - pre-test). Note that during study 2, two participants did not complete the post-test and so have been excluded from the analysis, leaving 73 participants whose learning gain was calculated. Note also in Table 8.3 the much lower average learning gain for study 3. This may be explained by participant motivation – participants of studies 1 and 2 wanted to revise SQL and actively engaged with the tutorial. For study 3, where participants were required to complete the tutorial but where the results of their learning 'did not count',

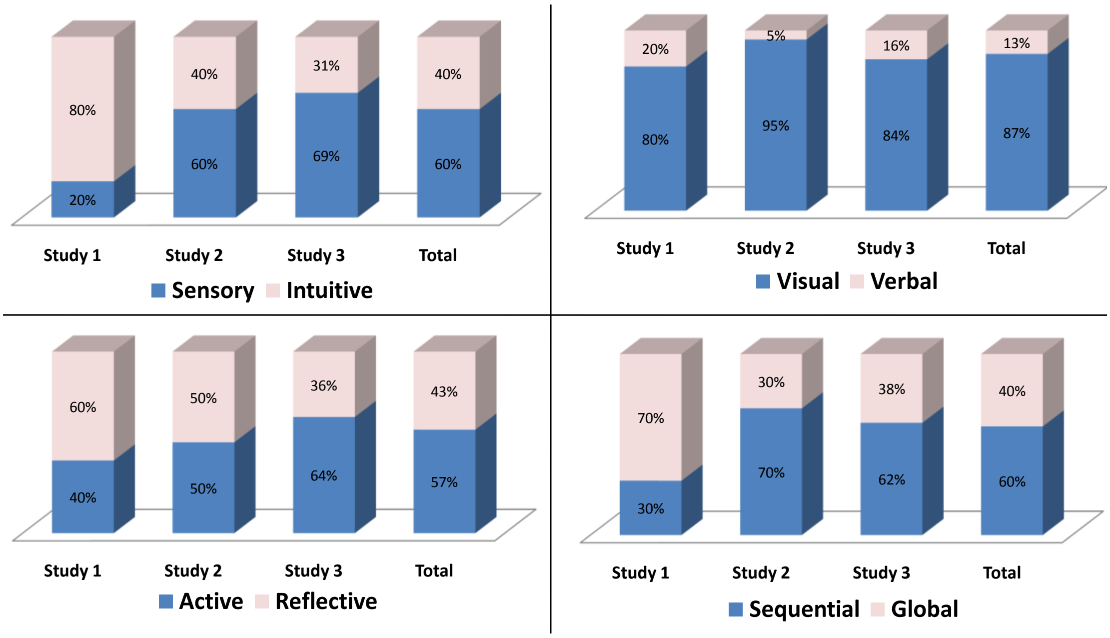


Figure 8.2. Learning Style Data Distribution

engagement with the SQL revision tutorial was much more variable. Also, study 3 took place in a real teaching/learning environment where negative factors such as distractions are expected to affect engagement. In study 3, six participants' test scores remained unchanged and 31 participants improved their test scores by an average of 19%, with three improving to achieve full marks. However, there were also eight participants whose test scores decreased by 16%. Even including these unchanged and decreased scores, the average learning gain for study 3 was 11%.

Table 8.3. Learning Gain Results

Study	n	Learning Gain		
		Mean (/12)	Standard Deviation	Mean %
Study 1	10	2.4	2.011	20%
Study 2	18	2.167	2.149	18%
Study 3	45	1.267	2.082	11%
Total	73	1.575	2.074	13%

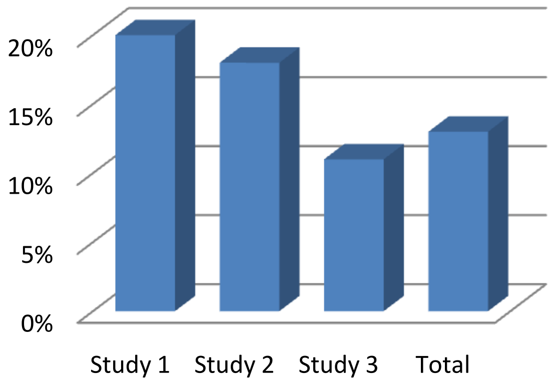


Figure 8.3. Percentage Learning Gain

The learning gain results show that participants increased their learning of SQL and improved their test results by an average of 13%. Therefore, the Oscar PCITS SQL revision tutorial did help participants learn. The results of each experiment testing the prediction of learning styles will now be discussed separately, followed by a summary of the participant feedback on using Oscar PCITS.

4.2 Experimental Results

The collated study results of the experiments to assess the prediction of learning styles are given in Table 8.4. As mentioned previously, the sample of 75 complete tutorial sessions has been supplemented for Experiments 11 to 14, as they do not require completed sessions, rather the requirement is that particular events have taken place.

In Table 8.4, the first line 'prior probability' is included for comparison, and represents the likelihood of predicting a learning style based on the distribution of learning styles across the sample of 75. This is a better comparison than simply using 50% because the spread of learning styles across the sample is not exactly equal. This is particularly true for the Visual/Verbal learning style dimension where 87% of participants are Visual.

The accuracies listed in Table 8.4 represent the ability of Oscar PCITS to predict a participant's learning style for that experimental measure. For example, in Experiment 1, Oscar PCITS has an accuracy of 80% in predicting the Intuitive learning style. This means that when a participant is predicted to be Intuitive using this measure, the prediction is accurate 80% of the time. For example, say a group of 20 students from a sample of 50 were predicted using the Experiment 1 measure to be Intuitive learners. If the ILS results agree that 16 of those 20 learners are Intuitive, however the ILS results show the remaining 4 are Sensory learners, the measure is accurate 80% of the time.

Note that for experiments where there are two predictions for each learning style because two comparisons were made (to the sample mean and median), only the best result is reported in Table 8.4. Each experiment's results will now be discussed separately.

Table 8.4. Experimental Results: Accuracy of Prediction of Learning Styles

	n	Sensory	Intuitive	Visual	Verbal	Active	Reflective	Sequential	Global
Prior Probability	*75*	*60%*	*40%*	*87%*	*13%*	*57%*	*43%*	*60%*	*40%*
Experiment 1 – Logic Rules	75	4%	80%	68%	10%	100%	0%	82%	33%
Experiment 2 – Extended Logic Rules	75	4%	77%	48%	30%	84%	0%	80%	33%
Experiment 3 – Tutorial Question Style	75	36%	50%	-	-	53%	73%	-	-
Experiment 4 – MCQ Question Style	73	60%	48%	-	-	52%	45%	-	-
Experiment 5 – Word Count	75	-	-	57%	30%	-	-	-	-
Experiment 5a – Word Count	49	-	-	55%	40%	-	-	-	-
Experiment 6 – Number of Interactions	75	-	-	48%	20%	-	-	-	-
Experiment 7 – Word Count per Interaction	75	-	-	60%	70%	-	-	-	-
Experiment 8 – FAQ Count	75	-	-	60%	20%	-	-	-	-
Experiment 9 – Tutorial Duration	75	67%	67%	-	-	-	-	-	-
Experiment 10 – Time per Interaction	75	49%	67%	-	-	-	-	-	-
Experiment 11 – Reading Time	75	49%	63%	72%	50%	-	-	-	-
Experiment 11b – Q1 Reading Time	95	51%	78%	47%	71%	-	-	-	-
Experiment 12a – Approach to SQL Queries (Q5)	89	65%	38%	-	-	-	-	74%	48%
Experiment 12b – Approach to SQL Queries (Q9)	76	70%	56%	-	-	-	-	70%	61%
Experiment 13 – Attention to Detail	94	59%	28%	94%	17%	-	-	-	-
Experiment 14 – Key Phrases	95	0%	-	85%	0%	50%	-	0%	-

Experiment 1: Logic Rules

For experiment 1, depending on the participant's path through the tutorial, learning styles were incremented by logic rules mapped to the FS model (described in Chapter 5). Figure 8.4 shows a comparison of the Oscar PCITS prediction accuracy for this experiment with the prior probability across the sample. It can be seen from the results that using this measure alone the Oscar PCITS is better at predicting learning styles than the prior probability for just three of the eight learning styles.

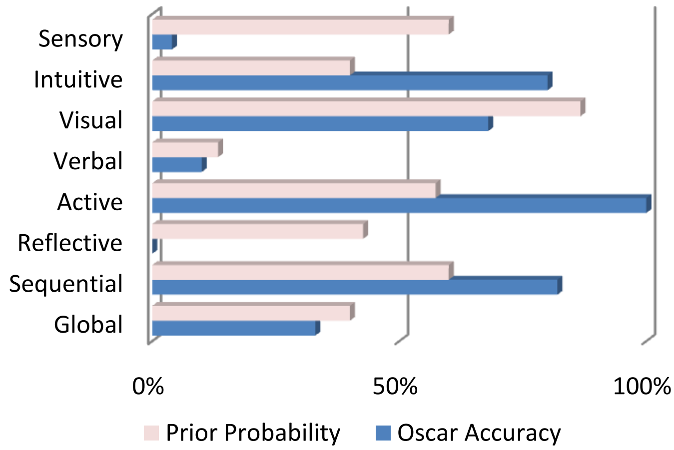


Figure 8.4. Experiment 1 Logic Rules Results

When compared to the ILS questionnaire results, Oscar accurately predicted an Intuitive learning style in 80% of cases, however it was not possible to predict a Sensory learning style, with an accuracy of only 4%.

For the Visual learning style, even though Oscar PCITS accurately predicts Visual participants in 68% of cases, the unequal spread of participants for this dimension means that this is worse than the prior probability of 87%. Oscar PCITS predicts Verbal participants in 10% of cases (compared to a prior probability of 13%). The very unequal spread of participants for the Visual/Verbal dimension means that it will be difficult to draw firm conclusions for this dimension.

Oscar PCITS accurately predicted an Active learning style in 100% of cases; however it was not possible to predict a Reflective learning style using this measure. The characteristics of reflective learners described in the FS model suggest that they spend time after learning to reflect on what they know and put it together as knowledge. As this activity happens after learning, it may not be possible to predict a reflective learning style during a tutorial. However, these results are not intended to be taken in isolation, and the combination of different analyses may improve accuracy.

Sequential participants were predicted with an accuracy of 82%, however Oscar PCITS was not able to predict Global participants using this method, with an accuracy of only 33%. This experiment has shown that logic rules are successful in predicting three learning styles, Intuitive, Active and Sequential, when compared to the prior probability. The Visual learning style prediction of 68% is not significant when compared to the prior probability.

Experiment 2: Extended Logic Rules

In experiment 2, some of the logic rules used to predict learning style in Experiment 1 were extended so that a learning style is only incremented where a particular style of resource has been shown if the participant then got the right answer directly afterwards.

The only learning style whose prediction accuracy improved was the Verbal learning style, where the prediction accuracy rose to 30%. Although very low, compared to the prior probability of 13% this accuracy is significant, but as previously stated, the spread of learning styles for the Visual/Verbal dimension across the sample is so uneven that no firm conclusions may be drawn from this data. The accuracy of predicting all other learning styles remained the same or was lower than in Experiment 1. Therefore we can conclude that using this measure of extended logic rules does not improve prediction accuracies overall.

Experiment 3: Tutorial Question Style

This experiment considers a participant's success in answering different styles of tutorial questions, and predicts a learning style depending on whether the participant is more successful in practical or theoretical questions. Participants who do better in theoretical questions are predicted to be Reflective and Intuitive, and in practical questions Active and Sensory. There were 70 participants who showed a preference for practical or theoretical tutorial questions; those participants whose success was the same for both question styles remained unclassified. Oscar PCITS was unable to predict the Sensory learning style, with an accuracy of just 36%. Intuitive participants could only be predicted in 50% of cases, but this was more accurate than the prior probability of 40%. Active participants could not be predicted, with an accuracy of 53%, but the Reflective learning style could be predicted in 73% of cases. The results of this measure show that it was the most successful factor in predicting the Reflective learning style, with the accuracy of 73% being far better than the prior probability of 43%.

Experiment 4: MCQ Question Style

Similar to experiment 3, this experiment considers a participant's preference for different styles of tutorial questions using improvements in the related MCQ test questions to predict learning style. Participants who do better in theoretical questions are predicted to be Intuitive and Reflective, and in practical questions Sensory and Active. There were 56 participants who showed more improvements in test questions related to a particular style of tutoring question; the remaining participants were unclassified. Figure 8.5 shows a comparison of the results of Experiments 3 and 4 with the prior probability. The results for this experiment are worse than Experiment 3 for predicting Intuitive and Reflective participants (those who do better in theoretical questions), but the prediction accuracies of 48% and 45% respectively are still slightly higher than the prior probabilities of 40% and 43% respectively. The Oscar PCITS prediction accuracy for the Active learning style of 52% is slightly worse than Experiment 3 and lower than the prior probability. Although the Oscar PCITS prediction of Sensory participants in this experiment is nearly double that of Experiment 3, at 60% it is still only the same as the prior probability. The results indicate that although this measure can predict the Intuitive and Reflective learning styles more accurately than the prior probability, the predictions are less accurate than using the similar measure in Experiment 3.

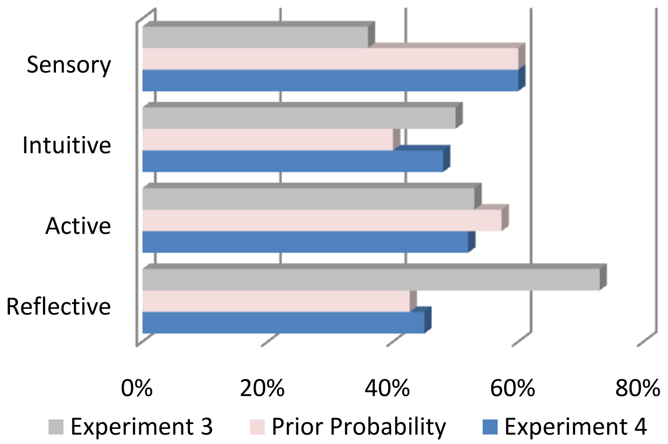


Figure 8.5. Comparison of Experiments 3 and 4 Results

Experiment 5: Word Count

This experiment compares the participant word count from the tutorial conversation to the mean and median across the sample to predict a Visual or Verbal learning style. The Oscar PCITS prediction accuracies of 57% for Visual participants and 30% for Verbal participants both resulted from a comparison with the sample mean word count. A further set of results of this experiment, shown in Table 8.4 as Experiment 5a, were gathered by analysing the subset of participants who completed the tutorial in a single session. During Study 3 in particular, it was observed that only around half of participants completed the tutorial in one session. As this represented a large group, it was decided to investigate whether prediction accuracies for this subset were different to those across the whole sample. For Experiment 5a, prediction results across the whole sample were slightly different, with 55% of Visual and 40% of Verbal participants being predicted successfully (compared to the prior probability for the sample of 90% for Visual and 10% for Verbal learning styles). However for Study 3, predictions were better for those participants who completed in a single session, with 62% of Visual and 50% of Verbal participants being predicted successfully. These comparative results are presented in Figure 8.6, along with the prior probability figures, which were approximately the same for Study 3 as for the collated figure for all studies.

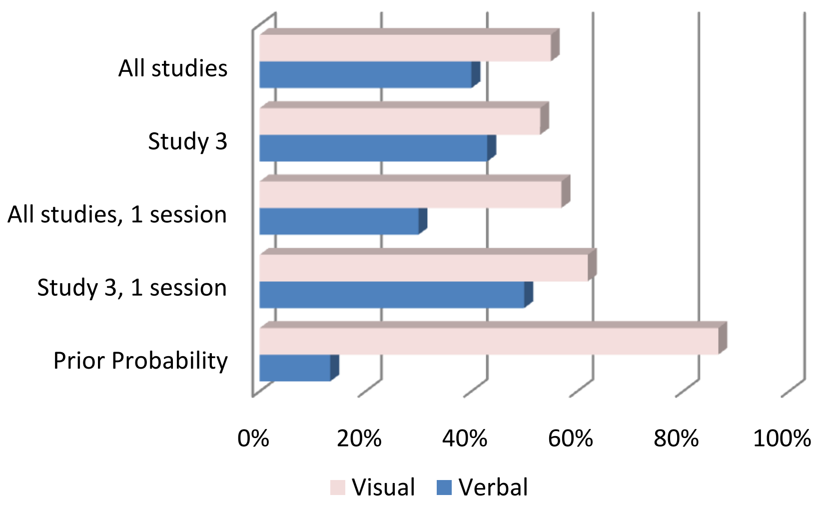


Figure 8.6. Experiment 5 Results

The best accuracies for predicting Visual participants are lower than the prior probability, and although the Verbal predictions are higher than the prior probability they are still very poor. This may be because each participant's tutorial has been personalised depending on their dialogue during the tutorial and prior knowledge of the subject, so tutorials are of different lengths and learning style is not the only differentiating factor. Interestingly, the mean word count for Visual learners (168.3) was higher than that for Verbal learners (136.1), so there is no evidence to support Hypothesis H4. The results may also be affected by the uneven distribution of this learning style dimension, with 87% of participants having a Visual learning style.

Experiment 6: Number of Interactions

For experiment 6, the number of tutoring discourse interactions was counted for each participant and compared to the mean and median values across the sample to predict a Visual or Verbal learning style. The prediction accuracies of 48% for Visual and 20% for Verbal learners indicate that this measure has not been successful in predicting learning styles. This may result from the different learning paths taken by each participant during the tutorial, depending on their dialogue and prior knowledge of the subject, so tutorials are of different lengths and learning style is not the only differentiating factor. Also, the uneven spread of Visual/Verbal learning styles across the sample may be another contributing reason. Like experiment 5, the mean number of interactions for Visual learners (49.5) is higher than that for Verbal learners (44.8), offering no evidence to support hypothesis H4.

Experiment 7: Word Count per Interaction

In experiment 7, each participant's word count per interaction was compared to the sample mean and median word count per interaction to predict a Visual or Verbal learning style. The prediction accuracies of 60% for Visual and 60% for Verbal participants, when compared to the prior probability of 87% and 13% respectively indicate that this measure is successful in predicting Verbal learners, but not Visual learners.

To investigate whether the method of calculating word count per interaction (using session totals) was affecting the result, more detailed analysis was done. A list of the student word count for every interaction was produced for each participant, which was used to calculate the mean, median and mode word count per interaction. These figures were compared to the sample averages (mean and median) and the prediction accuracies recalculated. The additional analysis had no effect on the Visual prediction results, but the Verbal prediction accuracy increased to 70% when comparing the mean participant word count per interaction with the sample median. Interestingly, the mean word count per interaction calculated for the set of Visual participants (mean 2.85, median 1.69) was slightly higher than for the set of Verbal participants (mean 2.59, median 1.2) but not significantly so. The results suggest that this measure can predict Verbal learners, but the uneven distribution of the Visual/Verbal participants in the experiment mean that no firm conclusions may be drawn.

Experiment 8: FAQ Count

In experiment 8, the number of FAQs asked during the tutorial was counted for each participant and compared to the mean and median values across the sample to predict a Visual or Verbal learning style. The prediction accuracies of 60% for Visual and 20% for Verbal learners show that FAQ count is not a successful measure for predicting learning styles. On the whole, the level of discussion with the PCITS was very low, with the median number of FAQs asked being 0 and the mean only 0.69. This appears to be a cultural problem, with observations noting that participants habitually ‘googled’ the answers to questions rather than asking the Oscar PCITS.

The participant who asked the most questions (8 FAQs) was a Visual learner (although with an ILS strength of only 1), as were the two participants who asked the second highest number of 4 FAQs. Indeed, the mean FAQ count for Visual learners, at 0.77, was far higher than that for Verbal learners of 0.2. Although this result is most likely caused by the uneven distribution of Visual/Verbal learners, there is no evidence to support the hypothesis that Verbal learners can be predicted from the amount of discussion measured by the number of FAQs asked.

Experiment 9: Tutorial Duration

In experiment 9, the duration of each participant’s tutoring session was compared to the mean and median duration across the sample to predict a Sensory or Intuitive learning style. The prediction accuracies of 56% for Sensory and 67% for Intuitive learners indicate that this measure was not successful in predicting learning styles. During analysis, it was observed that for Study 3 half of all participants (22 participants) did not complete the tutorial in one session, i.e. during the timetabled laboratory. All participants in the other studies completed the tutorial in one session. The 23 participants who completed the tutorial in a single session were analysed separately, and the prediction of Sensory learners in this group was more accurate at 67%.

The results highlight the inherent problem with comparing duration when each participant's tutoring session follows an individual path and so it is not possible to compare like with like.

Experiment 10: Time per Interaction

In experiment 10, each participant's time per interaction was compared to the sample mean and median to predict a Sensory or Intuitive learning style. The prediction accuracies of 49% (Sensory) and 60% (Intuitive) suggest that this measure was successful in predicting Intuitive learners.

To investigate whether the method of calculating average time per interaction (using session totals) affected the result, more detailed analysis was done. For each participant, the actual time for each interaction was listed and the mean, median and mode calculated. These figures were then compared to the sample average time per interaction. The additional analysis resulted in poorer prediction for Sensory learners, but better prediction of Intuitive learners, with an increased accuracy of 67%.

Experiment 11: Reading Time

For Experiment 11, the mean and median time taken to read 10 Oscar words was calculated for each participant and compared to the mean and median values across the sample to predict a Sensory/Intuitive and Visual/Verbal learning style. Compared to the prior probability the results indicate that the measure is successful in predicting the Intuitive (63%) and Verbal (50%) learners who have below average reading time, although the highest prediction accuracy achieved (72%) is for Visual learners.

The sample mean reading time (12.1) differed from the median (10.37) by almost 2 seconds. As each participant's learning path is individual, different numbers of Oscar words will be presented, however the indication is that the median is the most appropriate measure for comparison in this case.

To overcome the problem of comparing reading time for different learning paths more detailed analysis was undertaken to compare the reading time of a single piece of text. The only text that could be guaranteed to be identical for every participant was at the start of the tutorial. Therefore Experiment 11b compares the reading time of tutorial Question 1 to the mean and median values across the sample to predict a Visual or Verbal learning style. Reading time was calculated for 95 participants who had completed Question 1. The results were mixed, with poor predictions of Intuitive and Visual participants (those with a below average reading time) but good predictions of Sensory and Verbal participants (those with above average reading times). The prediction accuracies for the Intuitive (78%) and Verbal (71%) learning styles are much higher than the prior probabilities of 40% and 13% respectively. The results show that this measure is the best predictor of Verbal learning style, thus supporting the hypothesis H2.

Experiment 12: Approach to SQL Queries

This experiment predicted learning styles depending on a participant's approach to writing SQL queries. As there are two questions of this style, Table 8.4 reports results for tutorial question 5 as Experiment 12a and those for question 9 as Experiment 12b. This experimental analysis requires the completion of tutorial question 5 or 9 rather than the whole tutorial session, so the sample for 12a included 89 participants, and for 12b there were 76 participants. For the Sensory/Intuitive learning style dimension, the best predictions came from Experiment 12b, with 70% and 56% respectively, compared to 65% and 38% from Experiment 12a and the prior probability of 60% and 40%. For the Sequential/Global learning style dimension the results are mixed, with the best prediction for Sequential of 74% resulting from Experiment 12a, but the best prediction for Global of 61% resulting from Experiment 12b. However, all prediction accuracies for the Sequential/Global dimension compare favourably to the prior probability of 60% and 40%. Therefore the results from this experiment seem to support the hypothesis H7.

Experiment 13: Attention to Detail

This experiment predicts learning style depending on a participant's answer to a 'trick question'. As this experimental analysis relies on tutorial question 4 rather than a completed tutorial session, it included a sample of 94 participants. For the Sensory/Intuitive learning style dimension, the prediction accuracies of 59% and 28% are worse than the prior probability for the sample of 62% and 38% respectively. However, predictions for the Visual/Verbal learning style dimension were better than the prior probability at 94% and 17% respectively, and this measure produced the most accurate prediction overall for the Visual learning style. Therefore the results support hypothesis H6, a lack of attention to detail in answering questions is indicative of learning style.

Experiment 14: Key Phrases

This experiment predicts learning styles in the event of key words or phrases from a predefined list being uttered by a participant. As this experimental analysis does not require a complete tutorial session, the sample group contained 95 participants. 41 participants (43% of the sample) uttered at least one of the key phrases during the tutorial conversation, however only six of the list of 138 key words/ phrases (4%) were uttered. In Table 8.4, the prediction accuracies presented are the mean accuracy for each learning style and not the highest accuracy, as this would be misleading. Table 8.5 lists the results for each key phrase uttered during tutorials, and their accuracy in predicting the related learning style. In Table 8.5, *n* refers to the number of participants who uttered the key phrase. It can be seen that the key word ‘show’, which relates to the Visual learning style, is the most uttered with 32% of participants uttering the word (between one and four times, totalling 53 times during the tutorial). The 85% prediction accuracy for the key word ‘show’ is equal to the prior probability across the sample, as there is such an uneven distribution of students along the Visual/Verbal dimension. The frequency of key word utterances was normally one (30 participants, 73%), and therefore bore no relation to the prediction of learning styles. As the number of key phrases uttered is so small, it is not possible to draw any conclusions from this measure, and therefore the results do not support hypothesis H1.

Table 8.5. Experiment 14 Results: Key Phrases

Key Phrase	n	Sensory	Visual	Verbal	Active	Sequential
Prior Probability	*95*	*62%*	*85%*	*15%*	*59%*	*60%*
see	1	-	100%	-	-	-
show	39	-	85%	-	-	-
picture	1	-	100%	-	-	-
tell	2	-	-	0%	-	-
discuss	1	-	-	0%	100%	-
example	1	0%	-	-	0%	0%

It was observed that participants were reluctant to enter into discussion with Oscar PCITS, with a median of only 156 words uttered during the tutorial. This lack of discussion meant that there was a smaller chance of key words or phrases being uttered by participants. The lack of discussion may be due to the culture of Engineering students, with 85% of participants having a Visual learning style (Garcia et al. 2007 reported a similar observation, with 79% of participants making no use of forum and chat facilities). Alternatively the lack of discussion may be related to the subject domain.

4.3 Participant Evaluation

In general, the user feedback from the studies showed that Oscar was well received, understandable and helpful. 46 participants across all three studies completed the feedback questionnaire, with 10 participants from study 1, 10 participants from study 2 and 26 participants from study 3. Table 8.6 gives the collated results for questions 1 to 9 (three open questions will be described separately).

Table 8.6. Participant Evaluation Questionnaire Results

Please rate your experience of the following using the scale provided:	SCALE					
	High 6	5	4	3	2	1 Low
1. Instructions	42%	31%	16%	9%	0%	2%
2. Screen layout and design	37%	28%	24%	0%	4%	7%
3. Tutoring	51%	20%	16%	7%	4%	2%
4. How well did Oscar understand you?	37%	15%	22%	11%	11%	4%
5. Did you find the tutoring helpful?	72%	7%	15%	2%	0%	4%
6. Was the conversation natural?	43%	22%	15%	9%	7%	4%
7. Was the conversation frustrating?	33%	7%	13%	9%	20%	20%
8. Do you feel Oscar helped you to revise?	59%	15%	11%	4%	4%	7%
9. Would you use a resource like Oscar:						
a. Instead of attending a face-to-face tutorial?	Yes	35%	No	65%		
b. Instead of learning from a book?	Yes	52%	No	48%		
c. As well as classroom tutoring?	Yes	85%	No	15%		
d. Would you use the resource at all?	Yes	89%	No	11%		

It can be seen from Table 8.6 that 87% of participants rated the tutoring highly (Question 3), with 51% awarding the tutoring the highest rating of 6. In Question 5, 94% of participants found the tutoring helpful, with 72% giving the highest rating of 6. The only negative responses (i.e. rated 1 or 3) for Question 5 came from study 3 – 100% of participants in studies 1 and 2 thought that the tutoring was helpful. This may again be explained by the level of motivation of participants, as in study 3 participants were not revising SQL for assessment, and so may not have appreciated any direct benefit. In Question 6, 80% of participants rated the tutoring conversation as natural, however in Question 7, 52% of participants found the conversation frustrating. In Question 8, 85% of participants felt that Oscar PCITS had helped them to revise. Of those seven participants (15%) who did not feel that Oscar PCITS helped them to revise, five (11%) came from study 3, with one (2%) each from studies 1 and 2.

The results from Question 9, which investigates whether participants would choose to use a resource like the Oscar PCITS, are interesting, with an astounding 35% of participants stating that they would use Oscar PCITS tutorial instead of attending a face-to-face tutorial. Surprisingly, for study 3 the answers to this question were almost equal, with 46% of study 3 participants stating that they would use Oscar PCITS instead of attending face-to-face classes. Slightly more than half of the participants (52%) stated that they would use Oscar PCITS instead of reading a book, and 85% of participants would use Oscar PCITS to support classroom tutoring. Overall, 89% of participants stated that they would use a resource like Oscar PCITS if it were available. From these results it can be concluded that most people found the Oscar PCITS tutoring helpful, and would use Oscar PCITS to support their studies.

The three remaining questions on the feedback questionnaire were open questions, asking for the participants to state what else could be included to assist in learning, three positive and three negative points about using Oscar. Where possible, the answers were grouped into categories, as reported in Table 8.7. Note that in Table 8.7 *n* is the number of participants who answered the question, and not the number of answers given.

Table 8.7. Open Question Results

	n	%
13. What else could Oscar have done to help you learn?	44	96
Nothing – it's great	14	32
More resources (examples, movies, pictures)	8	18
More questions	3	7
14. Please state 3 positive points about using the Oscar computer tutor	46	100
Easy to use	23	50
Helpful	20	43
Step-by-step guidance	18	39
Flexible	11	24
Fast	11	24
Hints	8	17
15. Please state 3 negative points about using the Oscar computer tutor	45	98
Oscar doesn't always understand	13	29
GUI	12	27
Questions difficult to understand	6	13

When openly asked for comments, half of the group commented that the conversational interface was natural and easy to understand and 43% found Oscar's tutoring helpful. 39% of respondents found the step by step guidance positive and 17% thought the hints given were good, with one learner remarking "*it encouraged me to think rather than simply giving me the answer".* When asked for negative comments, 29% of respondents noted that Oscar did not always understand their input and 13% found the tutoring questions sometimes difficult to understand. Some quotes from the open questions are shown below:

- "doesn't just give you the answer straight away"
- "you could do it without the need for a tutor to be present"
- "can work at your own pace"
- "can revise at any point (day/night)"
- "would stop academics being asked basic questions!"
- "easy to understand, helpful tips"
- "breaks down questions"
- "you can just say you don't know"
- "easier than reading sql code from a book"
- "non-judgemental"
- "interactivity, real-time, social"
- "impersonal and slightly cold way to revise"
- "is like having your own friendly tutor"
- "I did learn something therefore would find it helpful"
- "simple feedback can be given to each student where an instructor can only deal with a single student"
- "no choices about topics to cover"
- "good at understanding of language"
- "doesn't always understand me"

4.4 General Observations

During the timetabled laboratory sessions, it was possible to unobtrusively observe the way that participants interacted with the Oscar PCITS. It was noticed that participants habitually opened a new browser window and 'googled' the answers to questions rather than asking the Oscar PCITS. The Oscar PCITS is designed to direct an individual, personalised tutorial and so no conclusions can be drawn about whether the lack of engagement in discussion with Oscar PCITS would occur when participants complete the tutorial alone. Such distractions are to be expected in real learning environments and it would appear that the availability of the Internet and its quick and easy access to a wealth of knowledge has changed the culture of learning.

For study 3, it was observed that half (23) of all participants did not complete the tutorial in a single session. Separate analysis of this group revealed several marked improvements in prediction accuracies (e.g. 100% of Sensory learners were predicted by Experiment 4, 100% of Verbal learners were predicted by Experiments 9 and 12b). However, when analysing separately the group of participants completing in more than one session across all three studies, most prediction accuracies were not significantly higher.

4.5 Additional Analysis

Further analysis related to (but not part of) the reported results was carried out. The prediction accuracies were calculated for the subset of participants who had a strong learning style preference (i.e. with an ILS score of 5 or more) as in Cha et al. (2006). When considering the Neutral group (i.e. participants with an ILS score of 1 or 3) separately, prediction accuracies for the Neutral group ranged between 58-88% for Experiment 2. However, removing the group of Neutral learners resulted in a much smaller number of participants (e.g. only 28 Sequential/Global learners) and the prediction accuracy results were lower.

Experiments have been undertaken (which are not part of the work submitted in this thesis) to use the behaviour captured during a tutoring conversation to build a fuzzy classification tree for two FS dimensions (perception (sensory-intuitive) and understanding (sequential-global)). Early results show that the model has increased the predictive accuracy of the Oscar CITS and discovered some interesting relationships amongst the variables (Crockett et al., 2011).

5 Experimental Results Summary

The experimental results supported six of the seven hypotheses, as follows:

- H2: the time taken for a particular tutor-learner interaction is indicative of learning style – supported by experiment 10 and 11 results.

- H3: the success of a learner in a particular style of tutoring question is indicative of learning style – supported by experiment 3 and 4 results.
- H4: the amount of discussion a learner enters into with the tutor is indicative of learning style – supported by experiment 7 results.
- H5: the success of a learner after experiencing a particular style of tutoring is indicative of learning style – supported by experiment 1 results.
- H6: a lack of attention to detail in answering questions is indicative of learning style – supported by experiment 1 and 14 results.
- H7: choosing to be guided through a process (or not) is indicative of learning style – supported by experiment 1 and 12 results.

There was no evidence supporting hypothesis H1 (certain key phrases uttered during a CA tutoring conversation is indicative of learning style). As described in Section 4.2 Experiment 14, there was a general lack of discussion with the Oscar PCITS and although 43% of the sample uttered key words, 73% uttered only a single key word. Analysis of tutoring dialogues failed to show any commonly uttered key words or phrases that could extend the set. As the number of key phrases uttered is so small, it was not possible to draw any conclusions.

The experimental results show that all eight learning styles could be successfully predicted with better accuracy than the prior probability (Table 8.8). Therefore, the main hypothesis (It is possible to estimate a learner's learning style from a two-way tutoring discourse with a conversational agent (CA) tutor) is supported. However, it must be borne in mind that the uneven distribution of participants for the Visual/Verbal dimension prevents firm conclusions from being drawn.

Table 8.8. Best Learning Styles Predictions

	Prior Probability	Prediction Accuracy	Experimental Measure
Sensory	*60%*	70%	Experiment 12b Approach to Queries (Q9)
Intuitive	*40%*	80%	Experiment 1 Logic Rules
Visual	*87%*	94%	Experiment 13 – Attention to Detail
Verbal	*13%*	71%	Experiment 11b – Q1 Reading Time
Active	*57%*	100%	Experiment 1 Logic Rules
Reflective	*43%*	73%	Experiment 3 – Tutorial Question Style
Sequential	*60%*	82%	Experiment 1 Logic Rules
Global	*40%*	61%	Experiment 12b Approach to Queries (Q9)

A comparison of results with other CITS is not possible as no other CITS can predict learning styles.

On a superficial level, the results compare favourably with menu-based ITS that predict FS learning styles (Popescu, 2009; Ozpolat and Akar, 2009; Garcia et al., 2007; Graf, 2009). However it is inappropriate to compare prediction accuracies with these ITS because the method of tutoring is markedly different, e.g. some behaviour analysed relies on student initiative, such as the selection or navigation of particular resources from a menu. Also, despite adopting the FS these ITS classify learning styles differently, introducing a third 'Neutral' class for each dimension that describes learners with low strength learning styles (i.e. those at the centre of the dimension). The method of calculating prediction accuracy for these ITS uses different scoring, by awarding a 0.5 score if the learning style prediction is mismatched with a Neutral classification, rather than a zero score for all mismatches used by Oscar PCITS. In the case of Ozpolat and Akar (2009) this method of scoring is particularly misleading as, although they report accuracies in the range of 53.3-73.3%, their results show that their method only managed to classify learners as Visual or Neutral across all dimensions.

Some results do not include the whole student group. In iLessons (Sanders and Bergasa-Suso, 2010), which analyses student navigation and interaction with the Internet to infer learning styles, 'dead bands' were introduced for 'unknown' classifications, and these learners were removed from the results. In Cha et al. (2006), all Neutral learners (i.e. those scoring 1 or 3 on the ILS questionnaire) are removed from the analysis, and so the results are only based on learners with a moderate to strong preference. However, Neutral learners are a large group who still need to be identified, and reducing the number of participants in this way (e.g. Cha et al., 2006 analysed only 23 of 70 participants for the Active/Reflective dimension) diminishes the validity of the results.

6 Conclusion

This chapter has described the experiments conducted to validate the Oscar PCITS methodology and architecture proposed in Chapter 6. Oscar PCITS was implemented to direct an SQL revision tutorial to undergraduate students and to predict FS learning styles during the conversation, as described in Chapter 7. Three studies were described, including an initial pilot study of ten participants followed by two larger uncontrolled studies of 105 participants. The uncontrolled studies involved real students in a real teaching/learning environment. The success of Oscar PCITS was evaluated in three ways, to investigate:

1. Can Oscar predict learning styles from a two-way tutoring discourse?
2. How do participants rate Oscar PCITS and would they use it?
3. Does Oscar PCITS successfully tutor participants?

To evaluate the prediction of learning styles, fourteen experiments were devised that tested seven hypotheses. The results supported six of the seven hypotheses, and showed that Oscar PCITS was successful in predicting all eight FS learning styles in a real teaching/learning environment. Additionally, 89% of participants said they would use Oscar PCITS if it were available. Participants did learn from the tutorial, improving test results by an average of 13%, although it is recognised that any revision activity is likely to lead to learning gains.

Participants valued Oscar PCITS as it offers a personalised tutorial online which can be done at their own pace, accessed at any time and place. Participants appreciated Oscar PCITS's intelligence in the instant feedback, problem solving support and hints, and the fact that they could repeat the tutorial until they fully understood the topic. However, there was a cultural problem in asking Oscar PCITS for help, participants instead searching the Internet for an answer. This may result from the novelty of the CITS, or the fact that most participants were Visual learners, who may not naturally ask questions or open discussions. A future prototype could consider removing access to the Internet to attempt to force more interaction.

The uneven distribution of undergraduate learners on the Visual/Verbal FS dimension, as noted in the FS model (Felder and Silverman, 1988), begs the question whether it is important for a PCITS to classify this dimension. 87% of participants in the studies were Visual learners, and so although Oscar PCITS predictions were more accurate (at 94% for Visual, 71% for Verbal), if adopting the FS model the importance of predicting Visual/Verbal learning styles should be considered carefully.

It can be concluded that the experimental studies described in this chapter successfully validate the generic Oscar PCITS Methodology and Architecture proposed in Chapter 6 in a real educational setting. However, before conclusions may be drawn about non-computing subject domains it is necessary to implement

Oscar CITS and empirically test its prediction of learning styles with different learning style models.

The next stage of the research is to produce a methodology for developing a CITS that can adapt to an individual's learning style while directing a tutoring conversation. The next two chapters will propose such a methodology and present the implementation and experimental analysis of the resulting CITS.

7 Chapter Highlights

- Three real-world studies were undertaken to validate the Oscar PCITS methodology and architecture proposed in Chapter 6.
- The studies involved 115 participants, resulting in 75 completed tutorials and 46 completed evaluation questionnaires.
- 14 experiments investigated the prediction of all four FS learning style dimensions. The results showed that all eight learning styles were predicted successfully in a real teaching/learning environment.
- The results showed a mean learning gain of 13% in test scores following the Oscar PCITS tutorial.
- User feedback results showed that 87% of participants rated the tutoring highly, 94% found the tutoring helpful and 89% stated that they would use a resource like Oscar PCITS if it were available.

Chapter 9 A Methodology and Architecture for Developing an Adaptive CITS

1 Introduction

Chapter 4 described how Intelligent Tutoring Systems personalise tutoring by adapting to individual factors such as existing knowledge (Weber and Brusilovsky, 2001), emotion (D'Mello et al., 2009) and learning styles (Sanders and Bergasa-Suso, 2010). Conversational ITS (CITS) are less common than menu-based ITS, and there are no CITS that adapt their teaching to suit an individual's preferred learning styles. In Chapter 6, the *Oscar CITS* was proposed, which is a CITS that can automatically predict and adapt to learning styles during a tutoring conversation. Chapters 5, 6, 7 and 8 described the design, development and experimental analysis of *Oscar Predictive CITS (PCITS)*, which was successful in predicting learning styles from a natural language tutorial. The generic methodology and architecture of Oscar PCITS (proposed in Chapter 6) were validated empirically and Oscar PCITS was also successful in improving learning. As described in Chapter 4, a CITS that can adapt its tutoring style to match student learning styles could lead to an enhanced learning experience and higher learning gains.

In this chapter, the achievements of the Oscar PCITS are used as a basis for designing a CITS that dynamically adapts its teaching style to match student learning styles. This chapter proposes an original methodology and architecture for developing the Oscar Adaptive CITS (ACITS) that can automatically adapt to a student's learning style during the tutoring conversation. The proposed methodology and architecture are generic, with the flexibility to adopt different learning styles models and tutoring domains. The adaptive CITS construction is based around delivering preferred teaching styles for individual learners during tutoring. An original generic adaptation algorithm is proposed which takes into account both the strength of each learner's preference and the availability of adaptations to select the best adaptation for each tutorial question.

2 Oscar Conversational ITS

As described in Chapter 6, the Oscar CITS is a novel CITS that can both predict and adapt to an individual's preferred learning style whilst holding a tutoring conversation. The pedagogical aim is to enhance the learning experience by providing the learner with tutoring material suited to their learning styles. Although the nature of learning styles has been questioned by some authors (Coffield et al., 2004a), others have found that matching teaching styles to learning styles can lead to greater learning (Felder and Brent, 2005; Allinson and Hayes, 1996). Studies in computer-based education systems have shown that adapting the teaching style to student learning styles can enhance learning (Kelly and Tangney, 2006; Tsianos et al., 2008; Walters et al., 2000).

Oscar CITS aims to mimic a human tutor by leading a tutoring dialogue and employing intelligent techniques to assist learners in constructing a deeper understanding of the topic. Oscar's natural language interface is familiar and intuitive to learners, helping to build confidence and motivation. Oscar CITS is available online, allowing learners to study when and where they choose at a fixed cost.

3 Devising the Adaptation Strategy

The *Oscar Adaptive CITS (ACITS)* is a CITS that incorporates the automatic adaptation of a tutoring conversation to an individual's learning styles. Although no CITS adapts to learning styles, there are a number of approaches used by ITS to adapt to student learning styles, as reviewed in Chapter 4, section 3.2. The simplest and most common adaptation strategy is to adapt to a single aspect of learning style (Sanders and Bergasa-Suso, 2010; Bajraktarevic et al., 2003). The disadvantage of selecting only one learning style is that this does not truly describe the student's preference for each step of the learning process, e.g. in the Felder-Silverman (1988) (FS) model there are four learning style dimensions. Therefore only a partial adaptation could be achieved. There are also difficulties when adapting to more than one aspect of learning style:

- The requirement for multiple versions of learning resources to suit different learning styles is a barrier to the development of an ITS (see Chapter 4). For example, in the FS model there are 32 separate learning styles.
- In real life it may not be appropriate or possible to incorporate every category of tutor material into every tutorial question, i.e. provide adaptations for every learning style, and still present a coherent learning experience.

- To overcome these problems, one strategy is to adapt to the strongest student learning style thus reducing the number of adaptive resources required, e.g. the FS model has four dichotomous dimensions, so eight versions of adaptive resource would be needed. However, adaptation is still only partial, and although reduced the development barrier still exists. Also, additional strategies are required to deal with conflict (when preference for several learning styles is equally strong).

It was thus concluded that the Oscar Adaptive CITS adaptation strategy needed to consider not only the strength of the student's learning style but also the availability of adaptation for each individual tutorial question. This would mean that students could be presented with the most appropriate tutoring material by considering all of their preferred learning styles and also the available adaptations for each question in the tutorial. The implication of such an adaptation mechanism for students is that they will be presented with the best fitting tutoring material whilst still encountering a coherent learning experience. Students may also experience more than one learning style adaptation during a tutorial, and such a variation in style should help to improve their interest and motivation. The implication of the selected adaptation strategy for developers is that tutorials that contain adaptations for only some of the learning styles can still be successfully used whilst the learning material is built up over time. As well as addressing an important development barrier by allowing partially complete adaptations to be used, such an approach allows tutorials to be tested and changed at an early stage before too much time has been committed to the development.

The Oscar ACITS adaptation approach is considerably more complex than adapting to a single aspect of a student's learning style, requiring an algorithm for conflict resolution when more than one learning style dimension wins. However, the Oscar ACITS adaptation approach aims to offer a more rounded and complete learning experience whilst still presenting a coherent tutorial.

A generic methodology and architecture for the construction of Oscar ACITS will now be proposed.

4 A Generic Methodology for Creating an Oscar Adaptive CITS

In Chapter 6, a generic three phase methodology was proposed for constructing an Oscar Predictive CITS that could predict learning styles whilst directing a natural language tutoring conversation. That methodology was followed to develop a prototype Oscar PCITS whose ability to predict learning styles and teach was tested empirically (Chapters 7 and 8). Thus validated, the methodology proposed in Chapter 6 has been used as a basis for developing a new methodology to construct an Oscar Adaptive CITS that can dynamically adapt its tutoring to suit student learning styles.

The proposed methodology is independent of the learning styles model and subject domain and consists of three phases, as shown in Table 9.1. Phase 1 instructs the creation of a Learning Styles Adapter module, and phase 2 the design of the adaptive tutoring conversation. Phase 3 incorporates the learning styles adapter module and tutorial conversation into an ACITS architecture. Each phase will now be described.

Table 9.1. 3-Phase Methodology for Creating Oscar Adaptive ACITS.

Phase 1: Create the Learning Styles Adapter Module

1.1. Select a Learning Styles Model and extract the behaviour characteristics
1.2. Map learning style behaviour to associated conversational tutoring style
1.3. Map learning styles to teaching material categories
1.4. Implement the generic adaptation algorithm for chosen learning styles model

Phase 2: Design a Tutorial Conversation

2.1. Capture the tutorial scenario and questions (including movies, voice, images, examples, etc.) from human tutors in a specific domain
2.2. Determine the conversational structure/style
2.3. Map tutorial questions onto the generic teaching material categories
2.4. Score tutorial questions for adaptation to each learning style
2.5. Script Conversational Agent natural language dialogue for each tutorial question using the 3-level model

Phase 3: Construct the ACITS Architecture

4.1 Phase 1: Create the Learning Styles Adapter Module

Phase 1 of the Oscar ACITS Methodology involves the analysis of a learning styles model in order to create a Learning Styles Adapter module for the ACITS. Steps 1.1 and 1.2 involve the selection and extraction of knowledge from a learning styles model, and are similar but not identical to steps 1.1 and 1.2 of the methodology for creating an Oscar Predictive CITS described in Chapter 6.

4.1.1 Step 1.1: Select a Learning Styles Model and Extract the Behaviour Characteristics

The first step in creating the learning styles adapter module requires a learning styles model (Felder and Silverman, 1988; Honey and Mumford, 1992) to be selected. To illustrate Phase 1 of the methodology, the Felder-Silverman (FS) model (Felder and Silverman, 1988) was selected as the initial experimental group will be university engineering students.

Once the model has been chosen, the typical learner behaviour characteristics described in the model must be extracted for each learning style. For the FS model, the behaviour characteristics were extracted and summarised in a table of common learner behaviour (as described in Chapter 5, shown in Table 5.3).

4.1.2 Step 1.2: Map Learning Style Behaviour to Associated Conversational Tutoring Style

To map learning style behaviour to the conversational tutoring style, first assess each behaviour characteristic extracted in step 1.1 to see if it can be mapped onto a two-way online conversational tutorial. If so, the behaviour trait should be included in a summary table, as described in Chapter 5, section 4 (see Table 5.4).

Next, each behaviour trait in this subset must be mapped to the associated teaching style. In the exemplar FS model, teaching styles associated with typical learner behaviours are described for each learning style. This information was extracted and is summarised in Table 9.2.

4.1.3 Step 1.3: Map Learning Styles to Generic Teaching Material Categories

In step 1.3 it is necessary to decide which styles of teaching material need to be incorporated in a tutorial. Each teaching style extracted from the learning styles model in Step 1.2 (Table 9.2) needs to be studied and reorganised so that similar teaching styles are grouped together. A set of generic teaching material categories has been produced which can be expanded and mapped to the chosen learning styles model. For example using the FS model, in Table 9.2 the teaching styles associated with Global learners have been categorised in two ways:

- ‘Present overview summary information’ can be categorised into ‘Introductions and overviews’ teaching material (Table 9.3 category 1).
- ‘Avoid detail’ can be categorised into ‘Explanation by bullet points and hyperlinks’ teaching material (Table 9.3 category 3).

Table 9.2. FS Learner Behaviour and Associated Teaching Styles

LEARNING STYLE	TEACHING STYLE
Sensor	
Prefer facts, data, experimentation	Present facts, examples and results
Dislike surprises	Include introductions, overviews and present material in a sequential predictable order
Careful but slow	Consider this if timing interactions
Comfortable with symbols (e.g. words)	Textual explanations and discussions OK
Intuitor	
Prefer principles and theories	Present principles rather than examples
Dislike repetition	Present information usually only once
Bored by detail	Summarise information e.g. bullet points
Quick but careless	Consider this if timing interactions
Uncomfortable with symbols	Favour bullet points and diagrams
Visual	
Remember what they see	Present diagrams, pictures, movies
Like pictures, diagrams, flow charts, time lines, films	Favour pictures, diagrams, flow charts, time lines, films
Prefer visual demonstration	Use visual walkthroughs (e.g. results of database queries using table snapshots and selecting/moving rows visually) rather than textual explanation
Verbal	
Remember what they hear, or what they hear then say	Favour movies and sound clips
Like discussion	Use the CA to discuss the topic
Prefer verbal explanation	Favour movies, sound clips and CA explanations
Learn by explaining to others	Use the CA to discuss the topic
Active	
Do something with information – discuss/explain/test	Use the CA to discuss the topic, include practical exercises
Experimentalists	Include practical exercises
Process information by setting up an experiment to test an idea, or try out on a colleague	Use the CA to discuss the topic, include practical exercises
Reflective	
Examine and manipulate information introspectively	Do not use CA for discussion
Theoreticians	Present principles rather than examples
Sequential	
Follow linear reasoning processes	Present information in a steady progression of complexity and difficulty
Learn best when information is presented in a steady progression of complexity and difficulty	Present information in a steady progression of complexity and difficulty
Global	
Sometimes better to jump directly to more complex and difficult material	Present overview summary information and allow student to choose where to start. Avoid detail and only include full explanations when asked.

Table 9.3 lists all categories derived from the FS analysis that have been mapped to the associated learning styles. In Table 9.3 the teaching material categories are based on teaching styles, and so are independent of a particular learning styles model or subject domain. The generic teaching material categories should be expanded when different learning styles models are adopted which require different teaching styles.

Table 9.3. Generic Teaching Material Categories Mapped to FS Learning Styles

TEACHING MATERIAL CATEGORY	FS LEARNING STYLES
1. Introductions and overviews	Sensor, Sequential, Global
2. Textual explanation – theories and principles	Intuitor, Verbal, Reflective, Sequential
3. Explanation – bullet points and hyperlinks	Intuitor, Global
4. Practical examples	Sensor, Active, Sequential
5. Practical exercises	Active, Sequential
6. Graphics – movies, pictures, diagrams	Intuitor, Visual
7. Verbal – movies & sound clips	Verbal
8. Visual demonstrations (walkthroughs etc)	Visual
9. Discussion scripts – explanation, FAQs and help for each subtopic	Sensor, Verbal, Active

The generic teaching material categories in Table 9.3 represents a list of the type of tutoring material required in an ACITS that adapts to student learning styles.

4.1.4 Step 1.4: Implement the Generic Adaptation Algorithm for Chosen Learning Styles Model

The final step in phase 1 (step 1.4) is to implement an algorithm that decides the most appropriate type of adaptation to be applied for each student's individual learning path. An adaptation algorithm was developed that selects the most appropriate adaptation for each tutorial question individually. The adaptation algorithm is generic as it is independent of the learning styles model selected. The chosen learning styles model should be applied to the algorithm (shown in Table 9.4) and implemented to produce the Learning Styles Adapter component. The development of the algorithm will now be described.

In Step 1.3 of the methodology, a list of generic teaching materials categories was produced (Table 9.3). When developing an ACITS tutorial, it may not be appropriate to incorporate all categories of teaching material and still present a coherent learning experience, so some learning styles may not be addressed. Consequently a strategy of adapting to only the student's strongest learning style (like Sanders and Bergasa-Suso (2010) and Bajraktarevic et al. (2003)) may not always be possible in real life, and also a single learning style does not completely describe the student's learning preferences. Therefore it was concluded that the Oscar ACITS adaptation algorithm should consider both the strength of the student's learning styles and the strength of adaptation for each individual tutorial question. This strategy ensures that students are presented with the most appropriate tutoring material to suit all aspects of their learning style and the available adaptations for each question in the tutorial. The implication of this adaptation mechanism for students is that they are presented with the best fitting tutoring material whilst still encountering a coherent learning experience. Learners may also experience more than one type of adaptation during a tutorial, and such a variation in style should help to improve their interest and motivation. The implication of the adaptation strategy for developers is that tutorials which contain adaptations for only some of the learning styles can still be successfully used whilst the learning material is built up over time. As well as addressing an important development barrier by allowing partially complete adaptations to be used, such an approach allows tutorials to be tested and changed at an early stage before too much time has been committed to the development.

A novel adaptation algorithm was developed that selects the best fitting adaptation per question independently of the tutorial domain and the learning styles model adopted. While developing the adaptation algorithm, the FS model was used as an example learning styles model with four learning style dimensions. Although the assessment of FS learning styles results in a learning style preference being assigned for each dimension, the model suggests that the group of students who have no strong preference for a particular learning style should be given learning material including a mixture of styles. Taking this into consideration, an additional *Neutral* learning style category was introduced to group students with a low preference on a FS dimension. Such *neutral* students will be presented with a Neutral adaptation that includes a mixture of teaching material styles.

Table 9.4 shows a logical representation of the generic Oscar ACITS adaptation algorithm. Seven assumptions are stated, as follows:

1. For each learning style dimension ***d*** in the chosen model, each student's preference is described by the tuple **<*c, s*>** where ***c*** is the learning style class and ***s*** is the score representing its strength. The values of ***c*** and ***s*** are assigned by an assessment (e.g. a questionnaire defined by the learning

styles model). For the FS model the student preference is evaluated using the Index of Learning Styles (ILS) (Felder and Soloman, 1997).

2. A student is classed as *Neutral* if they have a low preference for the learning style (as defined by the learning styles model). In the FS model, neutral students are defined as having ILS learning style scores of 1 or 3 (using the ILS scale 1 (weak) to 11 (strong)).
3. Each tutorial question is assigned a score $\boldsymbol{q}$ for adaptation to each possible learning style (LS) based on the number of opportunities in the question to adapt to that learning style. For the Neutral learning style, $\boldsymbol{q}$=0. The question scoring will be described in Section 4.2.4.
4. MAX is a function that, given a list of numbers, returns the classes associated with the maximum numbers. For example, given a list of numbers (3,6,3) associated with learning style classes (vis, act, glo) MAX returns (act); given a list of numbers (3,6,6) associated with learning style classes (vis, act, glo) MAX returns (act, glo).
5. $\boldsymbol{W}$ is the set of classes of winning (i.e. maximum) student question scores $\boldsymbol{a}$.
6. $\boldsymbol{S}$ is the subset of $\boldsymbol{W}$ containing classes of maximum student learning style scores $\boldsymbol{s}$.
7. $\boldsymbol{Q}$ is the subset of $\boldsymbol{W}$ containing classes of maximum question scores $\boldsymbol{q}$.

Following this algorithm, if the student is classed as Neutral for all learning style dimensions, they follow the Neutral adaptation learning path for every tutorial question. Otherwise, for each tutorial question:

- for each learning style class $\boldsymbol{c}$ a student question score $\boldsymbol{a_c}$ is calculated by multiplying the question score for that learning style $\boldsymbol{q_c}$ with the student learning style score $\boldsymbol{s_c}$.
- The learning style class $\boldsymbol{c}$ with the highest student question score $\boldsymbol{a_c}$ wins.

Additional rules are included for conflict resolution, i.e. selecting an adaptation when more than one maximum score $\boldsymbol{a}$ exists in $\boldsymbol{W}$, the set of winning scores:

- In the case where there is no clear winner (i.e. no single maximum score $\boldsymbol{a}$), the learning style class in $\boldsymbol{W}$ with the maximum student score $\boldsymbol{s}$ wins (i.e. the student's strongest learning style preference in the set of winners).

- If there is no clear winner again (i.e. the student has more than one learning style with the maximum score s), the question scores q for the learning style classes in W are compared, and the maximum question score q wins.
- Finally, if there is still no winner, a learning style adaptation is selected randomly from the learning style classes in W.

Table 9.5 shows examples of the adaptations selected by the algorithm (applied to the FS model) and of conflict resolution.

Table 9.4. Generic Oscar ACITS Algorithm for Selecting Best Adaptation Per Question

Assuming that:

1. For each learning style dimension d in the learning styles model, each student is assigned a learning style tuple $<c, s>$ where c is the class and s is the score.
2. A student is classed as *Neutral* if they have a low preference for the learning style (as defined by the learning styles model).
3. Each tutorial question is assigned a score q for adaptation to each possible learning style (LS) based on the number of opportunities in the question to adapt to that learning style. For the Neutral learning style, q=0.
4. MAX is a function that, given a list of numbers, returns the classes associated with the maximum numbers.
5. W is the set of classes of winning student question scores a.
6. S is the subset of W containing classes of maximum student learning style scores s.
7. Q is the subset of W containing classes of maximum question scores q.

```
IF (FOR ALL d (c IS Neutral))
  THEN ADAPT TO learning style class Neutral.
ELSE
  FOR EACH tutorial question
  {
    FOR EACH d
          student question score a_c = question LS score q_c × student LS score s_c.
    W = MAX(student question scores a_c).
    IF |W| IS 1
      THEN ADAPT TO learning style class c WHERE c ∈ W.
    ELSE
    {
      S = MAX(student LS scores s_c WHERE c ∈ W).
      IF |S| IS 1
        THEN ADAPT TO learning style class c WHERE c ∈ S.
      ELSE
        Q= MAX(question scores q_c WHERE c ∈ W).
        IF |Q| IS 1
          THEN ADAPT TO learning style class c WHERE c ∈ Q.
        ELSE
           ADAPT TO RANDOM (learning style class c) WHERE c ∈ W.
    }
  }
```

In Table 9.5, first the question learning style scores (q) are shown for three tutorial questions. Next, each student's learning style dimension class and score $<c, s>$ is listed, followed by (for each tutorial question) the calculated student question scores (***a***), the set of winners (***W***) and the resulting adaptation. The strength of learning style preference is scored by the ILS as 1, 3, 5, 7, 9 or 11. Neutral learners have low preference for a learning style, i.e. ILS scores of 1 or 3.

Table 9.5. Examples Demonstrating the Oscar ACITS Adaptation Algorithm

Question Scores: Q1: NEU=0, SNS=3, INT=3, VIS=3, VRB=5, ACT=4, REF=0, SEQ=1, GLO=4 Q2: NEU=0, SNS=2, INT=5, VIS=5, VRB=3, ACT=2, REF=1, SEQ=8, GLO=3 Q3: NEU=0, SNS=0, INT=0, VIS=2, VRB=0, ACT=0, REF=1, SEQ=2, GLO=0
Student 1 Learning Style Scores: SNS=1, VIS=3, REF=3, SEQ=1 Adaptation: NEUTRAL
Student 2 Learning Style Scores: SNS=7, VIS=9, ACT=11, SEQ=9 Q1: SI=7*3=21, VV=9*3=27, AR=11*4=44, SG=9*1=9; winner=AR, Adaptation=ACT Q2: SI=7*2=14, VV=9*5=45, AR=11*2=22, SG=9*8=56; winner=SG, Adaptation=SEQ Q3: SI=7*0=0, VV=9*2=18, AR=11*0=0, SG=9*2=18; winner={VV,SG}, Adaptation=VIS
Student 3 Learning Style Scores: INT=3, VIS=3, ACT=5, GLO=5 Q1: SI=3*3=9, VV=3*3=9, AR=5*4=20, SG=5*4=20; winner={AR,SG}, Adaptation=ACT Q2: SI=3*5=15, VV=3*5=15, AR=5*2=10, SG=5*3=15; winner={SI,VV,SG}, Adaptation=GLO Q3: SI=3*0=0, VV=3*2=6, AR=5*0=0, SG=5*0=0; winner=VV, Adaptation=VIS
Key: SI=Sensor/Intuitive; VV=Visual/Verbal; AR=Active/Reflective; SG=Sequential/Global NEU=Neutral; SNS=Sensory; INT=Intuitive; VIS=Visual; VRB=Verbal; ACT=Active; REF=Reflective; SEQ=Sequential; GLO=Global

Each example in Table 9.5 will now be described:

- Student 1 has no strong preference for any of the FS dimensions, scoring 1 or 3, and so is classified as Neutral for all dimensions. Following the algorithm the neutral adaptation is applied to all tutorial questions.
- Student 2 has strong preferences for all FS dimensions. For Q1, the student learning style score (shown first) is multiplied by the question learning style score, giving a student question score for that FS dimension. The winner is the FS dimension with the highest score; for Q1 the Active adaptation. For Q2 the winning adaptation is Sequential because, even though the student has a stronger preference for the Active learning style, the tutorial question offers more adaptations for the Sequential learning style. For Q3 there are two winners (Visual and Sequential), so first the student learning style scores are compared, followed by the tutorial question scores. As all scores are equal, the adaptation is selected randomly from the two winners, resulting in a Visual adaptation.

- Student 3 has a low preference for two FS dimensions (Intuitive and Visual) and a moderate preference for the Active and Global learning styles. For Q1, there are two winners, and as the student and question scores are equal, the Active adaptation was selected at random. For Q2, there are three winners, so the Global adaptation wins as it has the highest student learning style score. Q3 only adapts to three learning styles, so the clear winner is the Visual adaptation in this case.

An implementation of the adaptation algorithm applied to the FS model is described in Chapter 10. This step completes phase 1 of the methodology, describing the creation of the Learning Styles Adapter component.

4.2 Phase 2: Design a Tutorial Conversation

Phase 2 of the methodology involves capturing the tutorial from expert human tutors and iteratively developing a tutorial conversation with input from the human tutors. Several versions of tutorial questions (incorporating different teaching styles) are needed to match particular learning styles. Several steps in Phase 2 of the methodology (steps 2.1, 2.2 and 2.5) follow the same process as steps in the earlier Oscar PCITS methodology (Chapter 6), which was successfully validated empirically as described in Chapters 7 and 8.

4.2.1 Step 2.1: Capture the Tutorial Scenario and Questions from Human Tutors in a Specific Domain

The first step in designing a tutorial conversation involves capturing the tutorial scenario from human tutors and documenting it in the *tutorial conversation blueprint*. This step in the methodology follows the same process described in Chapter 6, Section 3.2.1 (Oscar PCITS Methodology, Step 2.1).

4.2.2 Step 2.2: Determine the Conversational Structure/Style

To structure the tutorial conversation, first apply the 3-level model of a tutorial conversation (Chapter 6, Figure 6.1) following the same process described in Chapter 6, Section 3.2.2 (Oscar PCITS Methodology, Step 2.2). As part of this process, a list of Frequently Asked Questions (FAQs) and answers is captured from human tutors.

Next, the tutorial questions should be mapped onto the generic question templates (Chapter 6, Figures 6.2 and 6.3) as described in Chapter 6, Section 3.2.3

(Oscar PCITS Methodology, Step 2.3). Any extra resources from the application of question templates should be included as required, and the dialogue updates recorded in the tutorial conversation blueprint.

4.2.3 Step 2.3: Map Tutorial Questions onto the Generic Teaching Material Categories

Step 2.3 of the methodology requires the tutorial questions documented in the *tutoring conversation blueprint* to be mapped to the generic teaching material categories developed in Step 1.3 of the methodology (Section 4.1.3). This mapping highlights each question where no adaptation exists for a learning style, thus promoting further capturing of different styles of teaching material from human tutors. The result is a list of available teaching materials for each question, and thus adaptations to each learning style.

The generic teaching material categories (Table 9.3) were designed from the tutor's point of view to simplify the development of consistent tutoring material for an ACITS. The standard organisation of tutor material into categories also facilitates modular development, as teaching materials can be expanded and added without the need for a total redesign of the tutoring conversation. This modular approach allows tutoring material to be developed in stages with no requirement for all learning styles to have adaptations before the ACITS can be used, and so speeds up the initial development.

4.2.4 Step 2.4: Score Tutorial Questions for Adaptation to Each Learning Style

In Step 2.4, each question is assigned a score for every learning style, which represents the number of opportunities for adaptation to that learning style. This is done using the mapping to teaching material categories resulting from Step 2.3. For each question, for each mapped teaching material category assign one point to every associated learning style (Table 9.3) each time the teaching material occurs. Where no adaptation exists for a learning style, the score assigned is zero. This scoring mechanism counts the number of adaptations a question has for each learning style, thus recording the strength of adaptation.

4.2.5 Step 2.5: Script Conversational Agent Natural Language Dialogue for each Tutorial Question using the 3-Level Model

Step 2.5 of the methodology involves creating Conversational Agent (CA) scripts to conduct the tutoring dialogue defined in steps 2.1, 2.2, 2.3 and 2.4 (and recorded in the tutorial conversation blueprint). Before scripting the dialogue, it is necessary to organise the CA scripts in two ways:

1. Apply the 3-level model, as described in Step 2.2 and recorded in the tutorial conversation blueprint.
2. By learning style adaptation, as defined in the mapping produced in Step 2.3.

The scripting of the CA dialogue follows the same process described in Chapter 6, Section 3.2.4 (Oscar PCITS Methodology, Step 2.4).

4.3 Phase 3: Construct the ACITS Architecture

Once the learning styles adapter module and the tutorial conversation have been designed, they must be incorporated into an ACITS architecture. The ACITS will require a number of components including a CA, a Tutorial Knowledge Base, a Graphical User Interface (GUI) and a Student Model. The components will be described in section 5, which proposes a standard Oscar ACITS architecture that is generic and incorporates the required components.

5 Oscar ACITS Architecture

The Oscar ACITS is independent of the learning styles model and subject domain being taught, so a modular architecture, which allows the reuse and replacement of individual modules, is most appropriate. The generic Oscar PCITS architecture proposed in Chapter 6 was validated empirically and proved successful in delivering online conversational tutorials (Chapters 7 and 8). Therefore the Oscar PCITS architecture was reused and adapted to suit the Oscar Adaptive CITS, as shown in Figure 9.1.

The generic Oscar ACITS architecture allows alternative learning styles models (i.e. the Learning Styles Adapter module created following phase 1 of the methodology) and subject domains (i.e. tutorial knowledge base and CA scripts created following phase 2 of the methodology) to be applied.

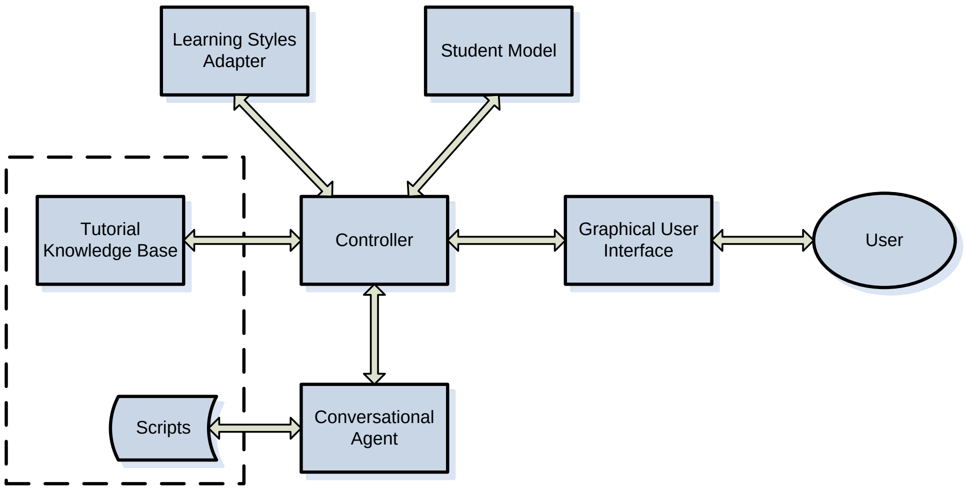


Figure 9.1. Generic Oscar ACITS Architecture

The main components of the architecture (described in Chapter 6, Section 5) have been reused, with some amendments relating to the adaptation of tutoring, which are summarised here. The *controller* and *Graphical User Interface (GUI)* modules manage the communication between components and the user. The *student model* module records information about the students, and was amended to also record the adaptations applied to each tutoring question. The *conversational agent (CA)* module is responsible for the natural language conversation. The *tutorial knowledge base* manages course information, and was amended to include available adaptations and adaptation scores for each tutorial question. The tutorial knowledge base and CA scripts are developed following phase 2 of the methodology.

The *learning styles adapter* component is responsible for accessing information about learning styles and related teaching styles, held in a learning styles database. This component will receive information from the CA, GUI, tutorial knowledge base and student model to select the best adaptation for a student's learning style. Given learning style values from the student model and tutorial question scores from the knowledge base, this component will apply the Oscar ACITS adaptation algorithm (section 4.1.4) to determine the most appropriate adaptation for each individual tutorial question. This module is developed by following phase 1 of the Oscar ACITS methodology.

An implementation of the Oscar ACITS architecture will be described in the next chapter, Chapter 10.

6 Conclusion

This chapter has proposed an original generic methodology and architecture for creating an Oscar ACITS that can dynamically adapt its tutoring to a student's learning styles. Adapting tutoring material to a student's learning styles aims to improve the effectiveness and the acceptance of tutoring with a CITS. It is hoped that this intelligent personalisation of tutoring will also increase student motivation and learning gain. The Oscar ACITS adaptation strategy considers both the strength of the student learning styles and the strength of available adaptations for the question. Students are given the best adaptation for each question separately, rather than only adaptations for their strongest learning style. Therefore the adaptation approach provides a variety of adaptations that address all aspects of the student's learning preferences. This approach allows flexibility in developing tutorials so that a coherent learning experience can be developed, regardless of the extent of adaptations available for every learning style. It recognises that in real life it may not be possible to provide adaptive material for all learning styles. The novel adaptation algorithm is generalised, and can be adopted for any learning styles model with strengths of learning style.

The proposed methodology is generic as it is independent of the learning styles model adopted and the subject domain taught. The methodology consists of three phases:

- The first phase relates to the analysis of the learning styles model in order to create the Learning Styles Adapter module. During this phase, a generic set of teaching material categories was proposed, which allow a more standard and teacher-friendly development of tutoring material for an Adaptive CITS. Also proposed was the Oscar ACITS adaptation algorithm, which is generic as it is independent of the learning styles model selected. The novel algorithm selects the best fitting adaptation for each tutorial question separately based on the strength of the student learning styles and also the strength of available adaptations for the question.
- The second phase of the methodology directs the design and development of the tutorial conversation. After capturing the tutorial scenario from human tutors, the tutorial questions are structured and mapped to the generic teaching material categories. Questions are assigned adaptation scores, and

then scripted for the selected conversational agent. A modular development approach is encouraged, and materials may be expanded at any time, with only the need to update question scores before the new material is taken into account.

- Phase 3 of the methodology involves constructing the ACITS architecture. A modular architecture based on that proposed in the earlier methodology is proposed, and most components require only minor amendments.

In order to validate the proposed methodology and architecture and investigate the success of the adaptive tutoring an experimental study is required. The next chapter will describe the development of a prototype Oscar ACITS and report the experiments conducted to test the success of the adaptation mechanism.

7 Chapter Highlights

- Oscar ACITS is a novel conversational intelligent tutoring system that automatically adapts to learning styles while directing a tutoring conversation.
- Oscar ACITS is independent of the learning styles model selected and the tutoring domain.
- An original, generic methodology was proposed to construct an adaptive Oscar ACITS.
- A set of generic teaching material categories to aid the development of an adaptive tutorial was created.
- A novel generic adaptation algorithm was proposed, which considers both the student learning styles strength and the strength of adaptation of each tutorial question to select the best fitting adaptation.
- A generic, modular architecture for Oscar ACITS was proposed.

Chapter 10 Adaptation to Learning Styles Experiments

1 Introduction

The Oscar ACITS proposed in Chapter 9 can dynamically adapt the style of each tutorial question to match a student's learning styles. A generic methodology and architecture for developing Oscar ACITS, which are independent of the learning styles model and subject domain, were proposed. In order to validate the proposed methodology and architecture, it is necessary to implement and empirically test the Oscar ACITS.

This chapter will describe the implementation of a prototype Oscar ACITS following the methodology and architecture in Chapter 9. The prototype adopts the Felder-Silverman (FS) learning styles model (Felder and Silverman, 1988). Using the generic teaching material categories to aid development (described in Chapter 9), an SQL revision tutorial was developed that adapts to different learning styles. Finally, a number of components from the Oscar PCITS prototype were reused and modified in implementing the Oscar ACITS architecture.

Next, a real-world study undertaken to validate the methodology and architecture proposed in Chapter 9 is described. Seven experiments were designed to test the hypotheses that adapting to a student's learning styles can improve the effectiveness and efficiency of an online conversational tutorial. As well as evaluating the success of Oscar ACITS adaptation approach, the study investigated the general success of the tutoring in terms of learning and user feedback. The experimental results show that the Oscar ACITS adaptation method has been successful in improving the learning of participants who experience a tutorial adapted to suit their learning styles.

2 Implementing the Oscar Adaptive CITS

To validate the methodology and architecture proposed in Chapter 9, a prototype Oscar ACITS was implemented following the 3-Phase Methodology proposed in Chapter 9 and repeated in Table 10.1. The development of the Oscar ACITS prototype will now be described.

Table 10.1. 3-Phase Methodology for Creating Oscar ACITS.

Phase 1: Create the Learning Styles Adapter Module

1.1. Select a Learning Styles Model and extract the behaviour characteristics
1.2. Map learning style behaviour to associated conversational tutoring style
1.3. Map learning styles to teaching material categories
1.4. Implement the generic adaptation algorithm for chosen learning styles model

Phase 2: Design a Tutorial Conversation

2.1. Capture the tutorial scenario and questions (including movies, voice, images, examples, etc.) from human tutors in a specific domain
2.2. Determine the conversational structure/style
2.3. Map tutorial questions onto the generic teaching material categories
2.4. Score tutorial questions for adaptation to each learning style
2.5. Script Conversational Agent natural language dialogue for each tutorial question using the 3-level model

Phase 3: Construct the ACITS Architecture

2.1 Phase 1: Create the Learning Styles Adapter Module

For the prototype Oscar ACITS, the Felder-Silverman (FS) model was adopted (Felder and Silverman, 1988) (see Chapter 5, section 2) as it has a small number of dimensions (which is more feasible to implement) and it describes the learning styles of engineering students who will make up the initial experimental group.

2.1.1 Steps 1.1 to 1.3

Following steps 1.1 and 1.2, the FS model was examined to extract the knowledge of learning styles and associated teaching styles. The resulting knowledge was summarised in a table of behaviour characteristics and related teaching styles (Table 9.2). In step 1.3, this knowledge was grouped to devise a list of categories of teaching materials to be included in the tutorial, which was then mapped to the FS learning styles (Table 9.3). Chapter 9 described these steps of the methodology with examples from the analysis of the FS model.

2.1.2 Step 1.4: Implement the Generic Adaptation Algorithm for Chosen Learning Styles Model

The final step in phase 1 (step 1.4) was to implement the generic algorithm (Chapter 9, Table 9.4) that decides the most appropriate type of adaptation to be applied for each student's individual learning path.

The FS model has four learning style dimensions, each with two opposite learning styles, so there are 8 learning styles plus the Neutral learning style. Learners are placed along each FS dimension according to the strength of their preference for a particular learning style, measured using the Index of Learning Styles (ILS) instrument (Felder and Soloman, 1997). In Chapter 5 (Section 3), a study of the results of 108 ILS questionnaires showed them to be consistent with the FS model, which states that on the Visual/Verbal scale most engineering students were found to be Visual learners (Felder and Silverman, 1988). Considering that Oscar ACITS tutorials are conducted using a conversation (Verbal) with supporting images and diagrams throughout (Visual), and the Massa and Mayer (2006) study shows no difference in learning achievement when adapting multimedia teaching to Visual/Verbal learners, it was decided to exclude the Visual/Verbal FS dimension. Therefore for the Oscar ACITS prototype adaptation was implemented for the remaining three FS learning style dimensions: Sensor/Intuitor, Active/Reflective and Sequential/Global.

The Oscar ACITS adaptation strategy states that learners with a *neutral* learning style class are given learning material adapted to a mixture of styles. In the FS model, neutral learners are defined as those with an ILS questionnaire score of 1 or 3 for a learning style, indicating a low preference and placing them at the centre of the dimension (as described in Chapter 2).

Table 10.2 shows a pseudo-code representation of the adaptation algorithm applied to the three selected dimensions of the FS model. The algorithm receives a list of student learning style scores and a list of question scores and returns the best fitting adaptation for each question. This completes phase 1 of the methodology, the implementation of the Learning Styles Adapter module.

2.2 Phase 2: Design a Tutorial Conversation

For the Oscar ACITS prototype, the tutorial conversation designed for the Oscar PCITS prototype (Chapter 7) was reused and modified. The subject domain of the database Sequential Query Language (SQL) was chosen because undergraduate computing students, for whom SQL is compulsory, make up the study group. The tutorial conversation was designed following phase 2 of the methodology, as described in the following sections.

2.2.1 Steps 2.1 and 2.2

For the Oscar ACITS prototype, it was decided to reuse the tutorial developed for the Oscar PCITS prototype, which delivers an SQL revision tutorial.

Table 10.2. Domain-independent Pseudo-code Adaptation Algorithm Applied to the FS model

Let:

- the question scores be q_{SNS}, q_{INT}, q_{ACT}, q_{REF}, q_{SEQ}, q_{GLO}, q_{NEU},
- the student learning style classes be c_{SI}, c_{AR} c_{SG}
- the student scores be s_{SI}, s_{AR}, s_{SG},
- the student question scores be a_{SI}, a_{AR}, a_{SG} (where *SI*, *AR* and *SG* are the classes of learning style dimensions),
- the set of winning student question scores be *W*.

```
IF ((c_SI==NEU) AND (c_AR==NEU) AND (c_SG==NEU))
THEN
    {adapt_to_class(NEU);}          // if all neutral learning styles, choose neutral adaptation
ELSE                                // calculate student question scores
{
    IF (c_SI==SNS) THEN a_SI=q_SNS*s_SI
        ELSE IF (c_SI==INT) THEN a_SI=q_INT*s_SI
        ELSE a_SI=0;
    IF (c_AR==ACT) THEN a_AR=q_ACT*s_AR
        ELSE IF (c_AR==REF) THEN a_AR=q_REF*s_AR
        ELSE a_AR=0;
    IF (c_SG==SEQ) THEN a_SG=q_SEQ*s_SG
        ELSE IF (c_SG==GLO) THEN a_SG=q_GLO*s_SG
        ELSE a_SG=0;
    W = get_max_list(a_SI, a_AR, a_SG);        // get list of winners – maximum scores
    IF size(W) == 1 THEN                       // if only one winner adapt to that class
        {
            IF MAX(a_SI, a_AR, a_SG) == a_SI THEN adapt_to_class(c_SI);
            ELSE IF MAX(a_SI, a_AR, a_SG) == a_AR THEN adapt_to_class(c_AR);
            ELSE adapt_to_class (c_SG);
        }
    ELSE                                       // if >1 winner adapt to max student score in W
        {
            X = get_max_list_from(s_SI, s_AR, s_SG,W);
            IF size(X) == 1 THEN               // if only one winner adapt to that class
            {
                IF MAX(s_SI, s_AR, s_SG) == s_SI THEN adapt_to_class(c_SI);
                ELSE IF MAX(s_SI, s_AR, s_SG) == s_AR THEN adapt_to_class(c_AR);
                ELSE adapt_to_class (c_SG);
            }
            ELSE                               // if >1winner adapt to max question score in W
            X = get_max_list_from(q_SI, q_AR, q_SG,W);
            IF size(X) == 1 THEN               // if only one winner adapt to that class
            {
                IF MAX(q_SI, q_AR, q_SG) == q_SI THEN adapt_to_class(c_SI);
                ELSE IF MAX(q_SI, q_AR, q_SG) == q_AR THEN adapt_to_class(c_AR);
                ELSE adapt_to_class (c_SG);
            }
            ELSE                               // if more than one winner adapt to Random in W
                adapt_to_class(RANDOM(W))
        }
}
```

Key:
SI=Sensor/Intuitive; AR=Active/Reflective; SG=Sequential/Global
NEU=Neutral; SNS=Sensory; INT=Intuitive; ACT=Active; REF=Reflective; SEQ=Sequential; GLO=Global

For step 2.1, the original capture of the tutorial conversation is described in Chapter 7, Section 4.1. The tutorial scenario consisted of ten tutorial questions (Table 7.2) with resources that support the adaptation of tutoring to different learning styles and a multiple choice question (MCQ) test.

For step 2.2, the captured tutorial conversation was structured by applying the 3-level model of a tutorial conversation, as described in Chapter 7, Section 4.2. Then the tutorial questions were mapped to the generic question templates, as described in Chapter 7 Section 4.3.

The resulting tutorial conversation was documented in the *tutorial conversation blueprint* document. Appendix 3 shows an excerpt of the tutorial conversation blueprint.

2.2.2 Step 2.3: Map Tutorial Questions onto the Generic Teaching Material Categories

In Step 2.3, the captured tutorial conversation was tailored to make it suitable for an adaptive tutorial. This was done by mapping the tutorial questions to the generic teaching material categories (Chapter 9, Table 9.3). Next, this mapping was reorganised to highlight the available adaptations per question, and learning styles requiring additional adaptive teaching material. Further interviews with human tutors were undertaken to gather more material to improve the adaptations in the SQL tutorial, such as introductions and more examples. The additional resources were then documented in the *tutorial conversation blueprint* document, agreed with human tutors and the mappings updated. Table 10.3 shows the final mapping of the SQL revision tutorial to learning style adaptations, derived from the mapping to teaching material categories. Every question also has a *Neutral* learning style adaptation which includes a mixture of styles.

As many tutor material categories as possible were included in the design of each tutorial question. It is recognised that to make a coherent learning experience, some tutorial questions may not lend themselves to adaptation (e.g. Q3 in Table 10.3 has only two adaptations other than Neutral). This demonstrates the strength of the adaptation algorithm in considering both the individual question adaptations and student learning styles strengths over algorithms that require adaptive material for all learning styles, or those that adapt to just one dimension (see Chapter 9, section 3).

Table 10.3. Learning Style Adaptations in the SQL Revision Tutorial

Question	Neutral	Active	Reflective	Sequential	Global	Sensor	Intuitive
Introduction	✓			✓	✓	✓	✓
Q1	✓	✓	✓	✓	✓	✓	✓
Q2	✓	✓	✓	✓	✓	✓	✓
Q3	✓	✓				✓	
Q4	✓	✓	✓	✓	✓	✓	✓
Q5	✓	✓	✓	✓	✓	✓	✓
Q6	✓	✓	✓	✓	✓	✓	✓
Q7	✓	✓	✓	✓		✓	✓
Q8	✓	✓	✓	✓		✓	✓
Q9	✓	✓	✓	✓	✓	✓	✓
Q10	✓	✓	✓	✓		✓	✓

2.2.3 Step 2.4: Score Tutorial Questions for Adaptation to Each Learning Style

In Step 2.4, each tutorial question was assigned a score for every learning style representing the number (or strength) of opportunities for adaptation to that learning style. This was done for each question by simply counting the number of times each category of teaching material related to a learning style was available. Where no adaptations existed for a learning style, the score assigned was zero. The neutral learning style was always assigned a score of zero. Table 10.4 shows the final scores for the SQL revision tutorial.

Table 10.4. Question Adaptation Scores

Question	Active	Reflective	Sequential	Global	Sensor	Intuitive
Introduction	0	0	1	2	1	1
Q1	4	1	1	1	3	3
Q2	2	1	1	1	2	2
Q3	2	0	0	0	2	0
Q4	5	3	3	1	5	4
Q5	9	5	11	11	9	11
Q6	3	2	2	1	3	3
Q7	4	2	2	0	4	3
Q8	3	2	2	0	3	2
Q9	7	3	7	7	7	3
Q10	4	2	2	0	4	2

2.2.4 Step 2.5: Script Conversational Agent Natural Language Dialogue for each Tutorial Question using the 3-Level Model

The development of scripts for the InfoChat CA (Convagent Ltd., 2005) is fully described in Chapter 7, Section 4.4. For the Oscar ACITS prototype, whilst the organisation of the scripts over three levels into separate contexts stayed the same, multiple versions of the CA scripts representing different adaptations were required. During this step, for each learning style the Oscar PCITS prototype scripts were amended to match the mapped teaching material categories for that learning style (as documented in the *tutorial conversation blueprint).* For example for the Active learning style, CA scripts for tutorial questions based on the *hints* question template (see Chapter 7, Table 7.2) were amended to replace any detailed explanations of theory with hints based on practical examples and exercises.

For each tutorial question, a CA script name was assigned to each learning style and to the *Neutral* learning style. Where a question score was zero for a learning style (i.e. no adaptation existed) the neutral CA script was assigned. The script names and scores were recorded for inclusion in the tutorial knowledge base component, which will be described in Section 2.3.

2.3 Phase 3: Construct the ACITS Architecture

The Oscar ACITS architecture proposed in Chapter 9, shown in Figure 10.1, was adopted. Oscar ACITS' modular structure allowed several of the components developed for the prototype Oscar Predictive CITS (described in Chapter 7, Section 5) to be reused, speeding up development.

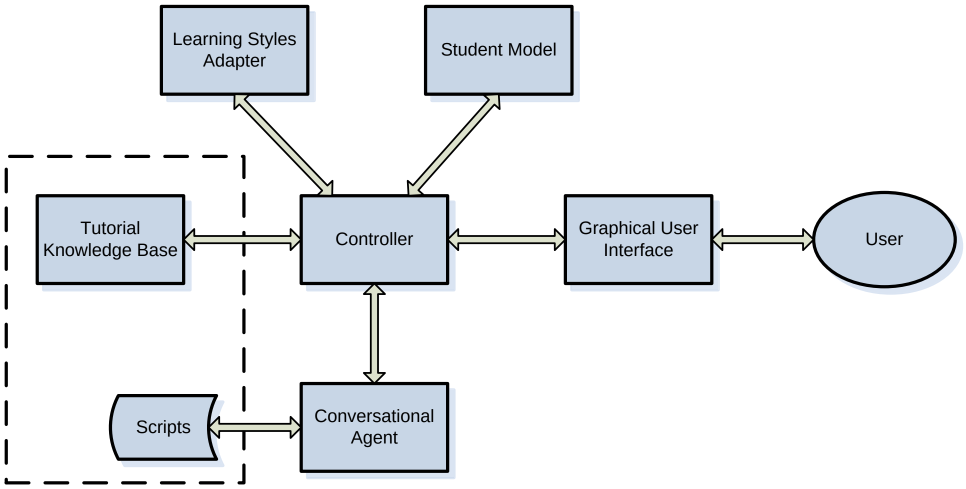


Figure 10.1. Oscar ACITS Architecture

Controller

The controller manages the tutorial and communicates with all components. The controller developed for the Oscar PCITS prototype (Chapter 7, section 5) was reused but changed. For the Oscar ACITS prototype, additional functionality relating to dynamically selecting adaptations was required, as follows:

- At the start of each tutoring question, retrieve the student learning style scores and tutorial question scores, and pass to the learning styles adapter component.
- On receiving the start context from the learning styles adapter, pass to the CA and start the conversation.

Apart from managing individual adaptations (as described above), the controller's management of the tutorial remained the same as in the Oscar PCITS prototype.

Graphical User Interface (GUI)

For the Oscar ACITS prototype, the GUI component created for the Oscar PCITS prototype (Chapter 7, section 5) was reused and required no changes.

Student Model

For the Oscar ACITS prototype, the student model developed for the Oscar PCITS prototype (described in Chapter 7, Section 5) was extended by adding the *Session* table to record the adaptations. This enables detailed information about the adaptive student learning experience to be analysed. The Oscar ACITS prototype student model class diagram is shown in Figure 10.2.

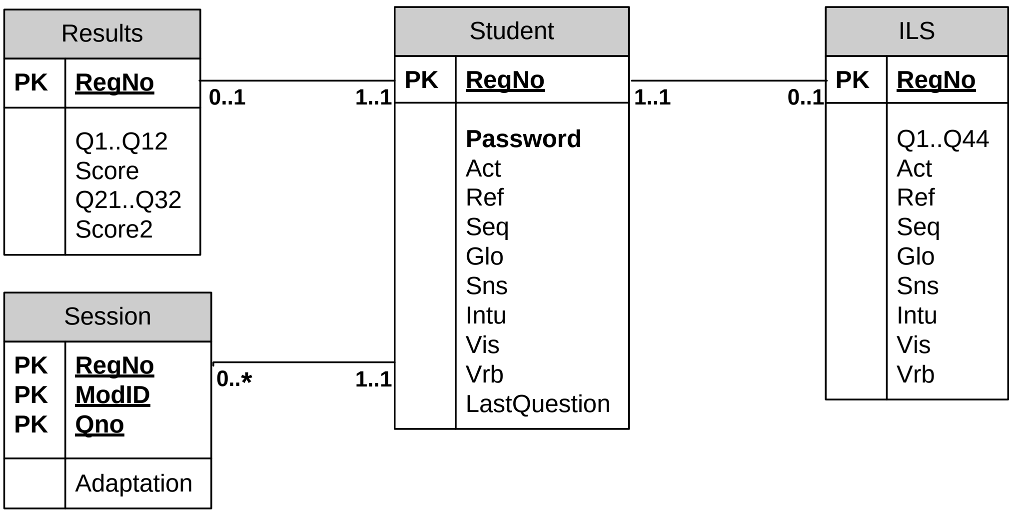


Figure 10.2. Student Model Class Diagram

- The *Session* table has a record for every tutorial question a student has attempted, and records the adaptation applied.

Conversational Agent (CA)

The CA module from the Oscar PCITS prototype, described in Chapter 7, Section 5, was reused for the Oscar ACITS prototype and required no changes. The design of the CA scripts followed step 2.5 of phase 2 of the Oscar ACITS methodology, described in Section 2.2.4.

Tutorial Knowledge Base

This module was developed by following phase 2 of the Oscar ACITS methodology, as described in Section 2.2. For the Oscar ACITS prototype, the tutorial knowledge base developed for the Oscar PCITS prototype (described in Chapter 7, Section 5) was extended to include a new table recording the adaptation details for each tutorial question, as shown in Figure 10.3. The adaptation scores are used by the Learning Styles Adapter module to determine the best adaptation for a tutorial question. The adaptation start contexts are passed to the CA at the start of each tutorial question.

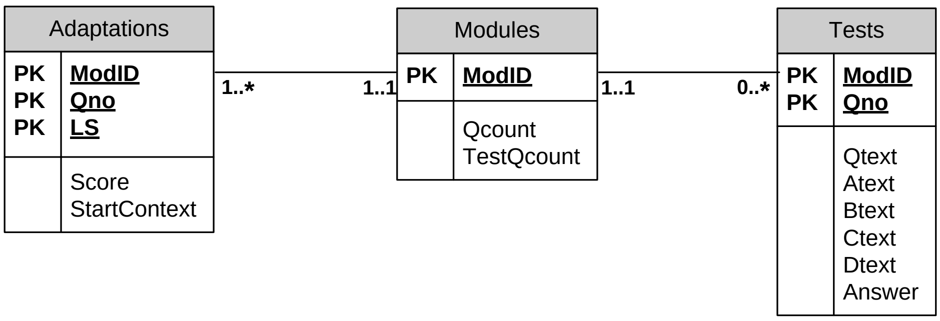


Figure 10.3. Tutorial Knowledge Base Class Diagram

- The *modules* table was amended to remove the StartContext column, which is now recorded in the *adaptations* table.
- The *adaptations* table has a record for each learning style for each module tutorial question, recording the adaptation score and the name of the CA script to start with (as there may be several linked contexts for a question, see section 2.2.4).

The tutorial knowledge base is related to (but not linked to) the CA scripts that deliver the tutorial conversation.

Learning Styles Adapter

The learning styles adapter module contains the implementation of the adaptation algorithm. The learning styles adapter receives the question adaptation scores for each learning style from the tutorial knowledge base and the student learning style scores from the student model (via the controller) and determines the best adaptation for each tutorial question. This module was developed by following phase 1 of the Oscar ACITS methodology.

This completes the development of the Oscar ACITS prototype. The next section will show some example dialogue that demonstrates how the tutoring is adapted for different learners.

2.4 Example Adaptive Learner Dialogues

Examples of the Oscar ACITS tutoring style adaptation can be seen in the dialogue snippets logged during the experiments (Table 10.5 and Table 10.6).

Table 10.5. Dialogue Snippet Logged During the Experiments: Adapting to a Sequential Learner.

Utterance	Dialogue
87. Oscar>	Now let's see if you can remember how to write a query which chooses values between a range. Look at the employees table. We need a query which displays the first and last names of all employees hired between 1 January 1994 and 1 January 1995. The first step in writing a query is to state the information you need. For our query to display the first and last names of all employees hired between 1 January 1994 and 1 January 1995, which columns do we need?
88. Student>	dont know
89. Oscar>	Look again at the columns in the EMPLOYEES table. If we want to display the first and last names of all employees hired between 1 January 1994 and 1 January 1995, which columns do we need?
90. Student>	first_name, last_name, hire_date
91. Oscar>	That's right, we need FIRST_NAME and LAST_NAME, so the first part of our query is shown in the picture. The next step in writing a query is to state which table the information is retrieved from. For our query to list the first and last names of all employees hired between 1 January 1994 and 1 January 1995, type in the name of the table our information will be taken from.
92. Student>	employees
93. Oscar>	That's right, the table is EMPLOYEES. The last part of writing a query is the WHERE clause, which lets us restrict the rows selected to only those satisfying one or more conditions. We want to restrict the employees displayed to only those hired between 1 January 1994 and 1 January 1995. There are a number of ways to write this restriction in ORACLE SQL as dates can be represented in different formats. Type in the WHERE clause which limits the employees in this way.
94. Student>	where hire_date in('01-01-1994' && '01-01-1995')
95. Oscar>	One way to tackle this query is to use the BETWEEN keyword to specify a range. Complete the last line of the query shown - type in the WHERE clause to limit the employees displayed to those hired between 1 January 1994 and 1 January 1995.
96. Student>	'01-01-1994' and '01-01-1995'
97. Oscar>	OK, excluding date conversion functions, ORACLE recognises dates in 3 basic formats: '01-JAN-94', '01-January-94' and '01-January-1994'. The picture shows two different ways in which the WHERE clause could have been written to give the right results.

Both example dialogues (Table 10.5 and Table 10.6) are taken from the same tutorial question where students are asked to write an SQL query, and both students were assigned to the Adapt experimental group (i.e. the adaptation matched their learning style). The student dialogue in Table 10.5 is an example of the tutoring style adapted to a Sequential learner, where Oscar guides the student through the steps of the query. Table 10.6 in contrast shows a student dialogue with tutoring adapted for a Global learner, where the student writes a query without initial guidance and learns from the feedback given about errors in their solution.

Table 10.6. Dialogue Snippet Logged During the Experiments: Adapting to a Global Learner.

Utterance	Dialogue
64. Oscar>	Now let's see if you can remember how to write a query which chooses values between a range. Look at the employees table. Please type in a query which displays the first and last names of all employees hired between 1 January 1994 and 1 January 1995.
65. Student>	select First_name, last_name from Employees where Hire_date between 1-JAN-1944 and 1-Jan-1995;
66. Oscar>	I'm afraid your WHERE clause is not quite right. Check your spelling and type in the last part of our query which displays the first and last names of all employees hired between 1 January 1994 and 1 January 1995.
67. Student>	select First_name, last_name from Employees where Hire_date between 1-JAN-1994 and 1-Jan-1995;
68. Oscar>	No. Excluding date conversion functions, ORACLE recognises dates in 3 basic formats: '01-JAN-94', '01-January-94' and '01-January-1994'. The picture shows two different ways in which the WHERE clause could have been written to give the right results.

3 Experimental Design

In order to validate the methodology and architecture proposed in Chapter 9, an empirical study was undertaken in a real-world natural learning environment. The study evaluated the success of Oscar ACITS in adapting its tutoring to individuals' learning styles. This section will describe the design of the experiment, including the hypotheses to be tested and method of evaluation.

3.1 Hypotheses to be Tested

There are two hypotheses to be tested, which relate to the success of the Oscar ACITS adaptation, as follows:

- *H1: it is possible to improve learning from an automated online conversational tutorial by presenting tutor material adapted to a student's learning style.*

Can any evidence be found to support the learning styles theory in suggesting that adapting teaching material to match preferred learning styles improves learning? A common measure of learning is learning gain (Kelly and Tangney, 2006; Graesser et al., 2003; Lee et al., 2004). Learning gain could be measured in a number of ways, for example an improvement in test scores or the number of tutorial questions a learner answers correctly. To test this hypothesis it will be necessary to compare learning gain for a group of learners who experience a tutorial adapted to suit their learning styles with a control group.

- *H2: it is possible to improve the efficiency of an automated online conversational tutorial by presenting tutor material adapted to a student's learning style.*

Is there any evidence that adapting teaching material to match preferred learning styles improves the efficiency of learning? Efficiency may be measured in a number of ways, for example by comparing the duration of a conversational tutoring session or the amount of discussion taking place.

3.2 Evaluation Criteria

In addition to evaluating the effect of the Oscar ACITS adaptation to learning styles by testing the hypotheses stated in section 3.1, Oscar ACITS' ability to tutor effectively will be investigated. Evaluation of the Oscar ACITS will therefore take place on three levels:

1. *Adaptation:* Can Oscar ACITS successfully adapt its tutoring to individuals' learning styles? Does the Oscar ACITS adaptation to learning styles improve the learning gain or efficiency of the tutoring?
2. *User evaluation:* How successful do learners believe Oscar ACITS is and would they use Oscar ACITS in practice?
3. *Learning gain:* Does Oscar ACITS successfully tutor learners, i.e. do they learn anything?

3.2.1 Adaptation to Learning Styles

This criterion evaluates the second main research question (stated in Chapter 1, Section 1), 'Does adapting to a student's learning style during a two-way tutoring discourse with a conversational agent tutor improve learning?'. In order to evaluate whether the Oscar ACITS adaptation to learning styles has a positive effect on the tutoring, it is necessary to split participants into different experimental groups. A match/mismatch approach was adopted (Tsianos 2008), whereby participants are randomly assigned to follow a tutorial either matched or mismatched to their learning styles. The match/mismatch approach was considered to be a better test of the adaptation than an approach where one control group experiences a basic tutorial, as it was concluded that any group experiencing additional learning material would be likely to show improved learning.

Each participant will be asked to complete the ILS questionnaire, and depending on their learning styles will be unknowingly assigned to one of three experimental groups, as follows:

- Learners whose learning styles are at the centre of all three FS scales (i.e. there is no strong preference, their ILS scores being 1 or 3) will be assigned to the *Neutral-Adapt* group. These learners will follow the neutral adaptation learning path which contains a mixture of styles.
- Learners with at least one preferred learning style will be randomly assigned to either the *Adapt* or *Mismatch* group according to a 2:1 ratio. These learners will follow an adaptive learning path assigned by the algorithm. Learners in the Mismatch group will be deliberately presented with learning material unsuited to their learning styles.

The average learning gain and efficiency of the tutorials will be compared for each experimental group, to evaluate whether adapting to learning styles positively affects the success of the tutoring.

3.2.2 Qualitative User Evaluation

In addition to evaluating the Oscar ACITS adaptation approach, qualitative user feedback will be gathered at the end of the Oscar ACITS tutorial. The user evaluation feedback questionnaire designed for the Oscar PCITS study (and described in Chapter 8, Section 2.2.3) will be reused to collect participant feedback.

3.2.3 Learning Gain

In order to additionally investigate whether participants have increased their knowledge at the end of the tutorial, learning gain will be measured using a pre-test and post-test approach (Kelly and Tangney, 2006; Graesser et al., 2003; Lee et al., 2004). The same Multiple Choice Question (MCQ) test will be completed before and after the tutoring conversation. The MCQ test scores will be compared to establish whether there is any improvement as follows:

Eq. 1. Learning gain = post-test score - pre-test score

4 Experimental Methodology

This section describes the experimental methodology followed to test where Oscar ACITS can deliver an effective conversational tutorial, and whether adapting to learning styles has a positive effect on the tutoring. As described in Section 2, the Oscar ACITS prototype was implemented to deliver an adaptive conversational tutorial for SQL revision. Oscar ACITS was integrated into a final year undergraduate module within the Department of Computing and Mathematics at Manchester Metropolitan University. An uncontrolled, real-world experiment was undertaken in a natural learning environment.

4.1 Description of Participants

There were 72 participants who were final year undergraduate students studying for a computer science degree. The participants had previously been taught SQL, although most would not have used SQL for at least six months. No participant had any previous experience using Oscar ACITS.

4.2 Methodology

The Oscar ACITS SQL Revision tutorial was integrated into a final year undergraduate module. During timetabled laboratory classes, participants were asked to refresh their SQL knowledge by completing the revision tutorial. In order to promote the completion of the tutorial, participants who completed it were awarded 2% of the module mark in recognition of engagement. Participants started the SQL revision tutorial during the laboratories, and those who did not complete the tutorial in a single session were able to continue the tutorial via the Internet at a convenient time.

Each participant was unknowingly assigned to one of three experimental groups (*Neutral-Adapt, Adapt* and *Mismatch*) as follows:

- Participants with no strong preference for all three FS dimensions (i.e. their ILS scores were 1 or 3) were assigned to the *Neutral-Adapt* group. These participants followed the neutral adaptation learning path, which contains a mixture of styles.
- Participants with at least one preferred learning style were randomly assigned to either the *Adapt* or *Mismatch* group according to a 2:1 ratio. These participants followed an adaptive learning path assigned by the algorithm and were given tutor material favouring particular learning styles (e.g. containing explanations of theory rather than practical examples). Participants in the Mismatch group were deliberately presented with learning material unsuited to their learning styles by reversing their learning style scores for all FS dimensions. For example, a participant with learning style scores of Active 9 and Reflective 2 was presented with learning material adapted to the scores Active 2 and Reflective 9.

Each participant followed an individual learning path depending on their experimental group, learning styles, dialogue and existing knowledge. The participant interaction with Oscar ACITS during the experiment will be described in Section 4.3. During the SQL Revision tutoring session, ten questions were posed, requiring eighteen answers (as some questions incorporated multiple steps or questions). Each participant's tutoring dialogue, adaptations, timings, knowledge and other behaviour factors were recorded in log files as described in Section 2.3. Following the study, the data gathered was analysed and the experimental group averages were compared to assess the success of the adaptation mechanism, as will be detailed in Section 4.4. In addition, the tutoring success was evaluated in terms of participant learning gain and participant experiences reported in the feedback questionnaires.

4.3 Participant Interaction

Figure 10.4 illustrates the stages involved in the participant interaction with Oscar ACITS during the study. As shown in Figure 10.4, after registering participants completed the formal ILS questionnaire before beginning the tutorial. Next, students completed a pre-tutorial multiple choice question (MCQ) test, known as the *pre-test*, to assess existing knowledge before starting the conversational tutorial. The conversational SQL revision tutorial took on average approximately 43 minutes, with each participant following an individual learning path depending on their knowledge, learning styles and experimental group. After completing the tutorial conversation, students repeated the same MCQ test, known as the *post-test*, and were then presented with some tutor feedback and a comparison of their test results (indicating their learning gain). Finally, students were asked to complete a user evaluation questionnaire.

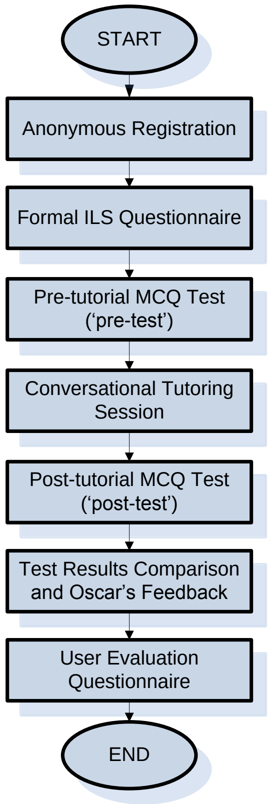


Figure 10.4. Stages in the Experimental Oscar ACITS Tutorial Interaction

4.4 Experimental Analysis

The data gathered from the participant interactions was analysed to explore whether the Oscar ACITS adaptation to learning styles improved the tutoring. Seven experiments were designed to test the two hypotheses (Section 3.1) and the results of each experimental group were compared to see if adapting to learning styles affected the tutoring. The analysis performed for each of the seven experiments will now be described.

Experiment 1 – Correct Tutorial Answers

This experiment tests hypothesis H1 by considering the performance of participants during the tutorial. For the ten tutorial questions posed, eighteen answers were required as some questions incorporated multiple steps or questions. The number of correct answers given to tutoring questions was counted, and a score out of 18 assigned for each participant. Next, the average percentage score was calculated for each experimental group. The experimental group averages were then compared to determine whether there was any difference in performance related to the adaptation style.

Experiment 2 – MCQ Test Score Improvement

This experiment tests hypothesis H1 by considering the actual improvement in test scores from the pre-test to the post-test (defined in Eq. 1 (Section 3) as *learning gain*). Average test score improvements were calculated for each experimental group and then compared.

Experiment 3 – MCQ Test Score Improvement/Opportunity

Experiment 3 extends experiment 2, also testing hypothesis H1, by considering the average improvement in test scores as a percentage of the possible improvement. This measure is more accurate than experiment 2 as it also considers the opportunity for improvement, i.e. excludes those participants who achieved 12/12 in the pre-test. Improvements were calculated using the formula:

Eq. 2. $$\frac{\text{learning gain}}{(\text{questionCount - preTestScore})}$$

Average improvements for each experimental group were then compared.

Experiment 4 – MCQ Test Questions Worse

Experiment 4 tests hypothesis H1 by considering participants' performance in individual MCQ test questions rather than their overall scores. It is possible that in some cases the Oscar ACITS adaptive tutoring had a negative impact on learning. This experiment investigates whether there is any difference between experimental groups in the number of times, following the tutoring, participants performed worse in test questions. Questions where participants selected the correct answer in the pre-test but the incorrect answer in the post-test were counted. The averages were then calculated for each experimental group using the formula:

Eq. 3. $$\frac{(\text{worseCount}/12)}{\text{groupSize}}$$

The results for each experimental group were then compared.

Experiment 5 – MCQ Test Questions Better

This experiment also tests hypothesis H1 by considering individual test questions, by counting the number of cases a participant's performance improved in questions, i.e. questions were answered incorrectly in the pre-test but correctly in the post-test. The averages were calculated for each experimental group using the formula:

Eq. 4. $$\frac{(\text{betterCount}/12)}{\text{groupSize}}$$

The results for each experimental group were then compared.

Experiment 6 – Session Duration

Experiment 6 tests hypothesis H2 by considering the duration of the tutoring sessions. The average duration of the conversational tutoring sessions (in seconds) was calculated for each experimental group and then compared.

Experiment 7 – Number of Interactions

This experiment tests hypothesis H2 by considering the number of interactions (i.e. participant dialogue acts) during the tutorial. The average number of interactions during the tutorial was calculated for each experimental group and then compared.

5 Results and Discussion

This section will present the results of the study to validate the Oscar ACITS methodology and architecture presented in Chapter 9.

5.1 Overall Results

63 of the 72 participants fully completed the tutoring session; incomplete tutorial sessions were disregarded. Of the 63 complete tutorial sessions, one was disregarded as the participant had not engaged with the tutorial, answering 'no' to all questions and selecting the same answer for all multiple choice test questions. Table 10.7 shows the distribution of the 62 participants across experimental groups and the average (mean) test scores.

Table 10.7. Experimental Groups

Experimental Group	Number of Participants	Average Pre-test Score (/12)	Average Post-test Score (/12)
Neutral-Adapt	10	8.7	10.7
Adapt	32	8.6	10.8
Mismatch	20	8.1	10.8
Total	62	8.5	10.8

In Table 10.7, the ten Neutral-Adapt participants had learning style results that showed no strong preference for a particular learning style (i.e. their styles were balanced in the centre of the scale), and followed a neutral adaptation learning path containing a mixture of styles. The Adapt group contained 32 participants who followed a learning path containing teaching material in a style adapted to their individual learning styles. The Mismatch group of 20 participants followed an adaptive learning path of teaching material that was mismatched to their learning styles. The Mismatch group had a slightly lower average pre-test score (out of 12), but the average post-test scores were approximately the same for all participants across the sample.

On the whole, the Oscar ACITS tutorial made a positive improvement in participant test scores, with an average learning gain (calculated using Eq. 1) of 19% over the sample.

Table 10.8 shows the distribution of learning styles across the Adapt and Mismatch groups. Two learning styles (Intuitive and Global) are not represented in the Mismatch group, which is sometimes unavoidable with random assignment to experimental groups and where not all learning style dimensions are evenly balanced.

Table 10.8. Learning Style Distribution

Learning	**Adapt Group**		**Mismatch Group**	
Style	**n**	**% of Total**	**n**	**% of Total**
Sensory	17	49	18	51
Intuitive	4	100	0	0
Active	11	69	5	31
Reflective	7	70	3	30
Sequential	9	56	7	44
Global	5	100	0	0

5.2 Experimental Results

Table 10.9 reports the results of the seven experiments. All results were tested for difference between the experimental groups using the Kruskal-Wallis test (Kruskal and Wallis,1952) with a confidence interval of 95%. The Kruskal-Wallis test is non-parametric so it does not require normality and works with data represented as percentages (unlike the more common ANOVA). Each experiment will now be discussed individually.

Table 10.9. Experimental Results

Experiment	Neutral-Adapt	Adapt	Mismatch	Total
1. Average correct tutorial answers	71%	73%	61%	69%
2. Average actual MCQ test score improvement	17%	18%	22%	19%
3. Average MCQ test score improvement/opportunity	61%	65%	62%	63%
4. Average MCQ test questions worse	6%	3%	3%	3%
5. Average MCQ test questions better	23%	21%	25%	23%
6. Average duration of session (seconds)	2860	2632	2345	2576
7. Average number of interactions	46	43	44	44

Experiment 1 – Correct Tutorial Answers

Participants in the Neutral-Adapt and Adapt groups have similar averages of correct answers given during the tutoring, of 73% and 71% respectively. However, the Mismatch group has a much lower average of only 61% correct answers, which is 12% less than the Adapt group average. The Kruskal-Wallis test results gave a Kruskal-Wallis statistic of 10.31 and a p-value of 0.006, indicating a significant difference in the Mismatch group.

The results suggest that participants who are presented with learning material matched to their learning styles perform significantly better (on average 12%) than participants presented with learning material that is not matched to their learning styles. The results therefore support the hypothesis H1 as the effectiveness of the tutoring has been improved by adapting to a participant's learning style and that adapting to learning styles has made a difference.

Experiment 2 – MCQ Test Score Improvement

It is good to note that the Oscar tutorial had a positive impact on test scores, with average improvements ranging from 17-22%. Participants in the Mismatch group made a 22% actual improvement in test scores, whereas those in the Adapt and Neutral-Adapt groups improved by 18% and 17% respectively. These results were not as expected, however as the possible improvement has not been taken into account (i.e. participants who achieved 11/12 in the pre-test have less chance of improvement as those achieving 5/12) they do not compare like with like. On further investigation it was found that the Mismatch group had a slightly lower average pre-test score than the other two groups (see Table 10.7), which may explain the higher improvement in scores. Experiment 3 addressed this issue in additionally considering the opportunity for improvement.

The Kruskal-Wallis statistic of 0.35 and p-value of 0.838 show there is no significant difference between groups for this experiment. Therefore the results of Experiment 2 provide no evidence supporting hypothesis H1.

Experiment 3 – MCQ Test Score Improvement/Opportunity

Participants in the Adapt group improved on average 3% more than those in the Mismatch group when considering the opportunity for improvement. This result supports the hypothesis H1that adapting to learning styles improves the effectiveness of tutoring. However, the Kruskal-Wallis statistic of 0.33 and p-value of 0.849 show that the difference is not significant.

Participants in the Neutral-Adapt group improved the least, at 61%, with a sample average of 63% improvement in test scores when considering the opportunity for improvement. Again, the results show that the Oscar tutorial has had a positive effect on participant test scores.

Experiment 4 – MCQ Test Questions Worse

In considering individual test questions where participants performed worse the second time, it can be seen that there is no difference in the average occurrences for the Adapt and Mismatch groups, which are the same as the sample average at 3%, but the Neutral-Adapt group average was 6%. It is possible that this difference is down to chance, as there will always be an element of this in a multiple choice question test. The Kruskal-Wallis statistic of 1.99 and p-value of 0.37 show that there is no significant difference in the average number of participants performing worse across the experimental groups. Thus the results of Experiment 4 do not support hypothesis H1.

Experiment 5 – MCQ Test Questions Better

When considering actual test questions where participants' performance improved, there are small differences in the groups, with the Mismatch average improvement being 4% better than the Adapt group average, and 2% better than the

Neutral-Adapt group and whole sample averages. As in experiment 2, this is explained by the difference in opportunity for improvement across the groups, i.e. the Mismatch group gave more incorrect answers in the pre-test on average than the other groups. The Kruskal-Wallis statistic of 0.64 and p-value of 0.728 indicate there is no significant difference in individual test question improvements across the sample. Therefore, the results of Experiment 5 do not support the hypothesis H1.

Experiment 6 – Session Duration

On comparing the average duration of tutoring sessions across groups, participants in the Mismatch group completed the tutorial in less time than those in the Adapt and Neutral-Adapt groups (287 and 515 seconds; 4.7 and 8.6 minutes respectively) and the whole sample (231 seconds; 3.8 minutes). Differences in duration are expected, as the adaptive system presents an individual learning path for each participant based on their learning styles, their level of knowledge and their discourse during the tutorial. However this experiment aimed to see if there were notable differences in duration in a particular group, and the results show that there are no significant differences in duration. The Kruskal-Wallis statistic of 1.94 and p-value of 0.379 suggest no evidence supporting hypothesis H2. A further analysis grouping participants by learning styles may reveal more significant differences, but even then each participant's learning path will be different.

Experiment 7 – Number of Interactions

There were very few differences in the average number of interactions (i.e. participant discourse acts) during the tutorial across the experimental groups, which ranged from 43 to 46. These results show that despite each individual's learning path being personalised, the tutorial was completed using approximately the same number of interactions in the conversation regardless of the adaptation method adopted. The Kruskal-Wallis statistic of 3.11 and p-value of 0.211 show there is no evidence to support hypothesis H2.

5.3 Participant Evaluation

50 participants completed the feedback questionnaire and the results show that in general Oscar ACITS was well received, understandable and helpful. Table 10.10 shows the total results for questions 1 to 9, which were distributed similarly for all experimental groups except where stated in the following discussion.

Table 10.10. Participant Evaluation Questionnaire Results

Please rate your experience of the following using the scale provided:	SCALE					
	High 6	5	4	3	2	1 Low
1. Instructions	62%	22%	12%	4%	0%	0%
2. Screen layout and design	50%	18%	16%	10%	4%	2%
3. Tutoring	52%	24%	16%	6%	2%	0%
4. How well did Oscar understand you?	36%	20%	18%	20%	6%	0%
5. Did you find the tutoring helpful?	74%	10%	10%	4%	0%	2%
6. Was the conversation natural?	62%	12%	12%	6%	4%	4%
7. Was the conversation frustrating?	42%	10%	10%	8%	8%	22%
8. Do you feel Oscar helped you to revise?	68%	14%	8%	8%	0%	2%
9. Would you use a resource like Oscar:						
a. Instead of attending a face-to-face tutorial?	Yes	46%	No	54%		
b. Instead of learning from a book?	Yes	78%	No	22%		
c. As well as classroom tutoring?	Yes	86%	No	14%		
d. Would you use the resource at all?	Yes	92%	No	8%		

The results in Table 10.10 show that 92% of participants (100% of the Adapt group) rated the tutoring highly (Question 3), with 52% awarding the tutoring the highest rating of 6. The 8% of participants who gave the tutoring a low rating came from the Mismatch and Neutral-Adapt groups. In Question 5, 94% of participants found the tutoring helpful (100% of the Adapt group), with 74% giving the highest rating of 6. In Question 6, 86% of participants rated the tutoring conversation as natural, however in Question 7, 62% of participants found the conversation frustrating. Question 7's result may reflect the unspecific nature of the question, as participants may have been frustrated by their inability to answer questions or remember the topic. In Question 8, 90% of participants felt that Oscar ACITS had helped them to revise. Of the five participants (10%) who did not feel that Oscar ACITS helped them to revise, the participant giving the lowest score came from the Neutral-Adapt adaptation group, with the rest split equally between the Adapt and Mismatch groups.

The results from Question 9, which investigates whether participants would choose to use a resource like the Oscar ACITS, are interesting, with nearly half (46%) of participants stating that they would use Oscar ACITS instead of attending a face-to-face tutorial. Notably fewer of the Mismatch group (36%) answered 'yes' compared to 48% of the Adapt and 56% of the Mismatch groups. 78% of respondents stated that they would use Oscar ACITS instead of reading a book, and 86% of participants (93% of the Adapt group) would use Oscar ACITS to support classroom tutoring. Overall, 92% of participants stated that they would use a resource like Oscar ACITS if it were available. Answers did not vary much between experimental groups, suggesting that different styles of learning material did not significantly affect the user experience. From these results it can be concluded that most people found the Oscar ACITS tutoring helpful, and would use Oscar ACITS to support their studies.

The three remaining questions on the feedback questionnaire were open questions, asking respondents to state what else could be included to assist in learning, three positive and three negative points about using Oscar. Where possible, the answers were grouped into categories, as reported in Table 10.11. Note that in Table 10.11 *n* is the number of participants who answered the question, and not the number of answers given.

Table 10.11. Open Question Results

		n	%
1.	What else could Oscar have done to help you learn?	49	98
	More resources (examples, movies, pictures)	12	24
	More questions	8	16
	Nothing – it's great	8	16
	More detailed explanations	6	12
2.	Please state 3 positive points about using the Oscar computer tutor	47	94
	Good explanation/ revision	25	53
	Easy to use	21	45
	Helpful	17	36
	Human-like	16	34
	Convenient	12	26
	Different/ fun	10	21
3.	Please state 3 negative points about using the Oscar computer tutor	48	96
	GUI	12	25
	Oscar doesn't always understand	10	21
	Not enough help/ feedback	8	17
	Doesn't allow small SQL syntax errors	6	13

When openly asked for comments, 53% of the group commented that Oscar gave good explanations and helped their revision. 45% of respondents stated that Oscar ACITS was easy to use and 36% found Oscar's tutoring helpful, with one participant saying "*the stopping and trying to help before giving you the answer helped me to realise it myself*". 26% of the group liked the convenience of the Oscar ACITS, with 21% finding it different and fun and 34% stated that the tutor was human-like. When asked for negative comments, 21% of respondents noted that Oscar did not always understand their input and 17% thought that there was not enough help or feedback. 13% of the group disliked the fact that Oscar ACITS did not allow small SQL syntax errors! Some quotes from the answers to open questions are shown below:

- "easy to understand, natural, efficient"
- "having a personal tutor all the time, goes at your pace"
- "easy and fun way to learn, available 24/7 and wherever you are"
- "tips when answered incorrectly"
- "doesn't show the answer straight away even if you say don't know"
- "should carry on if you get things wrong and not help as much"
- "gave instant responses"
- "not enough information given"
- "it helped me build up basic statements sequentially"
- "innovative and better than a lecture by far"
- "highlights weak areas"
- "more than one attempt allowed"
- "didn't give me enough chances to answer correctly"
- "it did not allow for slight syntax mistakes"
- "lacks difficulty"

5.4 Experimental Results Summary

The results have shown that the Oscar ACITS adaptation to learning styles was successful in improving participant learning. H1 has been shown to be true, as the results of experiments 1 and 3 show that adapting to participant learning styles during an automated conversational tutorial improves learning gain by 12% and 3% respectively. The results of experiments 6 and 7, relating to H2, are inconclusive as there is no indication that adapting to participant learning styles during the tutorial affects the efficiency (i.e. duration or number of interactions) of learning. It is recognised that in an adaptive ACITS presenting each individual with a personalised learning path, a true comparison of duration or number of interactions is not possible.

The additional evaluation criteria have shown that the Oscar ACITS successfully tutors learners, with an average learning gain of 19% across the sample. However it is recognised that any type of revision exercise is likely to lead to positive learning gain. The results of the participant evaluation showed that the Oscar ACITS was well received, with 94% of learners finding the tutoring helpful and 90% agreeing that Oscar helped them to revise. 92% of the sample said that they would use the Oscar ACITS resource, with 78% saying they would use Oscar instead of learning from a book. A surprising 46% of the sample said they would use Oscar in place of attending face-to-face tutorials. There were no significant differences in feedback between the experimental groups.

In summary, the adaptation to learning styles was successful as experiment 1 showed a statistically significant difference between the learning gain in the Adapt and Mismatch groups.

A comparison with other CITS is not possible, as no other CITS can adapt their tutoring style to match an individual's learning styles.

Chapter 4 described several ITS that adapt to learning styles, however evaluation of the effect of adaptation differs. For example, Sangineto et al. (2007) and Carver et al. (1999) adapt according to the Felder-Silverman model, but use qualitative feedback from a questionnaire to compare differences in non-adaptive and adaptive user experiences. In EDUCE (Kelly and Tangney, 2006), adaptation is to the Multiple Intelligence model (Gardner, 1983), and although learning gain was investigated, the results showed a higher learning gain for learners with mismatched learning material. The Adaptive Web System (Tsianos et al., 2008) adapts to cognitive style and emotion, and was found to improve learning performance.

6 Conclusion

This chapter has presented the implementation of the Oscar ACITS methodology and architecture proposed in chapter 9. The implementation drew on previous work in developing the Oscar PCITS prototype (described in Chapter 7) to speed up development. The analysis of the example learning styles model, the Felder-Silverman model, was described in Chapter 9. The generic algorithm proposed in Chapter 9 was implemented for three of the FS dimensions, completing phase 1 of the development. The implementation of the second phase involved the reuse and modification of a tutoring conversation for SQL revision designed for the Oscar PCITS prototype (Chapter 7). Following phase 2 of the methodology, the tutorial conversation was reorganised, expanded and scored to create a number of versions of the tutoring conversation that adapt to different learning styles. The generic teaching material categories proposed in Chapter 9 were used to aid development. Phase 3 of the methodology, constructing the Oscar ACITS architecture, involved the reuse and adaptation of a number of components developed for the Oscar PCITS prototype.

The resulting prototype Oscar ACITS was then used to experimentally validate the methodology and architecture. A real life study involving real students in a natural learning environment was presented. During the experiment, 72 students were unknowingly assigned to experimental groups according to the adaptation to be applied – the *Neutral-Adapt* and *Adapt* groups experienced a tutorial suited to their learning styles, and the *Mismatch* group a tutorial that did not suit their learning styles. Each group's tutorials were compared in seven experiments designed to test two hypotheses, whether adapting to learning styles improved the effectiveness and efficiency of the learning experience. In general, the Oscar tutorial was well received by students, who improved their test scores by an average of 19%.

The results have shown that there is a marked difference in the achievements of students in the Adapt group to those in the Mismatch group. In experiment 1 the Adapt group performed significantly better (on average 12%) than the Mismatch group during the tutorial, and in experiment 3 the Adapt group improved test scores on average 3% better than the Mismatch group. These results indicate that adapting to an individual's learning styles during tutoring improves learning. The other experiments produced inconclusive results which did not support the hypothesis that adapting to learning styles improved the efficiency of the learning. This hypothesis was difficult to test as, by definition, Oscar's adaptive nature means that student learning paths are individual and therefore cannot be compared.

It is concluded that the adaptation algorithm developed for the Oscar ACITS has made a positive difference in student learning experiences. Students whose tutorial was adapted to match their learning styles performed on average 12% better and improved test scores by an average of 3% more than students who were presented with a tutorial not suited to their learning styles. The 12% better performance by the Adapt group in experiment 1 was shown to be a statistically significant difference. On the whole, the Oscar ACITS tutorial produced positive results, with test scores across the sample an average 19% better following the tutorial.

Therefore, it may be concluded that the real-world experiments described successfully validate the generic Oscar ACITS Methodology and Architecture proposed in Chapter 9.

7 Chapter Highlights

- Prototype Oscar ACITS developed following the generic methodology and architecture from Chapter 9.
- Oscar ACITS delivers an online SQL revision tutorial and implicitly adapts to learning styles (FS model) during tutoring.
- A study was undertaken to validate the Oscar ACITS methodology and architecture proposed in Chapter 9.
- The study involved 72 participants, resulting in 63 completed tutorials and 50 completed evaluation questionnaires.
- 7 experiments investigated whether the adaptation to learning styles improved the effectiveness and efficiency of the tutoring.
- The results show that participants presented with learning material matched to their learning styles performed significantly better (on average 12%) than those whose learning material was not matched to their learning styles.
- The results show a mean improvement in learning gain of 19% following the Oscar ACITS tutorial .
- 94% of participants found the tutoring helpful and 92% stated that they would use Oscar ACITS if it were available.

Chapter 11 Applying the Generic Oscar CITS to Different Models and Domains

1 Introduction

Oscar CITS is an intelligent tutoring system which aims to mimic a human tutor. Oscar CITS directs a mixed initiative tutorial conversation and dynamically predicts and adapts to an individual's learning styles. This thesis has presented novel generic methodologies and architectures for creating an Oscar Predictive CITS (PCITS) and an Oscar Adaptive CITS (ACITS). The methodologies and architecture were validated by implementing prototype Oscar PCITS (Chapter 7) and Oscar ACITS (Chapter 10) systems to deliver a conversational SQL revision tutorial, adopting the Felder-Silverman (1988) learning styles model.

This chapter summarises the steps required to apply the generic methodologies and architectures to other learning styles models and subject domains. A project management course is the example subject domain selected as it is a different to the computing tutorial implemented in Chapter 7 and 8. Project management is a business management discipline involving the planning, organisation and management of resources required to successfully achieve project goals and objectives within specified constraints (Larson and Gray, 2010). As described in Chapter 2, the Honey and Mumford (1992, 2006) learning styles model is often used in business domains, such as project management, and so is the example model selected.

2 Creating an Oscar Predictive CITS

The novel three phase methodology for creating an Oscar Predictive CITS is described in Chapter 6 using the Index of Learning Styles model as an example. Table 6.1 shows the steps of the methodology, and is repeated here in Table 11.1 for reference. In Chapter 7, an implementation of Oscar PCITS which follows this methodology is described, using the ILS model and the subject domain of the database language SQL.

Table 11.1. 3-Phase Methodology for Creating Oscar Predictive CITS.

Phase 1: Create the Learning Styles Predictor Module

1.1. Select a Learning Styles Model
    a. Reduce the learning styles model if necessary
    b. Extract the behaviour characteristics
1.2. Map learning style behaviour to the conversational tutoring style
1.3. Analyse the learning styles model for language traits
1.4. Adapt the generic logic rules to predict learning styles

Phase 2: Design a Tutorial Conversation

2.1. Capture the tutorial scenario and questions (including movies, voice, images, examples, etc.) from human tutors in a specific domain
2.2. Determine the conversational structure/style
2.3. Map tutorial questions onto the generic question styles and templates
2.4. Script Conversational Agent natural language dialogue for each tutorial question using the 3-level model
2.5. Link tutorial dialogue to logic rules through Conversational Agent variables

Phase 3: Construct the PCITS Architecture

The domain of project management was selected to demonstrate the generic methodology as it is a business-related subject and thus different to the engineering-related computing subject adopted in Chapters 7 and 8. The Honey and Mumford (1992, 2006) learning styles model (described in Chapter 2) was selected as it is commonly used in business and management education domains such as project management. Honey and Mumford define four learning styles (Activists, Reflectors, Theorists and Pragmatists) which describe the learning cycle by the four stages of experiencing, reviewing, concluding and planning. Learning styles are assessed using the Learning Styles Questionnaire (LSQ), which are then compared to a list of group norms to categorise the preference as very strong, strong, moderate, low or very low.

The generic methodology to create an Oscar PCITS for a different learning styles model and a different subject domain will now be described.

2.1 Phase 1: Create the Learning Styles Predictor Module

Phase 1 involves the detailed analysis of a learning styles model, and its application to a tutorial conversation. There are many different models of learning styles, and the choice of an appropriate model for the learning environment is critical (see Chapter 2).

- In Step 1.1, once the learning styles model has been selected, it should be reduced if possible and then the typical behaviour characteristics of different learners should be extracted. For example, Theorists are methodical and like structured situations with a clear purpose.

- In Step 1.2, the behaviour characteristics are mapped to the conversational tutoring style, and a list produced of behaviour traits to be captured. For example, Theorists prefer concepts and Pragmatists prefer practical techniques, so behaviour cues such as the score for theoretical and practical questions would be captured.
- In Step 1.3, the learning styles model should be analysed and language traits extracted. For example, the words ‘practical’ and ‘example’ would be mapped to the Pragmatist style.
- In Step 1.4, the generic logic rules would be adapted to select those relevant to the model and add or change rules to cover other requirements. For example, for the two generic logic rules given in Table 6.2 (page 71), example 1 relating to images does not apply to the LSQ and so would be removed, but example 2 has been adapted for the LSQ (as shown in Table 11.2).

Table 11.2. Example Logic Rule to Adjust Learning Style Values Based on Tutoring Conversation.

Example rule to test how comfortable the student is with words and with detail:	
IF	answer is given in the explanation text
AND	student does not know the answer
THEN	increase ACTIVIST;

2.2 Phase 2: Design a Tutorial Conversation

Phase 2 involves capturing the tutorial from expert human tutors and developing a tutorial conversation for the Oscar PCITS.

- In Step 2.1, the tutorial scenario (including syllabus, tests and resources) is captured from human tutors and recorded in a tutorial conversation blueprint document. For example, a tutorial on project management may include a multiple choice test, and a number of examples, visuals and exercises related to the stages of a project.
- In Step 2.2, the 3-level model of a conversation (Figure 6.1) is applied to the conversation script, and a list of Frequently Asked Questions (FAQs) and answers captured from human tutors. For example, ‘What is the waterfall model?’.
- In Step 2.3, tutorial questions are mapped to the generic question styles and templates. For example, the generic question template with choice of approach (Figure 6.3) could be applied to a question about the stages of a project.

- In Steps 2.4 and 2.5, the tutorial conversation recorded in the tutoring conversation blueprint document is converted into conversational agent scripts and linked to the logic rules, as described in Chapter 6, section 3.2.

2.3 Phase 3: Construct the PCITS Architecture

In phase 3 of the methodology, the generic Oscar PCITS architecture (repeated for reference in Figure 11.1) is constructed as described in Chapter 6, section 4. As the Oscar PCITS architecture is modular, all components can be reused. To adapt the system it is only necessary to replace the Learning Styles Predictor (developed in Phase 1) and the Tutorial Knowledge Base and CA scripts (developed in Phase 2).

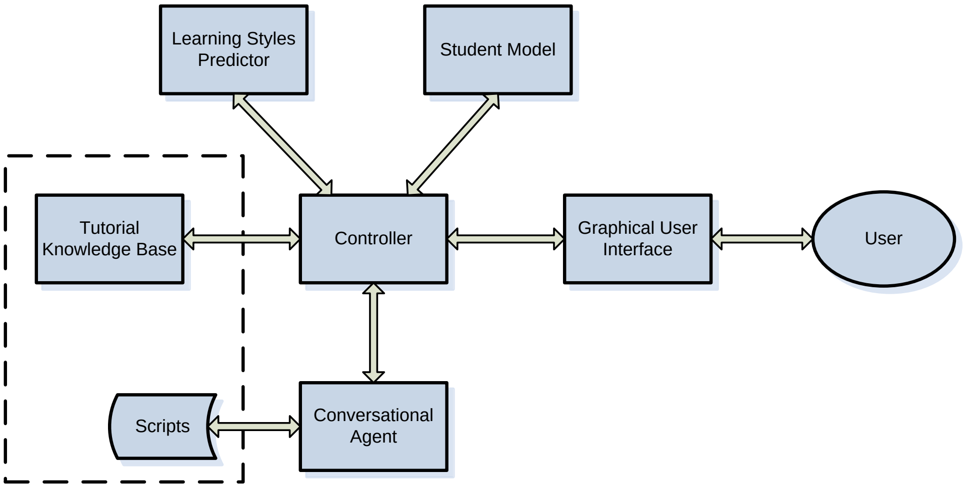


Figure 11.1. Generic Oscar PCITS Architecture

This phase completes the development of an Oscar Predictive CITS for a different learning styles model and subject domain.

3 Creating an Oscar Adaptive CITS

The novel three phase methodology for creating an Oscar Adaptive CITS is described in Chapter 9 using the Index of Learning Styles model as an example. Table 9.1 shows the steps of the methodology, and is repeated here in Table 11.3 for reference. In Chapter 10, an implementation of Oscar ACITS using the ILS model and the subject domain of the database language SQL is described.

Table 11.3. 3-Phase Methodology for Creating Oscar Adaptive CITS.

Phase 1: Create the Learning Styles Adapter Module

1.1. Select a Learning Styles Model and extract the behaviour characteristics
1.2. Map learning style behaviour to associated conversational tutoring style
1.3. Map learning styles to teaching material categories
1.4. Implement the generic adaptation algorithm for chosen learning styles model

Phase 2: Design a Tutorial Conversation

2.1. Capture the tutorial scenario and questions (including movies, voice, images, examples, etc.) from human tutors in a specific domain
2.2. Determine the conversational structure/style
2.3. Map tutorial questions onto the generic teaching material categories
2.4. Score tutorial questions for adaptation to each learning style
2.5. Script Conversational Agent natural language dialogue for each tutorial question using the 3-level model

Phase 3: Construct the ACITS Architecture

The generic methodology to create an Oscar ACITS will now be outlined again, using the Honey & Mumford (1992, 2006) LSQ and the project management domain as examples.

3.1 Phase 1: Create the Learning Styles Adapter Module

Phase 1 involves the selection and analysis of a learning styles model to create the Learning Styles Adapter module. Steps 1.1 and 1.2 are similar but not identical to those steps in the methodology described in section 2.

- In Step 1.1, the typical behaviour characteristics described in the learning styles model should be extracted. For example, Pragmatists like to try out and practice techniques.
- In Step 1.2, the extracted behaviour characteristics are mapped to the preferred conversational tutoring style. For example, Theorists prefer concepts so a matched teaching style would be ‘Present principles rather than examples’.
- In Step 1.3, learning styles should be mapped to the generic teaching material categories given in Table 9.3, and the categories expanded if necessary. For example, Pragmatists like to try out and practice techniques so would be linked to the teaching material category 5, Practical exercises.
- In Step 1.4, the generic adaptation algorithm must be implemented for the selected learning styles model. For example, the LSQ produces scores between 0 and 20 for each of the four dimensions. In the LSQ model, neutral learners with a very low preference for the dimension are classified using norms as the bottom 10% of scores, and so the scoring has different meanings for each dimension. For example, a score of 11 for the Reflector dimension is classed as ‘low preference’

but for the Activist dimensions is classed as ‘strong preference’. However, these differences are easily incorporated into the implementation of the adaptation algorithm, as the only requirements are scores for learning styles and scores for tutorial questions. It is a simple matter to recode the scoring of learning styles, if thought necessary, by grouping scores into the five classes (e.g. ‘low preference’) identified in the model.

3.2 Phase 2: Design a Tutorial Conversation

Phase 2 involves capturing the tutorial from expert human tutors and developing several versions of the tutorial conversation for the Oscar ACITS to incorporate different teaching styles.

- In Step 2.1, the tutorial scenario (including syllabus, tests and resources) is captured from human tutors and recorded in a tutorial conversation blueprint document, as described in section 2.2.
- In Step 2.2, the 3-level model is applied to the conversation script, a list of FAQs and answers captured from human tutors and the generic question styles and templates applied, as described in section 2.2 (Steps 2.2 and 2.3).
- In Step 2.3, tutorial questions are mapped to the generic teaching material categories, resulting in a list of available adaptations for each learning style for each question. For example, a question asking learners to produce a project plan for making and serving a cup of tea would be mapped to the teaching material category 5, Practical exercises.
- In Step 2.4, each question is given a score for each learning style representing the strength of its adaptation, i.e. the number of opportunities for adaptation. For example, a question mapped to category 5, practical exercises would result in a score of 1 for the Pragmatist learning style.
- In Step 2.5, the tutorial conversation recorded in the tutoring conversation blueprint document is converted into conversational agent scripts which are organised by learning style adaptation, as described in Chapter 9, section 4.2.

3.3 Phase 3: Construct the ACITS Architecture

In phase 3 of the methodology, the generic Oscar ACITS architecture (repeated for reference in Figure 11.2) is constructed as described in Chapter 9, section 5. As with the Oscar PCITS, the Oscar ACITS architecture is modular meaning all components can be reused. To adapt the system it is only necessary to replace the Learning Styles Adapter (developed in Phase 1) and the Tutorial Knowledge Base and CA scripts (developed in Phase 2).

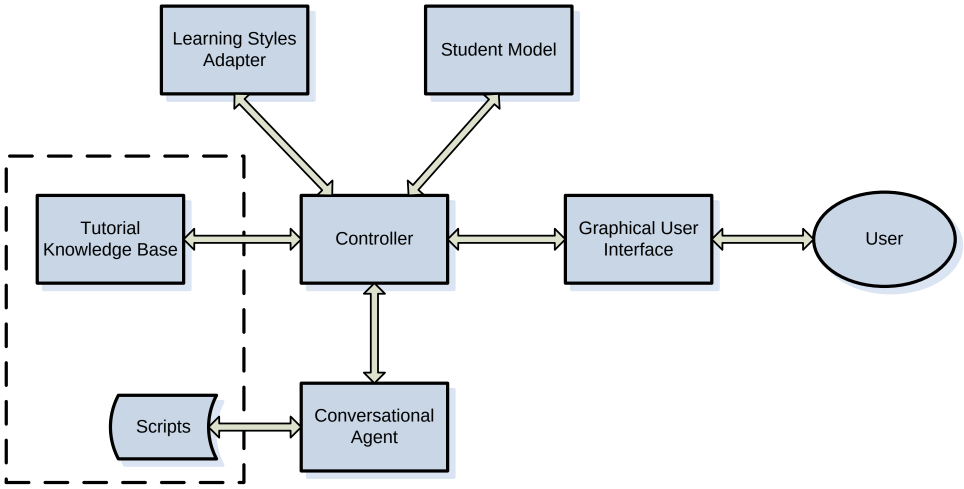


Figure 11.2. Generic Oscar ACITS Architecture

This phase completes the development of an Oscar Adaptive CITS for a different learning styles model and subject domain.

4 Conclusion

The original methodologies and architectures for developing an Oscar Predictive CITS and an Oscar Adaptive CITS are generic, and can easily be applied to different learning styles models and subject domains. This chapter has demonstrated this process using the Honey and Mumford LSQ and the different subject domain of project management as examples. Included in the methodologies are various generic tools, such as generic logic rules, question styles and templates, teaching material categories, which can be applied to other models and domains. Whilst the methodology is easy to apply, a large part of the development time of a CITS is devoted to designing tutorials and creating conversational agent scripts for the tutorial conversation. This is an unavoidably complex, recursive and lengthy process.

The Oscar ACITS adaptation algorithm is generic, and can be applied to any learning styles model where it is possible to score the strength of the individual's preference. The architectures are modular, encouraging the reuse of components to speed up the development of new systems. The analysis of new learning styles models need only be done once, as the learning styles predictor and adapter modules can then be reused with new subject domains.

The original, generic methodologies for developing an Oscar Predictive and Adaptive CITS were empirically validated, as described in Chapters 7, 8 and 10. The results of the experiments show that Oscar CITS can successfully predict and adapt to an individual's learning styles whilst directing a tutoring conversation, and that adapting to learning styles improves learning gain.

5 Chapter Highlights

- Oscar Predictive CITS methodology and architecture are generic and can be applied to new learning styles models and subject domains.
- In developing an Oscar PCITS, the generic logic rules, question styles and templates can be adapted to new models and domains.
- The generic Oscar PCITS architecture can be applied to new models and domains by replacing two components and CA scripts, but reusing other components.
- Oscar Adaptive CITS methodology and architecture are generic and can be applied to new learning styles models and subject domains.
- In developing an Oscar ACITS, the generic teaching material categories can speed up development of learning material.
- The generic Oscar ACITS adaptation algorithm can be applied to any learning styles model where it is possible to score the strength a learner's preference.

Chapter 12 Conclusion

1 Introduction

This chapter concludes the thesis by summarising the work and contributions in relation to the research goal and objectives. The significance and implications of the research are summarised. Finally, recommendations for the direction of future research are given.

2 Summary of the Work

This research has brought together three main areas of research, namely Learning Styles, Conversational Agents (CAs) and Intelligent Tutoring Systems (ITS). In Chapter 2, the current state of learning styles research was reviewed by introducing the different theories on the nature of learning styles, illustrated with reviews of several learning styles models. The key debate is whether learning styles are static or change over time, and Coffield et al. (2004a) concluded that the choice of learning styles model is fundamental. This review highlighted the need for ITS to be independent of the learning styles model selected and informed the design of the generic Oscar CITS. Chapter 3 reviewed two successful text-based conversational agents and described the many challenges of developing CAs which work in real time for extended dialogues and which can adequately mimic humans. The challenge of introducing social behaviour to CAs motivated the modelling of a learning environment, and the development of a CA tutor which can mimic a human tutor by dynamically detecting and adapting to learning styles during a tutoring conversation. Chapter 4 examined the current state of Intelligent Tutoring Systems research and the methods used to personalise tutoring by modelling and adapting to learning styles. The benefits of conducting tutoring through CA interfaces were described, and the finding that no Conversational ITS (CITS) could detect and adapt to learning styles motivated the development of Oscar CITS to fill this gap.

Chapter 5 presented the investigations undertaken in considering the first research question, ‘Is it possible to predict a student’s learning style from a two-way tutoring discourse with a conversational agent tutor?’. Drawing on the review of automatic modelling methods in Chapter 4, knowledge was extracted from an example learning styles model and applied to a natural language tutoring dialogue.

Original strategies were developed for predicting student learning styles from dialogue, including reducing the evaluation instrument (questionnaire), analysing behaviour traits, mapping discriminatory behaviour cues and the use of language. The knowledge captured from this analysis of behaviour and language was then converted into generic logic rules which predict learning styles by incrementing learning styles during the tutoring dialogue.

In Chapter 6 the novel Oscar CITS was proposed, which incorporates intelligent technologies to deliver a tutoring conversation, and dynamically predicts and adapts to student learning styles. A novel, generic 3-phase methodology was proposed for developing an Oscar Predictive CITS (PCITS) for any learning styles model and any subject domain. A generic architecture was proposed for Oscar PCITS which is modular to enable ease of maintenance and reuse of components. The methodology and architecture were validated (as described in Chapter 7) by implementing a prototype Oscar PCITS to intelligently deliver an SQL tutorial and dynamically predict learning styles during the conversation using the Felder-Silverman learning styles model (Felder and Silverman, 1988). Chapter 8 describes several experiments conducted in a real learning environment using real students. The experiments evaluated the success of the Oscar PCITS prototype in predicting learning styles. The experimental results show that Oscar PCITS successfully predicted all eight learning styles in the example model, with an accuracy ranging from 61-100%. The results supported six hypotheses, revealing factors in a conversational tutorial which are indicative of learning style. The results also show that Oscar PCITS was successful in tutoring, with test results improved by an average of 13%, and 89% of participants said they would use Oscar PCITS if it were available. Thus the generic Oscar PCITS Methodology and Architecture were successfully validated empirically in a real learning environment.

Next, the second research question (‘Does adapting to a student’s learning style during a two-way tutoring discourse with a conversational agent tutor improve learning?’) was considered. Chapter 9 proposed an original 3-phase methodology and architecture for developing an Oscar Adaptive CITS (ACITS) independently of a particular learning styles model or subject domain. The novel adaptation algorithm proposed determines the best adaptation for each individual tutorial question based on both the student’s learning style strength and the availability of adaptation for the tutorial question. In this approach, teaching style is varied, the development of learning material may be staged and it is not necessary to provide learning material adapted for all learning styles for every question. The methodology and architecture were validated by implementing and empirically testing a prototype Oscar ACITS as described in Chapter 10. The prototype Oscar ACITS delivered an SQL tutorial and dynamically adapted its teaching style to match participant learning styles during the conversation (using the Felder-Silverman learning styles model). The experiments were also conducted using real students in a real learning environment. The experimental results show that participants experiencing a tutorial matched to their learning styles performed significantly better (12%) and achieved better test score improvements (3%) than those with a mismatched tutorial. The results also show that Oscar ACITS was successful in tutoring as participants improved their test scores by an average of 19%, and 92% of participants stated they would use the Oscar ACITS resource. Therefore the Oscar ACITS was successful in adapting its tutoring to student learning styles, and such adaptation improved learning.

Chapter 11 summarises the generic Oscar PCITS and ACITS methodologies and architectures and shows how they may be applied to a completely different learning styles model (the Honey and Mumford (1992) model) and subject domain (a business studies tutorial on project management).

3 Summary of Contributions

The most significant contribution of this work is the proof of the concept that it is possible to predict student learning styles from a two-way natural language tutoring dialogue with a CITS. Other original contributions of the research include:

- A generic methodology for creating an Oscar Predictive CITS which dynamically predicts learning styles from a natural language tutoring dialogue. The methodology included several generic tools to aid the development of a predictive CITS:
    - Logic rules which match behaviour captured during a natural language tutoring dialogue to learning styles.
    - Question styles and templates which can aid in the development of conversational tutoring scenarios to predict learning styles.
    - 3-level model of a tutorial conversation.

- A general modular architecture for a CITS which dynamically predicts learning styles from a natural language tutoring dialogue.
- A generic methodology for creating an Oscar Adaptive CITS which can dynamically adapt its natural language tutoring style to match individuals' learning styles.
- A dynamic adaptation algorithm which combines both the strength of learning style and the strength of adaptation available for individual tutorial questions to produce the best fitting adaptation per question. It is independent of the learning styles model and the subject domain.
- A general architecture for a CITS which dynamically personalises its tutoring style to suit individuals' learning styles during a natural language tutoring dialogue.
- Two prototype CITS and experimental results which successfully validate the architectures and methodologies for a predictive and adaptive CITS which dynamically personalises tutoring to individuals' learning styles.

These contributions are expected to be of value to researchers and practitioners in the fields of learning styles and conversational intelligent tutoring systems (CITS).

4 Directions for Future Work

The research presented in this thesis does not represent the definitive solution for predicting and adapting to learning styles during a tutorial conversation with a CITS. Rather it proves that it is possible to predict and adapt successfully to learning styles during a tutoring dialogue, and provides a starting point for further research, such as the suggestions below.

- This research has investigated several strategies for the automatic prediction of student learning styles during a tutoring dialogue. The prediction of student learning styles could be extended to additionally model the strength of a student's learning style preference, as in Garcia et al. (2007) who use Bayesian networks to model the strength of student learning styles from ITS behaviour.

One of the barriers to the pervasive use of ITS is their lengthy development time (Murray, 1999). This is especially true of CITS, where the development of intelligent, natural language tutorial conversation scripts represents most of the development time. Two areas of further research could help to address this problem:

- The creation of a courseware authoring tool for developing conversational tutorials could automate the scripting of conversations for CAs, reducing the need for CA scripting expertise. Lesson authoring tools have been created for AutoTutor CITS (Susarla et al., 2003), however AutoTutor does not incorporate different styles of lessons for different learning styles. An authoring tool could also help to standardise the logic of conversational tutorial questions, by adopting and extending the generic question templates proposed in Chapter 6. However it is a complex task to develop an authoring tool which aids but does not restrict the flexibility of conversational tutoring.
- Designing learning resources for portability could allow different CITS to reuse learning resources, thus sharing the development time and cost (Boyle, 2003). Standards exist for sharing learning resources (e.g. the Sharable Content Object Reference Model framework (SCORM, 2004)) whereby standard descriptive Learning Object Metadata (IEEE, 2005) is included to support the retrieval of learning objects from libraries. However, for CITS learning objects must take the specific form of a conversation script, so there is potential for the development of a new standard template which can be applied when developing learning resources for CITS.

An interesting extension to this work would be to extend the conversational user interface to other forms, such as a spoken interface:

- The addition of voice capability to Oscar CITS could further mimic human tutoring, offer increased flexibility and widen access to learning, e.g. for younger children. Also, students value access to learning materials via mobile telephones for use when travelling and when there is no access to computers (Bradley et al., 2009). The introduction of speech recognition to AutoTutor CITS (D'Mello et al., 2010a) did not improve learning gain, but more content was covered (as speaking is faster than typing). Current speech recognition technology is not yet at a level where groups of learners can converse with a CITS in a natural learning environment as with a human tutor. However, working versions of speech recognition software have been produced when trained to recognise an individual voice, e.g. Dragon NaturallySpeaking (Nuance Communications Inc., 2011). Using individually trained software, speech recognition could be accomplished for use on personal computers or mobile phones which locally convert speech to text. However, for wider access and use in natural learning environments, which are often noisy, speech recognition errors are common and D'Mello et al. (2010a) reported that these errors negatively affected student feedback about the tutoring.

5 Overall Conclusions

In conclusion, this thesis reviewed the current state of research in learning styles theories, text-based conversational agents and intelligent tutoring systems. The findings indicated that although conversational tutoring was better able to model the social process of learning, no conversational intelligent tutoring systems (CITS) could automatically detect and adapt the tutoring to an individual's learning styles.

The Oscar CITS was proposed, which delivers a personalised natural language tutorial conversation incorporating intelligent solution analysis and problem solving support. Oscar CITS has the flexibility to use different pedagogical styles matched to a rich student model which is continuously updated during tutoring by dynamically detecting individual's learning styles. The Oscar CITS flexible adaptation algorithm combines the strength of student preference with the availability of adaptive learning material to determine the best fitting adaptation for each tutorial question.

A methodology and architecture for creating an Oscar Predictive CITS and Adaptive CITS were presented, which are independent of the learning styles model and subject domain. This enables a free choice of the learning styles model best suited to the subject domain, which Coffield et al. (2004a) found to be of fundamental importance.

To date, there have been few CITS developed. It is hoped that the results of this research (e.g. the generic 3-phase methodology for developing Oscar CITS) can be used by researchers and practitioners to create adaptive intelligent tutoring systems which can mimic human tutors by delivering personalised conversational tutorials.

References

ALICE AI Foundation Inc. (2007). *A.L.I.C.E. Artificial Intelligence Foundation*. [Online] [Accessed on 7 June 2011] http://www.alicebot.org/about.html

Allinson C. and Hayes J. (1996) 'The cognitive style index: a measure of intuition analysis for organizational research' *Journal of Management Studies*, Vol. 33, pp. 119-135

Ammar, M. B., Neji, M., Alimi, A. M. and Gouarderes, G. (2010) 'The Affective Tutoring System' *Expert Systems with Applications*, Vol. 37, pp. 3013-3023

Arnott, E., Hastings, P. and Allbritton, D. (2008) 'Research Methods Tutor: Evaluation of a dialogue-based tutoring system in the classroom' *Behavior Research Methods*, Vol. 40 no. 3, pp. 694-698

Avon and Somerset Constabulary (no date). *Avon and Somerset Constabulary - Welcome.* [Online] [Accessed on 16 June 2011] http://www.avonandsomerset.police.uk/

Bajraktarevic, N., Hall, W. and Fullick, P. (2003) 'Incorporating Learning Styles in Hypermedia Environment: Empirical Evaluation' *In Proceedings of the Fourteenth Conference on Hyper-text and Hypermedia.* Nottingham, 2003. de Bra, P., Davis, H.C., Kay, J. and Schraefel, M. (eds.), Nottingham, pp. 41-52

Bales, R.F. (1950) *Interaction process analysis: A method for the study of small groups*. Cambridge, MA: Addison-Wesley.

Becker, C., Kopp, S. and Wachsmuth, I. (2007) 'Why Emotions should be Integrated into Conversational Agents' *In* Nishida, T. (ed.) *Conversational Informatics: An Engineering Approach.* John Wiley and Sons Ltd., Chapter 3, pp. 49-68

Boyle, T. (2003) 'Design principles for authoring dynamic, reusable learning objects' *Australian Journal of Educational Technology*, Vol. 19 no. 1, pp. 46-58

Boyle, T. (2010) 'Layered learning design: Towards an integration of learning design and learning object perspectives' *Computers & Education*, Vol. 54, pp. 661-668

Bradley, C., Haynes, R., Cook, J., Boyle, T. and Smith, C. (2009) 'Design and Development of Multimedia Learning Objects for Mobile Phones' *In* Ally, M. (ed.) *Mobile Learning: Transforming the Delivery of Education and Training*. Athabasca University, Canada: AU Press, pp. 157–182

Brooks, C., Greer, J., Melis, E. and Ullrich, C. (2006) 'Combining ITS and eLearning Technologies: Opportunities and Challenges' *Lecture Notes in Computer Science*, Vol. 4053. Berlin: Springer-Verlag, pp. 278-287

Brusilovsky, P. (1996) 'Methods and Techniques of Adaptive Hypermedia' *User Modeling and User-Adapted Interaction*, Vol. 6 no. 2-3, pp. 87-129

Brusilovsky, P. and Henze, N. (2007) 'Open Corpus Adaptive Educational Hypermedia' *In* Brusilovsky, P., Kobsa, A., Neidl, W. (eds.) *The Adaptive Web: Methods and Strategies of Web Personalization*. *Lecture Notes in Computer Science*, Vol. 4321. Berlin: Springer-Verlag, pp. 671-696

Brusilovsky, P. and Millan, E. (2007) ‘User Models for Adaptive Hypermedia and Adaptive Educational Systems’ *In* Brusilovsky, P., Kobsa, A., Neidl, W. (eds.): *The Adaptive Web: Methods and Strategies of Web Personalization*. *Lecture Notes in Computer Science,* Vol. 4321. Berlin: Springer-Verlag, pp. 3-53

Brusilovsky, P. and Peylo, C. (2003) ‘Adaptive and Intelligent Web-based Educational Systems’ *International Journal of Artificial Intelligence in Education,* Vol. 13, pp. 156-169

Carpenter, R. (2007). *jabberwacky – live chat bot.* [Online] [Accessed on 20 June 2007] http://www.jabberwacky.com/

Carver, C.A., Howard, R.A. and Lane, W.D. (1999) ‘Enhancing Student Learning Through Hypermedia Courseware and Incorporation of Learning Styles’ *IEEE Transactions on Education*, Vol. 42 no. 1, pp. 33-38

Cassell, J. (2000) *Embodied Conversational Agents*. Cambridge, MA: MIT Press.

Cha, H. J., Kim, Y. S., Park, S. H., Yoon, T. B., Jung, Y. M. and Lee, J. H. (2006) ‘Learning styles diagnosis based on user interface behaviours for the customization of learning interfaces in an intelligent tutoring system’ *Lecture Notes in Computer Science,* Vol. 4053, pp. 513-524

Chen, S.Y. and Liu, X. (2008) ‘An Integrated Approach for Modeling Learning Patterns of Students in Web-Based Instruction: A Cognitive Style Perspective’ *ACM Transactions on Computer-Human Interaction,* Vol. 15 no.1, Article 1

Chi, M.T.H., Siler, S., Jeong, H., Yamauchi and T., Hausmann, R.G. (2001) ‘Learning from human tutoring’ *Cognitive Science*, Vol. 25, pp. 471-533

Coffield F., Moseley D., Hall E. and Ecclestone K. (2004a) *Learning Styles and Pedagogy in Post-16 Learning: A Systematic and Critical Review*. London: Learning and Skills Research Center.

Coffield F., Moseley D., Hall E. and Ecclestone K. (2004b) *Should we be using Learning Styles: What research has to say in practice*. London: Learning and Skills Research Center.

Convagent Ltd. (2005) *Convagent.* [Online] [Accessed on 17 May 2011] http://www.convagent.com/

Cooper, C. (2002) *Individual Differences*. London, UK: Arnold.

Crockett, K., Bandar, Z, O’Shea, J and Mclean, D. (2009) ‘Bullying and Debt: Developing Novel Applications of Dialogue Systems’ *In* International Joint Conferences on Artificial Intelligence. *Knowledge and Reasoning in Practical Dialogue Systems*. Pasadena: AAAI Press, pp. 1–9

Crockett, K., Latham, A., Mclean, D., Bandar, Z. and O’Shea, J. (2011) ‘On Predicting Learning Styles in Conversational Intelligent Tutoring Systems using Fuzzy Classification Trees’ *In* IEEE. *IEEE International Conference on Fuzzy Systems,* Taipei, Taiwan, June 27-30. IEEE, pp. 2481-2488

Curry, L. (1983) ‘An organization of learning styles theory and constructs’ ERIC document no. ED 235 185.

D’Mello, S.K., Craig, S.D., Fike, K. and Graesser, A. (2009) ‘Responding to learners’ cognitive-affective states with supportive and shakeup dialogues’ *In* Jacko,

J. (ed.) *Human-computer interaction. Ambient, ubiquitous and intelligent interaction,* Berlin: Springer, pp. 595-604

D'Mello, S.K., Craig, S.D., Witherspoon, A., McDaniel, B. and Graesser, A. (2008) 'Automatic detection of learner's affect from conversational cues' *User Modelling and User-Adapted Interaction,* Vol. 18, pp. 45-80

D'Mello, S.K., Graesser, A. and King, B. (2010a) 'Toward Spoken Human-Computer Tutorial Dialogues' *Human-Computer Interaction*, Vol. 25, pp. 289-323

D'Mello, S.K., Lehman, B., Sulins, J., Daigle, R., Combs, R., Vogt, K., Perkins, L. and Graesser, A. (2010b) 'A Time for Emoting: When Affect-Sensitivity Is and Isn't effective at Promoting Deep Learning' *Lecture Notes in Computer Science*, Vol. 6094. Berlin: Springer-Verlag, pp. 245-254

Dag, F. and Gecer, A. (2009) 'Relations between online learning and learning styles' *Procedia Social and Behavioral Sciences*, Vol. 1, pp. 862-871

Department for Education and Skills (2004a) *Pedagaogy and Practice: Teaching and Learning in Secondary Schools. Unit 1: Structuring learning.* DfES 0424-2004 G, Crown copyright.

Department for Education and Skills (2004b) *Pedagaogy and Practice: Teaching and Learning in Secondary Schools. Unit 4: Lesson design for inclusion*. DfES 0427-2004 G, Crown copyright.

Dunn, R., and Dunn, K. (1974) 'Learning Style as a Criterion for Placement in Alternative Programs' *Phi Delta Kappan,* Vol. 56 no. 4, pp. 275-278

Dunn, R. and Dunn, K. (1978) *Teaching Students through their individual learning styles: A practical approach*. Prentice Hall, Reston Publishing.

Dunn, R., Dunn, K., and Price, G. E. (1996) *Learning Style Inventory*. Lawrence, KS: Price Systems.

Dunn, R., and Griggs, S. A. (2000) *Practical Approaches to Using Learning Styles in Higher Education*. Bergin and Garvey.

Dunn, R. and Griggs, S. (2003) *Synthesis of the Dunn and Dunn learning styles model research: who, what, when, where and so what – the Dunn and Dunn learning styles model and its theoretical cornerstone.* New York: St John's University.

Dzikovska, M.O., Bental, D., Moore, J.D., Steinhauser, N.B., Campbell, G.E., Farrow, E. and Callaway, C.B. (2010) 'Intelligent tutoring with Natural Language Support in Beetle II System' *Lecture Notes in Computer Science,* Vol. 6383 Berlin: Springer-Verlag, pp. 620-625

El Bachari, E., Abdelwahe, E.H. and El Adnani, M. (2010) 'Design of an Adaptive E-Learning Model based on Learner's Personality' *Ubiquitous Computing and Communication Journal*, Vol. 5 no. 3

Entwistle, N.J. (1981) *Styles of Learning and Teaching*. New York: Wiley.

Entwistle, N. (1988) 'Motivational Factors in Students' Approaches to Learning' *In* Schmeck, R.R. (ed.), *Learning Strategies and Learning Styles.* New York, N.Y.: Plenum Press, Chapter 2.

Entwistle, N. J. (1997) *The Approaches and Study Skills Inventory for Students (ASSIST).* Edinburgh: Centre for Research on Learning and Instruction, University of Edinburgh.

Entwistle, N. J. (1998) 'Improving Teaching through Research on Student Learning' *In* Forrest, J.J.F. (ed.) *University Teaching: International Perspectives*. New York: Routledge Falmer Press, pp. 73-112

Entwistle, N.J., McCune, V., and Walker, P. (2001) 'Conceptions, Styles and Approaches within Higher Education: Analytic Abstractions and Everyday Experience' *In* Sternberg, R.J. and Zhang, L.F. (eds.) *Perspectives on Thinking, Learning and Cognitive Styles*. Mahwah, New Jersey: Lawrence Erlbaum, pp. 103-136

Felder, R. M. and Brent, R. (2005) 'Understanding Student Differences' *Journal of Engineering Education*, Vol. 94 no. 1, pp. 57-72

Felder, R. M. and Henriques, E.R. (1995) 'Learning and Teaching Styles in Foreign and Second Language Education' *Foreign Language Annals,* Vol. 28 no. 1, pp. 21–31

Felder, R. M. and Silverman, L. K. (1988) 'Learning and teaching styles in engineering education' *Engineering Education,* Vol. 78 no. 7, pp. 674–681. Preceded by a preface in 2002 [Online] [Accessed on 26 April 2008] http://www.ncsu.edu/felderpublic/ Papers/LS-1988.pdf

Felder, R. M. and Soloman, B. A. (2008) *Index of Learning Styles questionnaire*. [Online] [Accessed on 26 April 2008] http://www.ncsu.edu/felder-public/ILSpage.html

Felder, R. M. and Spurlin, J. (2005) 'Applications, Reliability and Validity of the Index of Learning Styles' *International Journal on Engineering Education,* Vol. 21 no. 1, pp. 103-112

Ford, N. and Chen, S.Y. (2000) 'Individual differences, hypermedia navigation and learning: an empirical study' *Journal of Educational Multimedia and Hypermedia*, Vol. 9 no. 4, pp. 281-312

Friesen, N. (2009) *Re-Thinking E-Learning Research: Foundations, Methods and Practices.* New York: Peter Lang Publishing, Inc.

Furnham, A. and Medhurst, S. (1995) 'Personality Correlates of Academic Seminar Behaviour: A Study of Four Instruments' *Personality and Individual Differences,* Vol. 19 no. 2, pp. 197-208

Garcia, P., Amandi, A., Schiaffino, S. and Campo, M. (2007) 'Evaluating Bayesian networks' precision for detecting students' learning styles' *Computers & Education,* Vol. 49, pp. 794–808

Garcia, P., Schiaffino, S. and Amandi, A. (2008) 'An enhanced Bayesian model to detect students' learning styles in Web-based courses' *Journal of Computer Assisted Learning*, Vol. 24, pp. 305-315

Gardner, H. (1983). *Frames of mind: The Theory of Multiple Intelligences*, 1st ed. New York: Basic Books.

Gardner, H. (1993). *Frames of mind*, 2nd ed. New York: Basic Books.

Graesser, A. C., Chipman, P., Haynes, B.C. and Olney, A. (2005c) 'AutoTutor: An Intelligent Tutoring System with Mixed-Initiative Dialogue' *IEEE Transactions on Education*, Vol. 48 no. 4, pp. 612-618

Graesser, A. C., Jeon, M., and Dufty, D. (2008b) 'Agent technologies designed to facilitate interactive knowledge construction' *Discourse Processes*, Vol. 45, pp. 298-322

Graesser, A. C., Lu, S., Jackson, G. T., Mitchell, H., Ventura, M., Olney, A., and Louwerse, M. M. (2004) 'AutoTutor: A tutor with dialogue in natural language' *Behavior Research Methods, Instruments, and Computers*, Vol. 36, pp. 180-193

Graesser, A. C., McDaniel, B., Chipman, P., Witherspoon, A., D'Mello, S. K. and Gholson, B. (2006) 'Detection of emotions during learning with AutoTutor' *In Proceedings of the 28th Annual Meetings of the Cognitive Science Society*, Erlbaum, pp. 285-290

Graesser, A. C., Moreno, K., Marineau, J., Adcock, A., Olney, A., Person, N. and The Tutoring Research Group (2003) 'AutoTutor improves deep learning of computer literacy: Is it the dialogue or the talking head?' *In* Hoppe, U., Verdejo, F. and Kay, J. (eds.), *Proceedings of Artificial Intelligence in Education*, Amsterdam: IOS Press, pp.47-54

Graesser, A. C., Wiemer-Hastings, K., Wiemer-Hastings, P., Kreuz, R., and the Tutoring Research Group. (1999) 'AutoTutor: A simulation of a human tutor' *Cognitive Systems Research*, Vol. 1, pp. 35-51

Graesser, A.C., D'Mello, S.K., Craig S.D., Witherspoon, A., Sullins, J., McDaniel, B. and Gholson, B. (2008a) 'The Relationship Between Affective States and Dialog Patterns During Interactions With AutoTutor' *Journal Of Interactive Learning Research*, vol. 19 no. 2, pp. 293-312

Graesser, A.C., Hu, X. and McNamara, D.S. (2005b) 'Computerized learning environments that incorporate research in discourse psychology, cognitive science and computational linguistics' *In* Healy, A.F. (ed.), *Experimental cognitive psychology and its applications: Fetschrift in honor of Lyle Bourne, Walter Kintsch and Thomas Landauer.* Washington D.C.: American Psychological Association, pp. 183-194.

Graesser, A.C., Person, N., Lu, Z., Jeon, M.G. and McDaniel, B. (2005a) 'Learning While Holding a Conversation With a Computer' *In Technology-Based Education: Bringing Researchers and Practitioners Together.* USA: Information Age Publishing Inc., pp. 143-167

Graf, S., Kinshuk and Liu, T.C. (2009) 'Supporting Teachers in Identifying Students' Learning Styles in Learning management Systems: An Automatic Student Modelling Approach' *Educational Technology & Society*, Vol. 12 no. 4, pp. 3-14

Graf, S., Viola, S. R., Kinshuk and Leo, T. (2007) 'In-depth analysis of the Felder-Silverman learning style dimensions' *Journal of Research on Technology in Education*, Vol. 40 no. 1, pp. 79-93

Graff, M. (2003) 'Assessing Learning from Hypertext: An Individual Differences Perspective' *Journal of Interactive Learning Research*, Vol. 14 no. 4, pp. 425-438

Honey, P. and Mumford, A. (1992) *The manual of learning styles*, Maidenhead: Peter Honey Publications Ltd.

Honey, P. and Mumford, A. (2006), *The Learning Styles Questionnaire: 80-item version*, Maidenhead: Peter Honey Publications Ltd.

Hsu, C.C., Wang, K.T. and Huang, Y.M. (2010) 'Modeling Personalized Learning Styles in a Web-Based Learning System' *Lecture Notes in Computer Science*, Vol. 6250. Berlin: Springer-Verlag, pp. 12-21

Hsieh, S.W., Jang, Y.R., Hwang, G.J. and Chen, N.S. (2011) 'Effects of teaching and learning styles on students' reflection levels for ubiquitous learning' *Computers & Education*, vol. 57, pp. 1194-1201

IEEE (2005) *IEEE LTSC | WG12 | Final LOM Draft Standard*. [Online] [Accessed on13 June 2011] http://ltsc.ieee.org/wg12/20020612-Final-LOM-Draft.html

Inter IKEA Systems BV (2004), *IKEA | Home.* [Online] [Accessed on 20 July 2007] http://www.ikea.com/gb/en/

Karagiannidis C. and Sampson D. (2004) 'Adaptation rules relating learning styles research and learning objects metadata' *In* Magoulas G. and Chen S. (eds.) *Proceedings of the Workshop on Individual Differences in Adaptive Hypermedia in AH2004*. Vol. 1. Eidhoven, Netherlands. pp. 60-69

Kelly, B. and Tangney, D. (2004) 'Predicting Learning Characteristics in a Multiple Intelligence Based Tutoring System' *Lecture Notes in Computer Science,* Vol. 3220. Berlin: Springer-Verlag, pp. 678-688

Kelly, B. and Tangney, D. (2005) 'Matching and Mismatching Learning Characteristics with Multiple Intelligence Based Content' *In Proceedings of AI in Education, AIED '05*. Amsterdam, Netherlands. pp. 354-361

Kelly, D. and Tangney, B. (2006) 'Adapting to intelligence profile in an adaptive educational system' *Interacting with Computers*, Vol. 18, pp. 385-409

Khoury, R., Karray, F., Kamel, M.S. (2008) 'Keyword extraction rules based on a part-of-speech hierarchy' *International Journal in Advanced Media and Communication*, Vol. 2 no. 2, pp. 138-153

Klasnja-Milicevic, A., Vesin, B., Ivanovic, M. and Budimac, Z. (2011) 'E-Learning personalization based on hybrid recommendation strategy and learning style identification' *Computers & Education,* Vol. 56, pp. 885-899

Kolb, D. A. (1976) *The Learning Styles Inventory: Technical manual*, Boston: McBer & Company.

Kruskal, W. and Wallis, W.A. (1952) 'Use of ranks in one-criterion variance analysis' *Journal of the American Statistical Association*, Vol. 47 no. 260, pp. 583-621

Kumar, R., Ai, H., Beuth, J.L. and Rose, C.P. (2010) 'Socially-capable Conversational Tutors can be Effective in Collaborative-Learning situations' *Lecture Notes in Computer Science* Vol. 6094. Berlin: Springer-Verlag, pp. 156-164

Larson, E.W. and Gray, C.F. (2010) *Project Management: The Managerial Process*. McGraw-Hill Higher Education.

Latham, A., Crockett, K. and Bandar, Z. (2010) 'A Conversational Expert System Supporting Bullying and Harassment Policies' *In* INSTICC *2nd*

International Conference on Agents and Artificial intelligence, ICAART 2010. Valencia, Spain, 22-24 January 2010. INSTICC, pp. 163-168.

Latham, A., Crockett, K., McLean, D. and Edmonds, B. (2009) 'Using Learning Styles to Enhance Computerised Learning Systems' *In* Manchester Metropolitan University *Annual Research Student Conference 2009.* Manchester Metropolitan University, Manchester, UK.

Lee, C.H., Evens, M.W. and Glass, M.S. (2004) 'The effect of cognitive styles on a visually-oriented task in online learning environment' *In Dialog Based Intelligent Tutoring Systems: State of the Art and New Research Directions, ITS 2004.* Alagoas, Brazil, August 30-September 3 2004. pp. 15-22

Lee, S.I., Sung, C. and Cho, S.B. (2001) 'An Effective Conversational Agent with User Modeling Based on Bayesian Network' *In Web Intelligence: Research and Development : First Asia-Pacific Conference*.: Springer Berlin: pp. 428-432

Leontidis, M. and Halatsis, C. (2009) 'Integrating Learning Styles and Personality Traits into an Affective Model to Support Learner's Learning'. *Lecture Notes in Computer Science*, Vol. 5686. Berlin: Springer-Verlag, pp. 225-234

Li, F.W.B., Lau, R.W.H., Dharmendran, P. (2010) 'An Adaptive Course Generation Framework' *Journal Of Distance Education Technologies,* Vol. 8 no. 3, pp 47-64

Li, Y., Bandar, Z., McLean, D., O'Shea, J. (2004) 'A Method for Measuring Sentence Similarity and its Application to Conversational Agents' *In 17th International FLAIRS Conference, FLAIRS 2004.* Miami Beach, Florida. AAAI Press, pp. 820-825

Loebner, H.G. (2011) *Home Page of the Loebner Prize*. [Online] [Accessed on 7 June 2011] http://www.loebner.net/Prizef/loebner-prize.html

Ma, C., Predinger, H. and Ishizuka, M. (2005) 'A Chat System Based on Emotion Estimation from Text and Embodied Conversational Messengers' *Lecture Notes in Computer Science*, Vol. 3711. Berlin: Springer-Verlag, pp. 535-538

Mairesse, F., Walker, M., Mehl, M. and Moore, M. (2007) 'Using Linguistic Cues for the Automatic Recognition of Personality in Conversation and Text' *Journal of Artificial Intelligence Research,* Vol. 30, pp 457-501

Massa, L.J. and Mayer, R.E. (2006) 'Testing the ATI hypothesis: Should multimedia instruction accommodate verbalizer-visualizer cognitive style?' *Learning & Individual Differences*, Vol. 16, pp. 321-336

Melis, E., Andrès, E., Büdenbender, J., Frishauf, A., Goguadse, G., Libbrecht, P., Pollet, M. and Ullrich, C. (2001) 'ActiveMath: A web-based learning environment' *International Journal of Artificial Intelligence in Education*, Vol. 12 no. 4, pp. 385-407

Michie, D. (2001) 'Return of the Imitation Game' *Electronic Transactions on Artificial Intelligence* Vol. 6, pp. 203-221

Michie, D. and Sammut, C. (2001) *Infochat Scripter's Manual.* Manchester, UK: ConvAgent Ltd.

Mitrovic, A. (2003) 'An Intelligent SQL Tutor on the Web' *International Journal of Artificial Intelligence in Education*, Vol. 13, pp. 171-195

Moodle (2011). *Moodle.org:open source community-based tools for learning.* [Online] [Accessed on 23 June 2011] http://www.moodle.org

Moreno, R. Mayer, R.E., Spires, H. and Lester, J. (2001) ‘The case for social agency in computer-based teaching: Do students learn more deeply when they interact with pedagogical agents?’ *Cognition and Instruction*, vol. 19, pp. 177-213

Murray, T. (1999) ‘Authoring intelligent tutoring systems: An analysis of the state of the art’ *International Journal of Artificial Intelligence in Education,* Vol. 10, pp. 98-129

Myers, I.B. (1962) *Manual: The Myers-Briggs Type Indicator*. Palo Alto, CA: Consulting Psychologists Press.

Myers, I.B. and McCaulley, M.H. (1998) *Manual: a guide to the development and use of the Myers-Briggs Type Indicator.* Palo Alto, CA: Consulting Psychologists Press.

Nuance Communications Inc. (2011) *Nuance*. [Online] [Accessed on 10 June 2011] http://www.nuance.com/

O’Shea, J., Bandar, Z. and Crockett, K. (2011) ‘Systems Engineering and Conversational Agents’ *In* Tolk, A. and Jain, L.C. (eds.) *Intelligence-Based Systems Engineering, Intelligent Systems Reference Library 10.* Berlin: Springer-Verlag.

O’Shea, K., Bandar, Z. and Crockett, K. (2010) ‘A Conversational Agent Framework using Semantic Analysis’ *International Journal of Intelligent Computing Research,* Vol. 1 no. 1/2, pp. 23-32

Ovesdotter Alm, C, Roth D and Sproat R (2005) ‘Emotions from text: machine learning for text-based emotion prediction’ *In HLT '05 Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing.* Vancouver, Canada, 6-8 October 2005. ACL Anthology, pp. 579-586

Owda, M., Bandar, Z. and Crockett, K. (2011) ‘Information Extraction for SQL Query Generation in the Conversation-Based Interfaces to Relational Databases (C-BIRD)’ *Lecture Notes in Artificial Intelligence,* Vol. 6682. Berlin: Springer-Verlag, pp. 44-53

Ozpolat, E. and Akar, G.B. (2009) ‘Automatic detection of learning styles for an e-learning system’ *Computers & Education*, Vol. 53, pp. 355-367

Papanikolaou, K. A., Grigoriadou, M. , Kornilakis, H. and Magoulas, G. D. (2003) ‘Personalizing the Interaction in a Web-based Educational Hypermedia System: the case of INSPIRE’ *User Modeling and User-Adapted Interaction*, Vol. 13, pp. 213-267

Papanikolaou, K. A., Mabbott , A., Bull, S., Grigoriadou, M. (2006) ‘Designing learner-controlled educational interactions based on learning/cognitive style and learner behavior’, *Interacting with Computers,* Vol. 18, no. 3, pp. 356-384.

Parades, P. and Rodriguez, P. (2004) ‘A Mixed Approach to Modelling Learning Styles in Adaptive Educational Hypermedia’ *Advanced Technology for Learning*, Vol. 1 no. 4, pp. 210-215

Popescu, E. (2008) ‘An Artificial Intelligence Course Used to Investigate Students’ Learning Style’ *Lecture Notes in Computer Science,* Vol. 5145. Berlin: Springer-Verlag, pp. 122-131

Popescu, E. (2009) ‘Diagnosing Students’ Learning Style in an Educational Hypermedia System’ *In* Mourlas, C., Tsianos, N. and Germanakos, P. (eds) *Cognitive and Emotional Processes in Web-based Education: Integrating Human Factors and Personalization, Advances in Web-Based Learning Book Series*. IGI Global, pp. 187-208

Popescu, E. (2010) ‘Adaptation provisioning with respect to learning styles in a Web-based educational system: an experimental study’ *Journal of Computer Assisted Learning*, Vol. 26, pp. 243-257

Popescu, E., Trigano, P. and Badica, C. (2007) ‘Towards a unified Learning Styles Model in Adaptive Educational Systems’ *In* IEEE *International Conference in Advanced Learning Technologies, ICALT 2007.* IEEE Computer Society Press, pp. 804-808

Rahati, A. and Kabanza, F. (2010) ‘Persuasive Dialogues in an Intelligent tutoring System for Medical Diagnosis’ *Lecture Notes in Computer Science*, Vol. 6095. Berlin: Springer-Verlag, pp. 51-61

Rasmussen, K.L. (1998) ‘Hypermedia and learning styles: can performance be influenced?’ *Journal of Multimedia and Hypermedia*, Vol. 7 no. 4, pp. 291-308

Riding, R. and Grimley, M. (1999) ‘Cognitive style and learning from multimedia materials in 11-year children’ *British Journal of Educational Technology*, Vol. 30 no. 1, pp. 43-56

Rundle, S. M., and Dunn, R. (2000) *The Guide to Individual Excellence: A Self Directed Guide to Learning and Performance Solutions*. New York: Performance Concepts International.

Sadek, D. (1999) ‘Design Considerations on Dialogue Systems: from theory to technology – the case of Artimis’, *In ESCA Workshop: Interactive Dialogue in Multi-modal Systems.* Irsee: Kloster, p. 173

Sammut, C. (2001) ‘Managing Context in a Conversational Agent’ *Linkoping Electronic Articles in Computer & Information Science*, Vol. 3 no. 7. Sweden: Linkoping University Electronic Press

Sancho, P., Martinez, I. and Fernández-Manjón, B. (2005) ‘Semantic Web Technologies Applied to e-learning Personalization in <e-aula>’ *Journal of Universal Computer Science*, Vol. 11 no. 9, pp. 1470-1481

Sanders, D.A. and Bergasa-Suso, J. (2010) ‘Inferring Learning Style from the way Students Interact with a Computer User Interface and the WWW’ *IEEE Transactions on Education*, Vol. 53 no. 4, pp.613-620

Sangineto, E., Capuano, N., Gaeta, M., and Micarelli, A. (2007) ‘Adaptive course generation through learning styles representation’ *Journal of Universal Access in the Information Society*, Vol. 7 no. 1, pp. 1-23

SCORM (2004). *Home SCORM®.* [Online] [Accessed on 13 June 2011] http://www.adlnet.gov/Technologies/scorm/default.aspx

Sharples, M. (2005) ‘Learning As Conversation: Transforming Education in the Mobile Age’ *In Conference on Seeing, Understanding, Learning in the Mobile Age.* Budapest, Hungary, April 2005. pp. 147-152

Shih, C. and Gamon, J.A. (2002) ‘Relationships among learning strategies, patterns, styles, and achievement in webbased courses’ *Journal of Agricultural Education,* Vol. 43 no. 4, pp. 1-11

Smith, W., Sekar, S. and Townsend, K. (2002) ‘The impact of surface and reflective teaching and learning on student academic success’ *In* Valcke, M. and Gombeir, D. (eds) *Learning styles: reliability and validity*, *Proceedings of the 7th Annual European Learning Styles Information Network Conference.* 26–28 June 2002, Ghent. Ghent: University of Ghent, pp. 407-418

Spallek, H. (2003) ‘Adaptive Hypermedia: A New Paradigm for Educational Software’ *Advances in Dental Research*, Vol. 17 no. 1, pp. 38-42

Stash, N. and De Bra, P. (2004) ‘Incorporating cognitive styles in AHA! (The adaptive hypermedia architecture)’ *In 2004 IASTED International Conference.* Calgary, Canada: ACTA Press, pp. 378-383

Stash, N., Cristea, A.I. and De Bra, P. (2004) ‘Authoring of Learning Styles in Adaptive Hypermedia: Problems and Solutions’ *In 13th International World Wide Web Conference (WWW 2004)*. 17-20 May, 2004, New York, US. pp. 114-123

Stern, M. K., and Woolf, B. P. (2000) ‘Adaptive Content in an Online Lecture System’ *In Proceedings of the Adaptive Hypermedia Conference*, pp. 227-238

Susarla, S., Adcock, A., Van Eck, R., Moreno, K., and Graesser, A. C. (2003) ‘Development and evaluation of a lesson authoring tool for AutoTutor’ *In* Aleven, V., Hoppe, U., Kay, J., Mizoguchi, R., Pain, H., Verdejo, F. and Yacef, K. (eds.) *AIED2003 Supplemental Proceedings*. Sydney, Australia: University of Sydney School of Information Technologies, pp. 378–387

The Myers and Briggs Foundation. (no date) *The Myers & Briggs Foundation*. [Online] [Accessed on 7 June 2011] http://www.myersbriggs.org/

Triantafillou, E, Pomportsis, A, Demetriadis, S and Georgiadou, E (2004) ‘The value of adaptivity based on cognitive style: an empirical study’ *British Journal of Educational Technology*, Vol. 35 no. 1, pp 95-106

Tsianos, N., Lekkas, Z., Germanakos, P., Mourlas, C. and Samaras, G. (2008) ‘User-centered Profiling on the basis of Cognitive and Emotional Characteristics: An Empirical Study’ *Lecture Notes in Computer Science,* Vol. 5149. Berlin: Springer-Verlag, pp. 214-223

Turing, A.M. (1950) ‘Computing Machinery and Intelligence’ *Mind, New Series*, Vol. 59 no. 236, pp. 433-460

VanLehn, K., Jordan, P., Rose, C.P., Bhembe, D., Bottner, M., Gaydos, A., Makatchev, M., Pappuswamy, U., Ringenberg, M., Roque, A., Siler, S. and Srivastava, R. (2002) ‘The architecture of Why2-Atlas: A coach for qualitative physics essay writing’ *In* Cerri, S.A., Gouarderes, G. and Paraguacu, P. (eds.), *6th International Conf. on Intelligent Tutoring Systems.* Berlin: Springer-Verlag, pp. 158-167

Villaverde, J.E., Godoy, D. and Amandi, A. (2006) ‘Learning styles’ recognition in e-learning environments with feed-forward neural networks’, *Journal of Computer-Assisted Learning*, Vol. 22, pp 197-206

VirtuOz (2007) *VirtuOz, conversational agents.* [Online] [Accessed on 21 August 2007] http://www.virtuoz.com/en/?gclid=CNX-1bDbiY4CFRAFEgodeXC2Ew

Wallace, R.S. (2009) 'The Anatomy of A.L.I.C.E.' *In Parsing the Turing Test Part III*. Berlin: Springer, pp. 181-210

Walonick, D.S. (2010) *Survival Statistics*. Minnesota: StatPac Inc.

Walters D., Egert C. and Cuddihy E. (2000) 'Learning styles and Web-based education: a quantitative approach' *In Proceedings of 9th Annual FACT Conference on Instructional Technology*, 30May-1 June 2000, Buffalo, NY, pp. 115–177

Wang, H. C., Li, T. Y. and Chang, C. Y. (2006) 'A web-based tutoring system with styles-matching strategy for spatial geometric transformation' *Interacting with Computers*, Vol. 18, pp. 331-355

Weber, G. and Brusilovsky, P. (2001) 'ELM-ART: An adaptive versatile system for Web-based instruction' *International Journal of Artificial Intelligence in Education,* Vol. 12 no. 4, pp. 351-384

Weizenbaum, J. (1966) 'ELIZA – A Computer Program for the Study of Natural Language Communication Between Man and Machine' *Communications of the ACM*, vol.9 no. 1, pp. 36-45

Wenger, E. (1987) *Artificial Intelligence and Tutoring Systems*. Los Altos, CA: Morgan Kaufmann.

Witkin, H.A. (1962) *Psychological differentiation: studies of development.* New York : Wiley.

Wolf, C. (2002) 'iWeaver: Towards an Interactive Web-Based Adaptive Learning Environment to Address Individual Learning Styles' *European Journal of Open, Distance and E-Learning 2002* [Online] [Accessed on 5 April 2011] Available from: www.eurodl.org/?p=archives&year=2002&article=61

Wolf, C. (2003) 'iWeaver: Towards 'Learning Style'-based E-learning in Computer Science Education' *In* Greening, T. and Lister, R. (eds.) *Conferences in Research and Practice in Information Technology.* Vol. 20. Adelaide, Australia: Australian Computer Society, pp. 273-279

Woo Woo, C., Evens, M. W., Freedman, R., Glass, M., Seop Shim, L., Zhang, Y., Zhou, Y. and Michael, J. (2006) 'An intelligent tutoring system that generates a natural language dialogue using dynamic multi-level planning' *Artificial Intelligence in Medicine*, Vol. 38, pp. 25-46

Woolf, B. P., Beck, J., Eliot, C. and Stern, M. (2001) 'Growth and maturity of intelligent tutoring systems: A status report' *In* Forbus, K.D. and Feltovich, P.J. (eds.) *Smart machines in education*. Cambridge, Massachusetts: MIT Press, pp. 99-144

Yannibelli, V., Godoy, D. and Amandi, A. (2006) 'A genetic algorithm approach to recognise students' learning styles' *Interactive Learning Environments*, Vol. 14 no. 1, pp 55-78

Appendix 1 The Index of Learning Styles

 For information about appropriate and inappropriate uses of the Index of Learning Styles and a study of its reliability and validity, see <http://www.ncsu.edu/felder-public/ILSpage.html>.

DIRECTIONS

Enter your answers to every question on the ILS scoring sheet. Please choose only one answer for each question. If both “a” and “b” seem to apply to you, choose the one that applies more frequently.

1. I understand something better after I
- **a)** try it out.
- **b)** think it through.

2. I would rather be considered
- **a)** realistic.
- **b)** innovative.

3. When I think about what I did yesterday, I am most likely to get
- **a)** a picture.
- **b)** words.

4. I tend to
- **a)** understand details of a subject but may be fuzzy about its overall structure.
- **b)** understand the overall structure but may be fuzzy about details.

5. When I am learning something new, it helps me to
- **a)** talk about it.
- **b)** think about it.

6. If I were a teacher, I would rather teach a course
- **a)** that deals with facts and real life situations.
- **b)** that deals with ideas and theories.

7. I prefer to get new information in
- **a)** pictures, diagrams, graphs, or maps.
- **b)** written directions or verbal information.

8. Once I understand
- **a)** all the parts, I understand the whole thing.
- **b)** the whole thing, I see how the parts fit.

9. In a study group working on difficult material, I am more likely to
- **a)** jump in and contribute ideas.
- **b)** sit back and listen.

10. I find it easier
- **a)** to learn facts.
- **b)** to learn concepts.

11. In a book with lots of pictures and charts, I am likely to
- **a)** look over the pictures and charts carefully.
- **b)** focus on the written text.

12. When I solve math problems
- **a)** I usually work my way to the solutions one step at a time.
- **b)** I often just see the solutions but then have to struggle to figure out the steps to get to them.

13. In classes I have taken
  a) I have usually gotten to know many of the students.
  b) I have rarely gotten to know many of the students.

14. In reading nonfiction, I prefer
  a) something that teaches me new facts or tells me how to do something.
  b) something that gives me new ideas to think about.

15. I like teachers
  a) who put a lot of diagrams on the board.
  b) who spend a lot of time explaining.

16. When I'm analyzing a story or a novel
  a) I think of the incidents and try to put them together to figure out the themes.
  b) I just know what the themes are when I finish reading and then I have to go back and find the incidents that demonstrate them.

17. When I start a homework problem, I am more likely to
  a) start working on the solution immediately.
  b) try to fully understand the problem first.

18. I prefer the idea of
  a) certainty.
  b) theory.

19. I remember best
  a) what I see.
  b) what I hear.

20. It is more important to me that an instructor
  a) lay out the material in clear sequential steps.
  b) give me an overall picture and relate the material to other subjects.

21. I prefer to study
  a) in a study group.
  b) alone.

22. I am more likely to be considered
  a) careful about the details of my work.
  b) creative about how to do my work.

23. When I get directions to a new place, I prefer
  a) a map.
  b) written instructions.

24. I learn
  a) at a fairly regular pace. If I study hard, I'll "get it."
  b) in fits and starts. I'll be totally confused and then suddenly it all "clicks."

25. I would rather first
  a) try things out.
  b) think about how I'm going to do it.

26. When I am reading for enjoyment, I like writers to
  a) clearly say what they mean.
  b) say things in creative, interesting ways.

27. When I see a diagram or sketch in class, I am most likely to remember
  a) the picture.
  b) what the instructor said about it.

28. When considering a body of information, I am more likely to
a) focus on details and miss the big picture.
b) try to understand the big picture before getting into the details.

29. I more easily remember
a) something I have done.
b) something I have thought a lot about.

30. When I have to perform a task, I prefer to
a) master one way of doing it.
b) come up with new ways of doing it.

31. When someone is showing me data, I prefer
a) charts or graphs.
b) text summarizing the results.

32. When writing a paper, I am more likely to
a) work on (think about or write) the beginning of the paper and progress forward.
b) work on (think about or write) different parts of the paper and then order them.

33. When I have to work on a group project, I first want to
a) have "group brainstorming" where everyone contributes ideas.
b) brainstorm individually and then come together as a group to compare ideas.

34. I consider it higher praise to call someone
a) sensible.
b) imaginative.

35. When I meet people at a party, I am more likely to remember
a) what they looked like.
b) what they said about themselves.

36. When I am learning a new subject, I prefer to
a) stay focused on that subject, learning as much about it as I can.
b) try to make connections between that subject and related subjects.

37. I am more likely to be considered
a) outgoing.
b) reserved.

38. I prefer courses that emphasize
a) concrete material (facts, data).
b) abstract material (concepts, theories).

39. For entertainment, I would rather
a) watch television.
b) read a book.

40. Some teachers start their lectures with an outline of what they will cover. Such outlines are
a) somewhat helpful to me.
b) very helpful to me.

41. The idea of doing homework in groups, with one grade for the entire group,
a) appeals to me.
b) does not appeal to me.

42. When I am doing long calculations,
a) I tend to repeat all my steps and check my work carefully.
b) I find checking my work tiresome and have to force myself to do it.

43. I tend to picture places I have been
- **a)** easily and fairly accurately.
- **b)** with difficulty and without much detail.

44. When solving problems in a group, I would be more likely to
- **a)** think of the steps in the solution process.
- **b)** think of possible consequences or applications of the solution in a wide range of areas.

Appendix 2 Logic Rules for the Felder-Silverman Learning Styles Model

This Appendix lists the logic rules which govern the increments to learning styles values, applied to the Felder-Silverman (1988) (FS) learning styles model. A full description of the FS model is given in Chapter 2. The 29 logic rules listed here were applied during the experiments described in Chapter 8.

```
IF (answer IS (wrong OR don't-know) AND give-theory-explanation)
THEN
{
  IF (next-answer=right)
  THEN
    (INCREMENT INTUITOR) AND (INCREMENT REFLECTIVE) AND (INCREMENT VERBAL);
}
```

```
IF (answer IS (wrong OR don't-know) AND show-movie)
THEN
{
  IF (next-answer=right)
  THEN
    (INCREMENT VISUAL) AND (INCREMENT ACTIVE);
}
```

```
IF (answer IS (wrong OR don't-know) AND show-image)
THEN
{
  IF (next-answer=right)
  THEN
    INCREMENT VISUAL;
}
```

```
IF (answer IS (wrong OR don't-know) AND show-example)
THEN
{
  IF (next-answer=right)
  THEN
    INCREMENT ACTIVE;
}
```

```
IF (partial-answer-given)
THEN
  (INCREMENT INTUITOR) AND (INCREMENT SEQUENTIAL);
```

```
IF (mostly-correct AND small-mistakes)
THEN
  INCREMENT INTUITOR;
```

```
IF (complex-answer-required AND (mistakes-made > 1))
THEN
  INCREMENT INTUITOR;
```

```
IF (complex-answer-required AND first-answer IS correct)
THEN
  (INCREMENT SENSOR) AND (INCREMENT GLOBAL);
```

```
IF (question-template-applied IS choice-of-approach)
THEN
{
  IF (student-chooses-onego OR student-attempts-query)
  THEN
    (INCREMENT GLOBAL) AND (INCREMENT ACTIVE);
}
```

```
IF (question-template-applied IS choice-of-approach)
THEN
{
  IF (student-chooses-steps OR student-chooses-don't-know)
  THEN
    (INCREMENT SEQUENTIAL) AND (INCREMENT REFLECTIVE);
}
```

```
IF (question-template-applied IS choice-of-approach)
THEN
{
  IF ((student-chooses-onego OR student-attempts-query) AND (wrong-answers > 1 OR
      don't-know-answers > 1))
  THEN
    (INCREMENT VERBAL) AND (INCREMENT SEQUENTIAL);
}
```

```
IF (question-style-applied IS trick-question)
THEN
{
  IF (answer-correct-first-time)
  THEN
    (INCREMENT SENSOR) AND (INCREMENT VERBAL);
  ELSE
    (INCREMENT INTUITOR) AND (INCREMENT VISUAL);
}
```

```
IF (tutorial-question IS right)
THEN
{
  IF (related-tutoring-style IS practical)
  THEN
    (INCREMENT ACTIVE) AND (INCREMENT SENSOR);
  ELSE
  {
    IF (related-tutoring-style IS theoretical)
    THEN
      (INCREMENT REFLECTIVE) AND (INCREMENT INTUITOR);
  }
}
```

```
IF ((pre-test-question IS wrong) AND (post-test-question IS right))
THEN
{
  IF (related-tutoring-style IS practical)
  THEN
    (INCREMENT ACTIVE) AND (INCREMENT SENSOR);
  ELSE
  {
    IF (related-tutoring-style IS theoretical)
    THEN
      (INCREMENT REFLECTIVE) AND (INCREMENT INTUITOR);
  }
}
```

```
IF (total-student-wordcount >= average-student-wordcount)
THEN
  INCREMENT VERBAL;
ELSE
  INCREMENT VISUAL;
```

```
IF (total-student-wordcount-per-interaction >= average-student-wordcount-per-
interaction)
THEN
  INCREMENT VERBAL;
ELSE
  INCREMENT VISUAL;
```

```
IF (number-of-interactions >= average-number-of-interactions)
THEN
  INCREMENT VERBAL;
ELSE
  INCREMENT VISUAL;
```

```
IF (number-of-FAQs-asked >= average-number-of-FAQs-asked)
THEN
  INCREMENT VERBAL;
ELSE
  INCREMENT VISUAL;
```

```
IF (tutorial-duration >= average-tutorial-duration)
THEN
  INCREMENT SENSOR;
ELSE
  INCREMENT INTUITOR;
```

```
IF (duration-per-interaction >= average-duration-per-interaction)
THEN
  INCREMENT SENSOR;
ELSE
  INCREMENT INTUITOR;
```

```
IF (reading-time >= average-reading-time)
THEN
  INCREMENT SENSOR AND (INCREASE VISUAL);
ELSE
  (INCREMENT INTUITOR) AND (INCREMENT VERBAL);
```

IF (student-word IN [sensor-words-list]) THEN   INCREMENT SENSOR;
IF (student-word IN [intuitive-words-list]) THEN   INCREMENT INTUITOR;
IF (student-word IN [visual-words-list]) THEN   INCREMENT VISUAL;
IF (student-word IN [verbal-words-list]) THEN   INCREMENT VERBAL;
IF (student-word IN [active-words-list]) THEN   INCREMENT ACTIVE;
IF (student-word IN [reflective-words-list]) THEN   INCREMENT REFLECTIVE;
IF (student-word IN [sequential-words-list]) THEN   INCREMENT SEQUENTIAL;
IF (student-word IN [global-words-list]) THEN   INCREMENT GLOBAL;

Appendix 3 Tutorial Conversation Blueprint Excerpt

This appendix shows an excerpt from the *Tutorial Conversation Blueprint* document produced during the development of a conversational SQL tutorial when implementing Oscar PCITS, as described in Chapter 7. The excerpt is from a working document, rather than the finalised version of the conversation script, and it can be seen that some decisions on the tutorial conversation have been left open until the CA is scripted. The excerpt demonstrates the complexity of capturing and documenting a tutoring conversation, even for a simple tutorial question such as that shown.

Oscar Tutor: Do you remember that SQL functions can be classified into two broad categories: data definition language (DDL) commands and data manipulation language (DML) commands? DDL commands facilitate the creation of tables, indexes, and views as well as commands to define access rights to those database objects. Can you type in one DDL command?

Learner answers could be yes / no / don't know or the actual command itself.

<u>Route 1</u>

Learner Answer: Yes

Oscar Tutor: Tell me then:

Learner Answer: {CREATE, ALTER, DROP,VIEW, etc……} – see appendix A for full list of DDL commands.

Oscar Tutor: That's right, well done. Let's move on to the next question. and proceed to Question 2.

NB – Go on to Q2

<u>Route 2</u>

Learner Answer: {CREATE, ALTER, DROP,VIEW, etc……} – see appendix A for full list of DDL commands.

<u>Route 3</u>

Learner Answer: No/ don't know/ what do you mean/ what's a DDL

NB- Learner may have forgotten what DDL etc is and therefore may have to jump to the FAQ layer to do a FAQ on key terms.- OR better present as a theory/concept see below

NB- Question arises if the student does not know what the DDL is after being told the definition. Strategy is then to present as a Theory/ Concept and then ask the question again.

Oscar Tutor: DDL, otherwise known as Data Definition Language statements, are used to define the database structure or schema. They are a set of statements which allow us to create new database OBJECTS. So supposing I wanted to add a table to the database, what command would I use ?

Route 3a Learner could now say CREATE or CREATE TABLE which is correct and the tutor could proceed to question 2 after proving feedback in the form of "well done"

If they get right answer after DDL definition, increase INTUITIVE

Route 3b: Learner gets it wrong – show them an example –create table movie1.

Oscar Tutor: Watch this short movie clip and when it is finished type in the command you think was used to add a table to the database.

NB – if student gets it right – say "well done" and proceed to Question 2. **– Go on to Q2**

If they get right answer after movie, increase VISUAL

If the student gets it wrong:

Display the image Characteristics_of_sql.jpg on a new screen

Oscar Tutor: This picture displays the three types of SQL Commands: Data Definition, Data Manipulation and Data Control, along with examples of the types of SQL statements that could be used. Do you now remember that the DDL is used to create objects such as new tables in the database?

<u>Route 4a</u>

Learner Answer: Yes

Oscar Tutor: Good, Lets now proceed with the next question, should be easier as I have given you a hint !

NB:– Go on to Q2

<u>Route 4b</u>

Learner Answer: No / Can't remember / not sure

Oscar Tutor: You really need to remember the basics before starting the PL/SQL course. It might be worth going to revise some of your previous notes. Or if you would like to access the full guide to the SQL language visit **http://download.oracle.com/docs/cd/B28359_01/server.111/b28286/toc.htm**

Oscar Tutor: Would you like to continue to see what else you can remember ?

Learner Answer: Yes

NB – Go on to Q2

Learner Answer: No

Oscar Tutor: OK we can save your progress – why not go away and revise the SQL basics and come back and see how you do. Click "end session" if you want to log out and come back later.

End Session

Appendix 4 SQL MCQ Test

This appendix shows the Multiple Choice Question (MCQ) test designed to test learners' knowledge of the SQL covered by the Oscar CITS tutorial. The MCQ test is completed both before (known as the *pre-test*) and after (known as the *post-test*) the Oscar CITS tutorial, and is used to assess learning gain.

1. CREATE is a ____

- DCL command.
- Transaction control command.
- DDL command.
- DML command.

2. To list the contents of a table, you use the DISPLAY command.

- True
- False

3. Which query will output the PRODUCT table contents when the value of the character field P_CODE is alphabetically less than 1558-QW1?

- SELECT P_CODE, P_DESCRIPT, P_QOH, P_MIN, P_PRICE FROM PRODUCT WHERE P_CODE = (1558-QW1);
- SELECT P_CODE, P_DESCRIPT, P_QOH, P_MIN, P_PRICE FROM PRODUCT WHERE P_CODE < '1558-QW1';
- SELECT P_CODE, P_DESCRIPT, P_QOH, P_MIN, P_PRICE FROM PRODUCT WHERE P_CODE = [1558-QW1];
- SELECT P_CODE, P_DESCRIPT, P_QOH, P_MIN, P_PRICE FROM PRODUCT WHERE P_CODE = {1558-QW1};

4. Which SQL query will list all the rows in which the inventory stock dates occur on or after January 20, 2006?

- SELECT P_DESCRIPT, P_QOH, P_MIN, P_PRICE, P_INDATE FROM PRODUCT WHERE P_INDATE >= '20/01/2006';
- SELECT P_DESCRIPT, P_QOH, P_MIN, P_PRICE, P_INDATE FROM PRODUCT WHERE P_INDATE >= #01/20/2006#;
- SELECT P_DESCRIPT, P_QOH, P_MIN, P_PRICE, P_INDATE FROM PRODUCT WHERE P_INDATE >= '20-JAN-2006';
- SELECT P_DESCRIPT, P_QOH, P_MIN, P_PRICE, P_INDATE FROM PRODUCT WHERE P_INDATE >={01-20-2006};

5. The basic SQL aggregate function that gives the arithmetic mean for the specified column is ____

- MEAN.
- AVG.
- MAX.
- SUM.

6. What is the SQL syntax requirement to list the table contents for either V_CODE = 21344 or V_CODE = 24288?

- SELECT P_DESCRIPT, P_INDATE, P_PRICE, V_CODE FROM PRODUCT WHERE V_CODE = 21344 OR V_CODE <= 24288;
- SELECT P_DESCRIPT, P_INDATE, P_PRICE, V_CODE FROM PRODUCT WHERE V_CODE = 21344 AND V_CODE = 24288;
- SELECT P_DESCRIPT, P_INDATE, P_PRICE, V_CODE FROM PRODUCT WHERE V_CODE = 21344 AND V_CODE > 24288;
- SELECT P_DESCRIPT, P_INDATE, P_PRICE, V_CODE FROM PRODUCT WHERE V_CODE = 21344 OR V_CODE = 24288;

7. The basic SQL aggregate function that gives the number of rows containing not null values for the given column is ____

- COUNT.
- MIN.
- MAX.
- SUM.

8. Which command is used to list all different values of V_CODE from the PRODUCT table with no duplication?

- SELECT ONLY V_CODE FROM PRODUCT;
- SELECT UNIQUE V_CODE FROM PRODUCT;
- SELECT DIFFERENT V_CODE FROM PRODUCT;
- SELECT DISTINCT V_CODE FROM PRODUCT;

9. What is the SQL query to display the P_DESCRIPT and P_PRICE fields from the PRODUCT table and the V_NAME, V_CONTACT, V_AREACODE and V_PHONE fields from the VENDOR table where the PRODUCT and VENDOR tables are joined by V_CODE and the output is in price order?

- SELECT P_DESCRIPT, P_PRICE, V_NAME, V_CONTACT, V_AREACODE, V_PHONE FROM PRODUCT, VENDOR WHERE PRODUCT.V_CODE <> VENDOR.V_CODE SORT BY P_PRICE;

- SELECT P_DESCRIPT, P_PRICE, V_NAME, V_CONTACT, V_AREACODE, V_PHONE FROM PRODUCT, VENDOR WHERE PRODUCT.V_CODE => VENDOR.V_CODE ORDER BY P_PRICE;
- SELECT P_DESCRIPT, P_PRICE, V_NAME, V_CONTACT, V_AREACODE, V_PHONE FROM PRODUCT, VENDOR WHERE PRODUCT.V_CODE == VENDOR.V_CODE SORT BY P_PRICE;
- SELECT P_DESCRIPT, P_PRICE, V_NAME, V_CONTACT, V_AREACODE, V_PHONE FROM PRODUCT, VENDOR WHERE PRODUCT.V_CODE = VENDOR.V_CODE ORDER BY P_PRICE;

10. The special operator used to define a range limit in the WHERE clause is ____

- BETWEEN.
- NULL.
- LIKE.
- IN.

11. Which of the following queries would return a list of all COURSE_ID adjoined to COURSE_TITLE?

- SELECT COURSE_ID ++ COURSE_TITLE FROM COURSE;
- SELECT JOIN(COURSE_ID, COURSE_TITLE) FROM COURSE;
- SELECT CONCAT(COURSE_ID, COURSE_TITLE) FROM COURSE;
- SELECT (COURSE_ID && COURSE_TITLE) FROM COURSE;

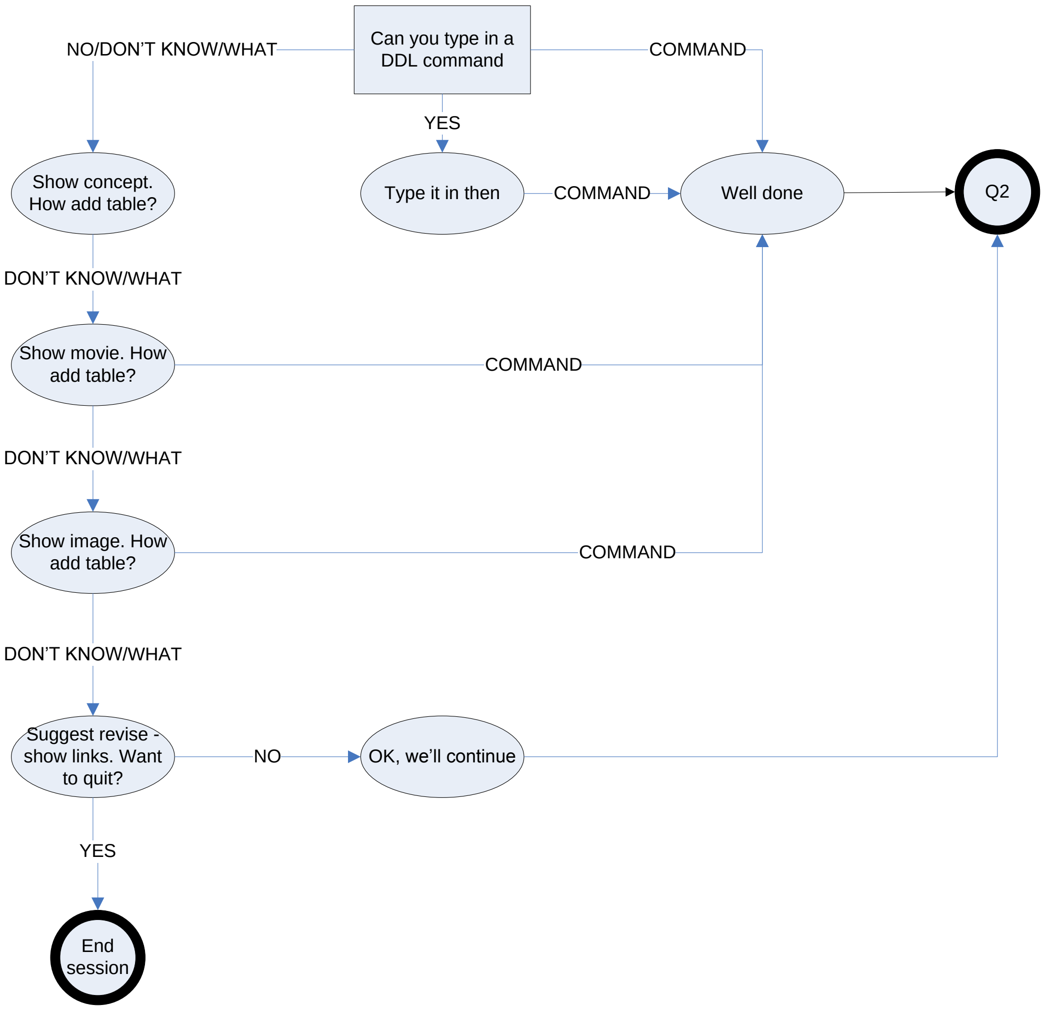
12. Which of the following keywords is always required when using an aggregate function?

- GROUP WITH
- GROUP USING
- COLLATE
- GROUP BY

Appendix 5 Examples of Applying Generic Question Templates

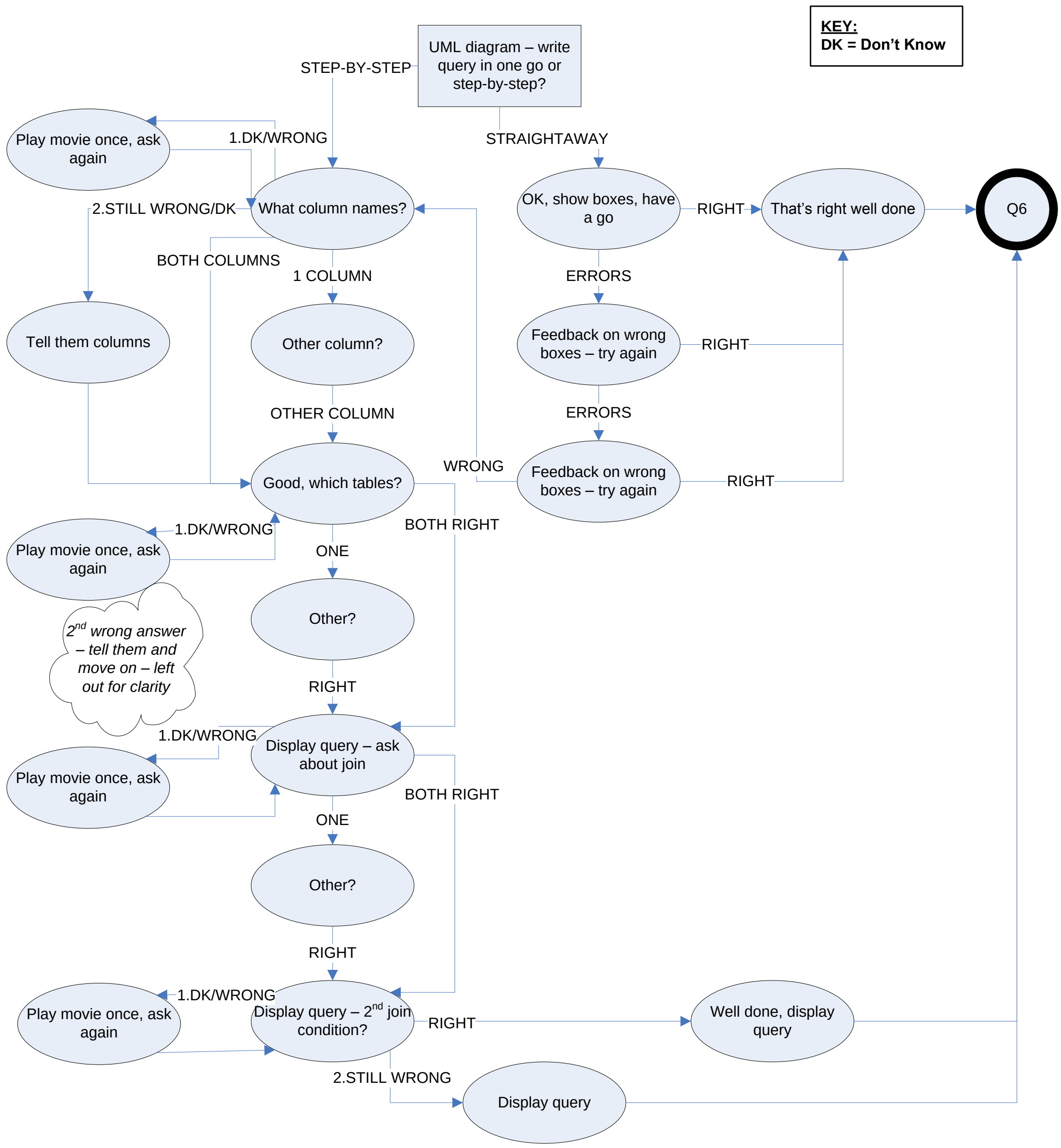
This appendix includes two diagrams showing the application of the generic question templates (described in Chapter 6) to the SQL tutorial questions captured for the implementation of Oscar PCITS described in Chapter 7.

Example 1 – Applying the 'with Hints' question template.



The diagram above shows the 'Generic Question Template with Hints' (Figure 6.2) applied to tutorial question 1 (see Table 7.2), which is a question about SQL Data Definition Language (DDL) commands.

Example 2 – Applying the 'Choice of Approach' question template.



The diagram above shows the application of the 'Generic Question Template with Choice of Approach' (Figure 6.3) to tutorial question 5 (see Table 7.2). Tutorial question 5 asks the learner to write an SQL query to solve a problem which requires two database tables to be joined.

Appendix 6 MCQ Test Mapped to Tutorial Questions and Styles

This appendix shows the mapping of the Multiple Choice Question (MCQ) test questions to tutorial questions and their theoretical or practical style. This information is used in Experiment 4, described in Chapter 8. The MCQ test (see Appendix 4) is completed both before (known as the *pre-test*) and after (known as the *post-test*) the Oscar CITS tutorial.

MCQ Test Question	Tutorial Question	Question Style
Q1	Question 1 – DDL	Theoretical
Q2	Question 2 – DML	Theoretical
Q3	Question 3 – SELECT *	Practical
Q4	Question 9 – Query with range	Practical
Q5	Question 6 – Functions	Theoretical
Q6	Question 5 – Query with join	Practical
Q7	Question 6 – Functions	Theoretical
Q8	Question 10 – DISTINCT	Theoretical
Q9	Question 5 – Query with join	Practical
Q10	Question 9 – Query with range	Practical
Q11	Question 7 – Query with functions	Practical
Q12	Question 8 – GROUP BY	Theoretical

Appendix 7 Related Refereed Publications

This appendix includes details of refereed publications which relate to the research presented in this thesis. The publications are listed below and a copy of each paper is attached.

Latham, A.M., Crockett, K.A., McLean, D.A. and Edmonds, B. (2012) 'A conversational intelligent tutoring system to automatically predict learning styles' *In press: Computers & Education Journal* special issue, Elsevier (accepted July 2011)

Latham, A.M., Crockett, K.A., McLean, D.A. and Edmonds, B. (2011) 'Oscar: An Intelligent Adaptive Conversational Agent Tutoring System' *Lecture Notes in Artificial Intelligence*, Vol. 6682. Berlin: Springer-Verlag, pp. 563-572

Latham, A.M., Crockett, K.A., McLean, D.A. and Edmonds, B. (2011) 'Oscar: A Personalised Online Conversational Intelligent Tutoring System' *In The CAL Conference 2011, Learning Futures: Education, Technology, Sustainability.* Manchester, 13-15 April 2011. Elsevier.

Latham, A.M., Crockett, K.A., McLean, D.A. and Edmonds, B. (2010) 'Predicting Learning Styles in a Conversational Intelligent Tutoring System' *Lecture Notes in Computer Science*, Vol. 6483. Berlin: Springer-Verlag, pp. 131-140

Latham, A.M., Crockett, K.A., McLean, D.A., Edmonds, B. and O'Shea, K. (2010) 'Oscar: An Intelligent Conversational Agent Tutor to Estimate Learning Styles' *In Proc. IEEE WCCI 2010.* Barcelona, 18-23 July 2010. IEEE, pp. 2533-2540.

A conversational intelligent tutoring system to automatically predict learning styles

Annabel Latham[a], Keeley Crockett[a], David McLean[a], Bruce Edmonds[b]
[a]*The Intelligent Systems Group, School of Computing, Mathematics and Digital Technology*
[b]*The Centre for Policy Modelling*
The Manchester Metropolitan University, Chester Street, Manchester M1 5GD, UK
Email: a.latham@mmu.ac.uk, k.crockett@mmu.ac.uk, d.mclean@mmu.ac.uk, b.edmonds@mmu.ac.uk

Abstract

This paper proposes a generic methodology and architecture for developing a novel conversational intelligent tutoring system (CITS) called Oscar that leads a tutoring conversation and dynamically predicts and adapts to a student's learning style. Oscar aims to mimic a human tutor by implicitly modelling the learning style during tutoring, and personalising the tutorial to boost confidence and improve the effectiveness of the learning experience. Learners can intuitively explore and discuss topics in natural language, helping to establish a deeper understanding of the topic. The Oscar CITS methodology and architecture are independent of the learning styles model and tutoring subject domain. Oscar CITS was implemented using the Index of Learning Styles (ILS) model (Felder & Silverman 1988) to deliver an SQL tutorial. Empirical studies involving real students have validated the prediction of learning styles in a real-world teaching/learning environment. The results showed that all learning styles in the ILS model were successfully predicted from a natural language tutoring conversation, with an accuracy of 61-100%. Participants also found Oscar's tutoring helpful and achieved an average learning gain of 13%.

Keywords:

Architectures for educational technology system
Human-computer interface
Intelligent tutoring systems
Interactive learning environments
Teaching/learning strategies

1. Introduction

The widespread use of computers and access to the Internet has created many opportunities for online education, such as improving distance-learning and classroom support. Intelligent Tutoring Systems (ITS) extend traditional content-delivery computerised learning systems by adding intelligence to improve the effectiveness of a learner's experience (Brusilovsky & Peylo 2003). This normally involves personalising tutoring using factors such as learner knowledge, emotion or learning style to alter the sequence and style of learning material. Most ITS are hyperlink menu based (Cha, Kim, Park, Yoon, Jung & Lee 2006; Klasnja-Milicevic, Vesin, Ivanovic & Budimac 2011; Popescu 2010; Wang, Wang & Huang 2008) and adapt the tutoring by reordering menu items (Garcia, Amandi, Schiaffino & Campo 2007), allowing learners to manage their own study at a time and place to suit them. This experience has more in common with computerised textbooks than classroom tutorials, where human tutors direct the learning. An extension of ITS is Conversational Intelligent Tutoring Systems (CITS), which integrate natural language interfaces rather than menus, allowing learners to explore topics through discussion and to construct knowledge as they would in the classroom. However, it is a complex and time consuming task to develop a CITS which can converse naturally. Consequently only a few CITS exist at present (D'Mello, Lehman, Sullins, Daigle, Combs, Vogt *et al* 2010; Arnott, Hastings & Allbritton 2008; Sarrafzadeh, Alexander, Dadgostar, Fan & Bigdeli 2008).

A CITS that can imitate a human tutor by leading an adaptive tutorial conversation uses a familiar format which can help improve learner confidence and motivation, leading to a better learning experience. Human tutors adapt their tutoring style and content based on cues they pick up from learners, such as their level of understanding and learning style. Learning styles model the way groups of people prefer to learn (Felder & Silverman 1988; Hsieh, Jang & Hwang 2011), for example by active experimentation or by observation. Some ITS adapt tutoring to an individual's learning style, either determined using a formal questionnaire (Papanikolaou, Grigoriadou, Kornilakis & Magoulas 2003) or by analysing learner behaviour (Kelly & Tangney 2006). However, there are no tutor-led CITS that can predict and adapt to learning style during the tutoring session like a human tutor.

This paper describes the architecture and methodology for creating a novel CITS called Oscar that dynamically predicts and adapts to an individual's preferred learning style during a tutorial conversation. The aim of the research was to imitate a human tutor by using knowledge of learning styles and learner behaviour to predict learning style rather than an interface specifically designed to capture learning styles, as in (Cha et al 2006). Whilst this considerably increases the complexity of predicting learning styles, conversational interfaces are intuitive to use and the discussion of problems can prompt a deeper understanding of topics. This paper also describes a series of experiments that aim to determine if it is possible to predict learning style from a learner's behaviour during a tutorial conversation, and thus validate the proposed methodology and architecture.

In this paper, section 2 introduces some background and related work of intelligent tutoring systems, conversational agents and the Index of Learning Styles (Felder & Silverman 1988). Section 3 introduces the Oscar CITS, and Sections 4 and 5 describe a generic methodology and architecture for creating an Oscar CITS. Section 6 describes the implementation of Oscar CITS and the real-world experiments conducted to investigate the prediction of learning styles from a natural language tutoring dialogue. Section 7 presents the experimental results and discussion and Section 8 outlines the conclusions and future work.

2. Related work

2.1. Intelligent tutoring systems

Computerised learning systems were traditionally information-delivery systems developed by converting tutor or distance-learning material into a computerised format (Brooks, Greer, Melis & Ullrich 2006). The popularity of the Internet has enhanced the opportunities for e-learning, however most online systems are still teacher-centred and take little account of individual learner needs (Spallek 2003). Within the field of computerised learning systems, adaptive educational systems attempt to meet the needs of different students by offering individualised learning (Brusilovsky & Peylo 2003). Intelligent Tutoring Systems (ITS) are adaptive systems which use intelligent technologies to personalise learning according to individual student characteristics, such as knowledge of the subject, mood and emotion (D'Mello et al. 2010) and learning style (Yannibelli, Godoy & Amandi 2006).

There are three main approaches to intelligent tutoring (Brusilovsky & Peylo 2003):

- *Curriculum sequencing* introduces adaptation by presenting students with learning material in a sequence and style best suited to their needs (Klasnja-Milicevic *et al* 2011).
- *Intelligent solution analysis* adds intelligence to ITS by giving students detailed feedback on incomplete or erroneous solutions, helping them learn from their mistakes (Mitrovic 2003).
- *Problem solving support* techniques offer learners intelligent assistance to reach a solution (Melis, Andres, Budenbender, Frishauf, Goguadse, Libbrecht *et al* 2001).

Curriculum sequencing is the most widely used technique (Brusilovsky and Peylo 2003). Traditionally ITS adapt to existing student knowledge but more recently learner affect factors have been incorporated, such as emotion (Ammar, Neji, Alimi & Gouarderes 2010), personality (Leontidis & Halatsis 2009) and learning style (Popescu 2010). Few ITS incorporate all three techniques as they are complex and time-consuming to develop, but the Oscar CITS presented in this paper will incorporate all three intelligent technologies by personalising learning material and discussing problems and solutions with students.

ITS are normally menu or hyperlink based, with choices re-ordered or ranked to recommend a particular sequence to learners (Klasnja-Milicevic et al 2011; Garcia et al 2007). Whilst this design simplifies the capture of learner behaviour and choices, learners are really being assisted in self-learning rather than tutored, which is little different from recommending chapters of a book. CITS address this issue by employing natural language interfaces whose intuitive, dialogue-based tutoring is more comparable to classroom tutorials (Chi, Siler, Jeong, Yamauchi & Hausmann 2001; D'Mello et al 2010; Sarrafzadeh et al 2008). However, despite their more instinctive, teacher-led learning experience (which supports the construction of knowledge adopted by human tutors), it is difficult to automate natural conversations and so CITS are uncommon (D'Mello et al 2010; Woo Woo, Evens, Freedman, Glass, Seop Shim, Zhang et al 2006; Sarrafzadeh et al 2008).

ITS that adapt to learning styles capture them using a formal questionnaire (Papanikolaou et al 2003) or by analysing learner behaviour (Cha et al 2006; Garcia et al 2007). Whilst questionnaires are the simplest measure of learning styles, learners find them onerous and may not lend enough time or attention to complete them accurately (Yannibelli, Godoy & Amandi 2006). Implicitly modelling learning styles by analysing learner behaviour history (Garcia et al 2007) removes the requirement for a questionnaire, but delays adaptation until several modules have been completed. Also, this method does not incorporate changes in learning style which may occur over time or for different topics. EDUCE (Kelly & Tangney 2006) and WELSA (Popescu 2010) both estimate learning style dynamically for curriculum sequencing, but do not include a conversational interface or other intelligent tutoring technologies. The Oscar CITS will dynamically predict learning style throughout the tutoring conversation and adapt its intelligent tutoring style to suit the learning style predicted.

2.2. Conversational agents

Conversational agents (CAs) are computer programs which allow people to communicate with computer systems using natural language (O'Shea, Bandar & Crockett 2011). CA interfaces are intuitive to use, and have been used effectively in many applications, such as web-based guidance (Latham, Crockett & Bandar 2010), database interfaces (Pudner, Crockett & Bandar 2007) and tutoring (D'Mello et al 2010). CAs can add natural dialogue to ITS, but are used infrequently as they are complex and time-consuming to develop, requiring expertise in the scripting of dialogues (O'Shea, Bandar & Crockett 2011). ITS which aim to mimic a human tutor (such as Oscar CITS) need CA interfaces to support the construction of knowledge through discussion (Chi et al 2001).

Textual CAs usually adopt a pattern matching (Michie 2001) or semantic based (Li, Bandar, McLean & O'Shea 2004; Khoury, Karray & Kamel 2008) approach. Semantic-based CAs seek to understand the meaning of the natural language whereas pattern-matching CAs use an algorithm to match key words and phrases from the input to a set of pattern-based rules (Pudner, Crockett & Bandar 2007). As pattern matching CAs match key words within an utterance, they do not require grammatically correct or complete input. However, there are usually numerous patterns in a given context (Sammut 2001), leading to many hundreds of rules in the CA's knowledge base, which demonstrates the complexity and time required to script rules for a pattern-matching CA. A pattern matching CA was adopted for Oscar CITS as it must cope with grammatically incomplete or incorrect utterances that are commonly found in text-based chat by students.

2.3. Index of learning styles

The Index of Learning Styles (ILS) model (Felder & Silverman 1988) describes the teaching and learning styles in engineering education. The ILS model represents an individual's learning style as points along four dimensions that indicate both the strength and the nature of their learning style preference. Each learning style dimension relates to a step in the process of receiving and processing of information, as illustrated in Fig. 1. The ILS is assessed using a 44-question forced-choice questionnaire (11 questions per learning style dimension), that assigns a style and score for each dimension.

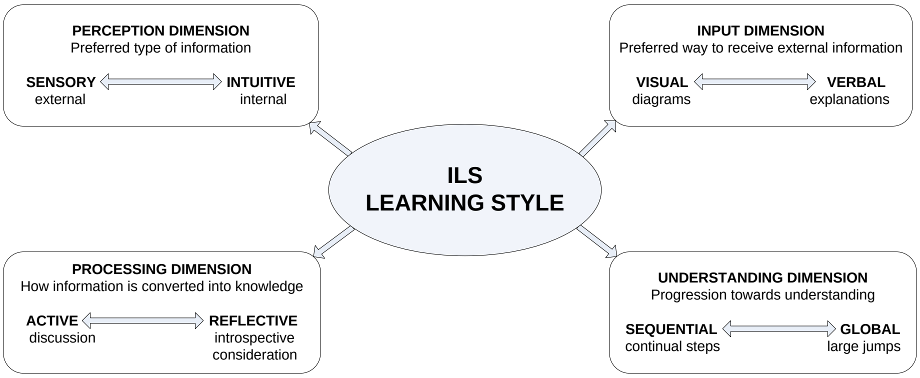


Fig. 1. ILS dimensions.

In addition to the formal assessment questionnaire, the ILS model describes typical learner behaviours that can be used to informally group types of learners. The ILS model was adopted when implementing the Oscar CITS as engineering students make up the initial experimental groups. However, the Oscar CITS is generic and its flexible modular structure does not restrict the choice of learning styles model to the ILS.

3. Oscar CITS

The Oscar CITS is a novel conversational intelligent tutoring system which dynamically predicts a student's learning style during a tutoring conversation, and adapts its tutoring style appropriately. Oscar's pedagogical aim is to provide the learner with the most appropriate learning material for their learning style to promote a more effective learning experience and a deeper understanding of the topic. Rather than being designed with the purpose of picking up learning styles (such as Cha et al 2006) the Oscar CITS aims to imitate a human tutor by leading a two-way discussion and using cues from the student's dialogue and behaviour to predict and adapt to their learning style. Oscar CITS incorporates intelligent technologies to sequence the curriculum according to learner knowledge and learning style, intelligently analyse solutions and give hints to assist learners in constructing knowledge. Oscar's natural language interface and classroom tutorial style are modelled on classroom tutorials (Crown copyright 2004), enabling learners to draw on their experience of face-to-face tutoring to feel more comfortable and confident in using the CITS. Oscar CITS is an online personal tutor which can answer questions, provide hints and assistance using natural dialogue, and which favours learning material to suit each individual's learning style. The Oscar CITS offers 24-hour personalised learning support at a fixed cost.

General descriptions of Oscar CITS, including its implementation, example learner dialogue and the results of initial studies in predicting learning styles, have been reported in Latham, Crockett, McLean, Edmonds & O'Shea (2010) and Latham, Crockett, McLean & Edmonds (2010). The Oscar CITS adaptation strategies were described in Latham, Crockett, McLean & Edmonds (2011), which reported empirical results showing that students whose learning material matched their learning styles performed 12% better than those with unmatched material.

The rest of this paper will describe an original methodology and architecture for creating an Oscar CITS and the experiments conducted to investigate its success in predicting learning styles in a real teaching/learning environment.

4. Predicting learning styles through natural language dialogue

CITS are complex and time-consuming to develop, requiring expertise in knowledge engineering (the capture and formatting of expert knowledge (O'Shea, Bandar & Crockett 2011), such as tutoring, learning styles and domain knowledge) and CA scripting. Formalising the development of a CITS which can be applied to different learning styles models and tutoring domains will help to speed up the development. This section presents a methodology for creating an Oscar CITS which can predict learning styles from a natural language dialogue.

4.1. Methodology for creating Oscar CITS

The methodology for creating an Oscar CITS consists of three phases as shown in Table 1. The first phase of the methodology relates to the creation of the learning styles module and the second phase to the tutorial subject domain. The third phase incorporates the learning styles predictor and tutorial conversation into a CITS architecture. Each phase will now be described.

Table 1.
3-Phase methodology for creating Oscar CITS.

Phase 1: Create the Learning Styles Predictor Module

1.1. Select a Learning Styles Model
  a. Reduce the learning styles model if necessary
  b. Extract the behaviour characteristics
1.2. Map learning style behaviour to the conversational tutoring style
1.3. Analyse the learning styles model for language traits
1.4. Adapt the generic logic rules to predict learning styles

Phase 2: Design a Tutorial Conversation

2.1. Capture the tutorial scenario and questions (including movies, voice, images, examples, etc.) from human tutors in a specific domain
2.2. Determine the conversational structure/style
2.3. Map tutorial questions onto the generic question styles and templates
2.4. Script CA natural language dialogue for each tutorial question using the 3-level model
2.5. Link tutorial dialogue to logic rules through CA variables

Phase 3: Construct the CITS Architecture

4.2. Methodology phase 1: create the learning styles predictor module

4.2.1. Step 1.1: select a learning styles model

The first step in creating the learning styles predictor module requires a learning styles model (Felder & Silverman 1988, Honey & Mumford 1992) to be selected. To illustrate and validate Phase 1 of the methodology, the ILS model (Felder & Silverman 1988) was selected as the initial experimental group will be university engineering students. The ILS questionnaire contains 44 questions, which is too many to incorporate into a single tutoring session without being onerous for students. To reduce the ILS model, a study was undertaken to investigate which were the best predictor questions (Latham, Crockett, McLean & Edmonds 2009). The study of 103 completed ILS questionnaires found that 17 questions predicted the overall learning style result in at least 75% of cases, with the top three questions predicting the result in 84% of cases. The resulting subset of the best ILS predictor questions formed the basis of further analysis in developing the Oscar CITS strategy for the prediction of learning styles.

The ILS model describes typical behaviour characteristics for each learning style. For clarity and ease of analysis, the behaviour characteristics were extracted from the ILS model and summarised in a table of common learner behaviour (Table 2).

Table 2.
Typical learner behaviour characteristics extracted from the ILS model.

Sensor	**Intuitor**
Prefer facts, data, experimentation	Prefer principles and theories
Prefer solving problems using standard methods	Prefer innovation
Dislike surprises	Dislike repetition
Patient with detail	Bored by detail
Do not like complications	Welcome complications
Good at memorising facts	Good at grasping new concepts
Careful but slow	Quick but careless
Comfortable with symbols (eg. words)	Uncomfortable with symbols
Visual	**Verbal**
Remember what they see	Remember what they hear, or what they hear then say
Like pictures, diagrams, flow charts, time lines, films	Like discussion
Prefer visual demonstration	Prefer verbal explanation
	Learn by explaining to others
Active	**Reflective**
Do something with information – discuss/explain/test	Examine and manipulate information introspectively
Active experimentation	Reflective observation
Do not learn much in passive situations (lectures)	Do not learn much if no chance to think (lectures)
Work well in groups	Work better alone
Experimentalists	Theoreticians
Process information by setting up an experiment to test an idea, or try out on a colleague	Process information by postulating explanations/interpretations, drawing analogies, formulating models
Sequential	**Global**
Follow linear reasoning processes	Make intuitive leaps
Can work with material they have only partially or superficially understood	Difficulty working with material not understood
Strong in convergent thinking and analysis	Divergent thinking and synthesis
Learn best when information is presented in a steady progression of complexity and difficulty	Sometimes better to jump directly to more complex and difficult material

4.2.2. Step 1.2: map learning style behaviour to the conversational tutoring style

To map learning style behaviour to the conversational tutoring style, each behaviour characteristic extracted in step 1.1b (in Table 2) is assessed using the following criteria:

1. Is it possible to map the behaviour trait onto a two-way online conversational tutorial?
2. How could the behaviour trait be used to implicitly predict learning styles?

All behaviour traits that can be mapped onto a tutorial conversation and used to predict learning styles should be included in a summary table along with a description of how they could be used to predict learning styles (Table 3).

Table 3.
Aspects of learner behaviour for predicting learning styles from a natural language tutorial dialogue.

Behaviour by Learning Style	Implication for Learning Style Prediction
Sensor	
Prefer facts, data, experimentation	Perform better in questions with facts, examples and results
Dislike surprises	Prefer introductions, overviews and working in a sequential predictable order
Careful but slow	Consider timing interactions and number of errors
Comfortable with symbols (e.g. words)	Consider amount of discussion with the tutor
Intuitor	
Prefer principles and theories	Perform better in theory questions
Dislike repetition	Present information usually only once
Bored by detail	Perform better where information is summarised
Quick but careless	Consider timing interactions and number of errors
Uncomfortable with symbols	Consider amount of discussion with the tutor
Visual	
Remember what they see	Perform better in questions with diagrams, pictures, movies
Like pictures, diagrams, flow charts, time lines, films	Perform better in questions with pictures, diagrams, flow charts, time lines, films
Prefer visual demonstration	Perform better in questions with visual walkthroughs rather than textual explanation
Verbal	
Remember what they hear, or what they hear then say	Perform better in questions with movies and sound clips
Like discussion	Consider amount of discussion with the tutor
Prefer verbal explanation	Perform better in questions with movies, sound clips and tutor explanations
Learn by explaining to others	Consider amount of discussion with the tutor
Active	
Do something with information – discuss/explain/test	Consider amount of discussion with the tutor; perform better in questions with practical exercises
Experimentalists	Perform better in practical questions
Process information by setting up an experiment to test an idea, or try out on a colleague	Consider amount of discussion with the tutor; perform better in questions with practical exercises
Reflective	
Examine and manipulate information introspectively	Consider amount of discussion with the tutor
Theoreticians	Perform better in theoretical questions
Sequential	
Follow linear reasoning processes	Perform better when information presented in a steady progression of complexity and difficulty
Learn best when information is presented in a steady progression of complexity and difficulty	Perform better when information presented in a steady progression of complexity and difficulty
Global	
Sometimes better to jump directly to more complex and difficult material	Perform better where information is summarised and when they can attempt problems in one go

Next, it is necessary to decide which aspects of behaviour need to be captured during a tutoring conversation. Each behaviour trait in Table 3 was studied in turn and the list was reorganised according to behaviour, with similar behaviours grouped together. For example, as both Verbal and Active learners like discussion, they were grouped together under the 'like discussion' behaviour category. Next, this list of behaviours was reduced further by considering the behaviour that would need to be captured from a natural language conversation. For example, the 'like discussion' category now became the 'discussion' category and included also the Sensor (like discussion), Intuitor (do not like discussion) and Reflective (do not like discussion) learning styles. The result of this analysis is a list of behaviour cues to be captured during the conversational tutorial that can be used to predict learning style. Table 4 lists the behaviour to be captured during a tutorial conversation in order to predict ILS learning styles, and relates each behaviour variable to the learning styles it may be used to predict.

Table 4.
Learner behaviour cues to be captured during tutoring.

Behaviour variable to be captured	Learning style
Number of discourse interactions	Sensor, Intuitor, Verbal, Active, Reflective
Number of questions asked	Sensor, Intuitor, Verbal, Active, Reflective

Tutorial duration	Sensor, Intuitor,
Reading time	Sensor, Intuitor,Visual, Verbal
Number of errors due to not reading the question	Sensor, Intuitor
Right answer after seeing an image	Visual
Right answer after seeing a movie/walkthrough	Visual, Verbal, Active
Right answer after an explanation of theory	Intuitor
Right answer after seeing an example	Sensor
Choose to be guided through the steps of solving a problem	Sensor, Sequential
Choose to solve a problem straight away	Intuitor, Global
Score for practical questions	Active, Sensor
Score for theoretical questions	Reflective, Intuitor

4.2.3. Step 1.3: analyse the learning styles model for language traits

Mairesse, Walker, Mehl & Moore (2007) showed that it was possible to automatically recognise an individual's personality type using language cues (such as the type of vocabulary used) from conversation and text (essays). As learning style is linked to personality (Coffield, Moseley, Hall & Ecclestone 2004), it may be possible that the type of vocabulary used can indicate an individual's learning style. Özpolat and Akar (2009) mapped a short list of key words to ILS learning styles, and analysed student Internet search terms to successfully predict learning styles for three of the four ILS dimensions. Step 1.3 of the methodology involves analysing the learning styles model to extract any language traits that could be indicative of learning style. The key words list in Ozpolat & Akar (2009) was extended by analysing the descriptions of behaviour traits in the ILS model. Indicative words and phrases used to describe behaviour traits were extracted and mapped to learning styles. This key words list was then expanded using a thesaurus to produce an initial set of key words and phrases that were indicative of learning style. For example, the key word *show* (e.g. "Can you show me an example") indicates a Visual learning style, whereas the keyword *tell* (e.g. "Can you tell me what to do") indicates a Verbal learning style. The process of discovering associations between key words and particular learning styles requires experimentation and analysis of tutoring dialogues, so the content of the list should be tested and expanded by analysing actual tutoring discourse once the Oscar CITS has been developed for a particular domain.

4.2.4. Step 1.4: adapt the generic logic rules to predict learning styles

The final step in phase 1 is to convert the knowledge of the learning styles model (the captured behaviour factors and key words gathered from steps 1.2 and 1.3) into a set of logic rules. The aim of such rules is to continually increment student learning style values as the tutoring conversation takes place. A generic set of 33 logic rules was created using the learner behaviour captured from the ILS (Table 4). As the generic logic rules relate to learner behaviour, the set should be adapted and expanded for different learning styles models that may define other behaviours. Table 5 shows two examples of logic rules developed using the behaviour cues in Table 4 and mapped to the ILS. The first example, rule 1, is generated from the behaviour cue 'Right answer after seeing an image' and is linked to the Visual learning style. If a student does not know the answer, is shown an image and then gets the answer right, this visual presentation has helped their understanding so the Visual learning style value is incremented. Rule 2 is generated from the 'Number of errors due to not reading the question' behaviour, linked to the Intuitor and Visual learning styles. If the answer to a question is given in the explanation text and a student gets the answer wrong, this behaviour indicates they are careless and not comfortable with reading text, so the Intuitor and Visual learning style values are incremented.

Table 5.
Example logic rules to adjust student learning style values based on tutoring conversation.

1. Example rule to test whether presenting information visually helps the student's information perception:

```
IF      student shown image/diagram
AND     student gives correct answer
THEN    increase VISUAL;
```

2. Example rule to test how comfortable the student is with words and with detail:

```
IF      answer is given in the explanation text
AND     student does not know the answer
THEN    increase INTUITOR
AND     increase VISUAL;
```

The set of logic rules resulting from this step are to be applied during a tutoring conversation to dynamically predict learning styles.

This section has described the steps in phase 1 of the generic Oscar CITS methodology to create a Learning Styles Predictor module using the ILS model as an example.

4.3. Methodology phase 2: design a tutorial conversation

Phase 2 of the methodology involves capturing the tutorial from human tutors and iteratively developing a tutorial conversation with input from the human tutors. This part of the methodology will be illustrated using an example tutoring domain of the database Sequential Query Language (SQL).

4.3.1. Step 2.1: capture the tutorial scenario and questions (including movies, voice, images, examples, etc.) from human tutors in a specific domain

The first step in designing a tutorial conversation is to capture a tutorial scenario from human tutors. The domain of SQL was selected as the target audience for the pilot study is undergraduate computing students, for whom a Databases course including SQL is compulsory. First, interviews were conducted with undergraduate level database course tutors to identify important SQL concepts for the tutorial syllabus. Ten tutorial questions and a multiple choice question (MCQ) test were devised to cover the learning outcomes of the tutorial. To capture the tutorial scenario, a document was produced in consultation with lecturers that contained a conversation script for each question, including possible learner answers and tutor's responses to these. For each learner response, a further tutor response was written, and so on, until each question in the tutorial had a number of different paths depending on individual learner knowledge and responses. The design of the tutorial conversation was a time consuming and iterative process. However, by planning and detailing the dialogue at this point, the development of the conversational agent was more efficient. Resources such as examples, movies, images etc. were embedded into the tutorial conversation as appropriate.

4.3.2. Step 2.2: determine the conversational structure/style

A CITS that attempts to mimic a human tutor must be able to manage a tutoring conversation on a number of levels, each with a different goal. Step 2.2 of the methodology determines the structure of the CA tutorial conversation. Drawing on experience of classroom tutorials (Crown copyright 2004), three parts of a tutorial conversation with separate goals were distinguished and a three-level model of a tutorial conversation was designed (Fig. 2). At the highest level (the 'social level'), Oscar CITS needs the ability to maintain a natural language tutorial conversation, and like a human tutor must pick up cues if the learner is not engaging in the tutorial (e.g. use of bad language) and choose to end the session. At the main 'tutoring level', Oscar CITS directs the tutorial, explains topics and asks questions, guiding the learner towards an understanding of the topic. This may involve Oscar CITS giving feedback on erroneous or incomplete solutions (intelligent solution analysis), explaining the topic using different methods if required, such as practical examples (curriculum sequencing) and giving hints to help the learner construct a solution (problem solving support). During a tutorial, learners may discuss a related topic to help their understanding, requiring a deeper 'discussion level' with the ability to discuss and explain a predefined set of Frequently Asked Questions related to the domain.

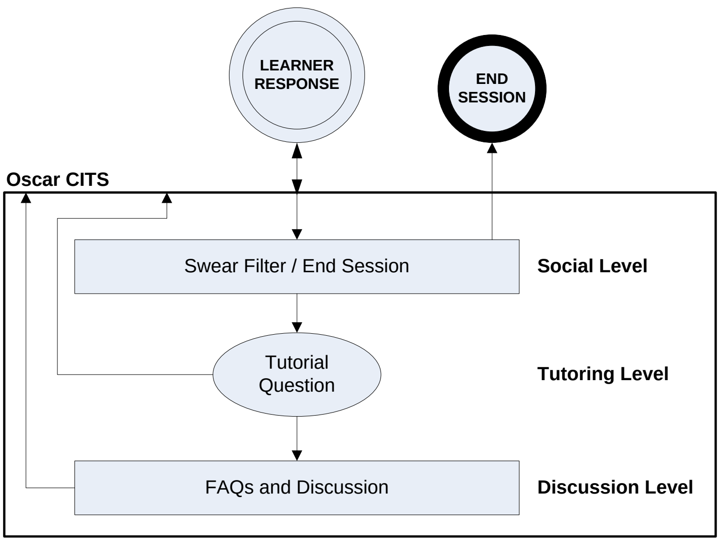


Fig. 2. 3-Level model of a tutorial conversation.

As part of this step, a list of FAQs and answers should be captured from the human tutors, scripted in natural language and added to the tutorial conversation document.

4.3.3. Step 2.3: map tutorial questions onto the generic question styles and templates

The third step in phase 2 of the methodology links the captured tutorial questions to the behaviour characteristics identified in phase 1 step 1.2. This is done by mapping tutorial questions to the set of generic question styles and templates. During the development of the Learning Styles Predictor module (Phase 1 steps 1.1 and 1.2), questions and behaviour from the ILS model were mapped to a conversational tutoring style. Applying this knowledge, a set of four generic question styles (e.g. practical and theoretical style questions) and two generic question templates were developed. The set of question styles and templates should be expanded when different learning styles models and domains are implemented.

Fig. 3 shows an example generic question template that could be applied to both practical and theoretical question styles. The template is for a question where different kinds of hints are given to learners and information is captured about the type of help that is most effective. In Fig. 3, the question is asked in box 1 and if the learner responds with the correct answer at any point, they are given feedback and taken to the next question (response 2). If the learner does not know the answer or their answer is wrong, Oscar explains the concept and repeats the question (response 3). If the learner still does not know the answer or their answer is wrong, Oscar shows different resources and repeats the question (responses 4, 5 and 6). Finally, if the learner still does not know the correct answer, Oscar tells them the answer, suggesting that they revise the topic (showing additional resource links), then asks if they wish to continue with the tutorial (response 7). If the learner wishes to continue, they are taken to the next question; if not the tutorial is ended.

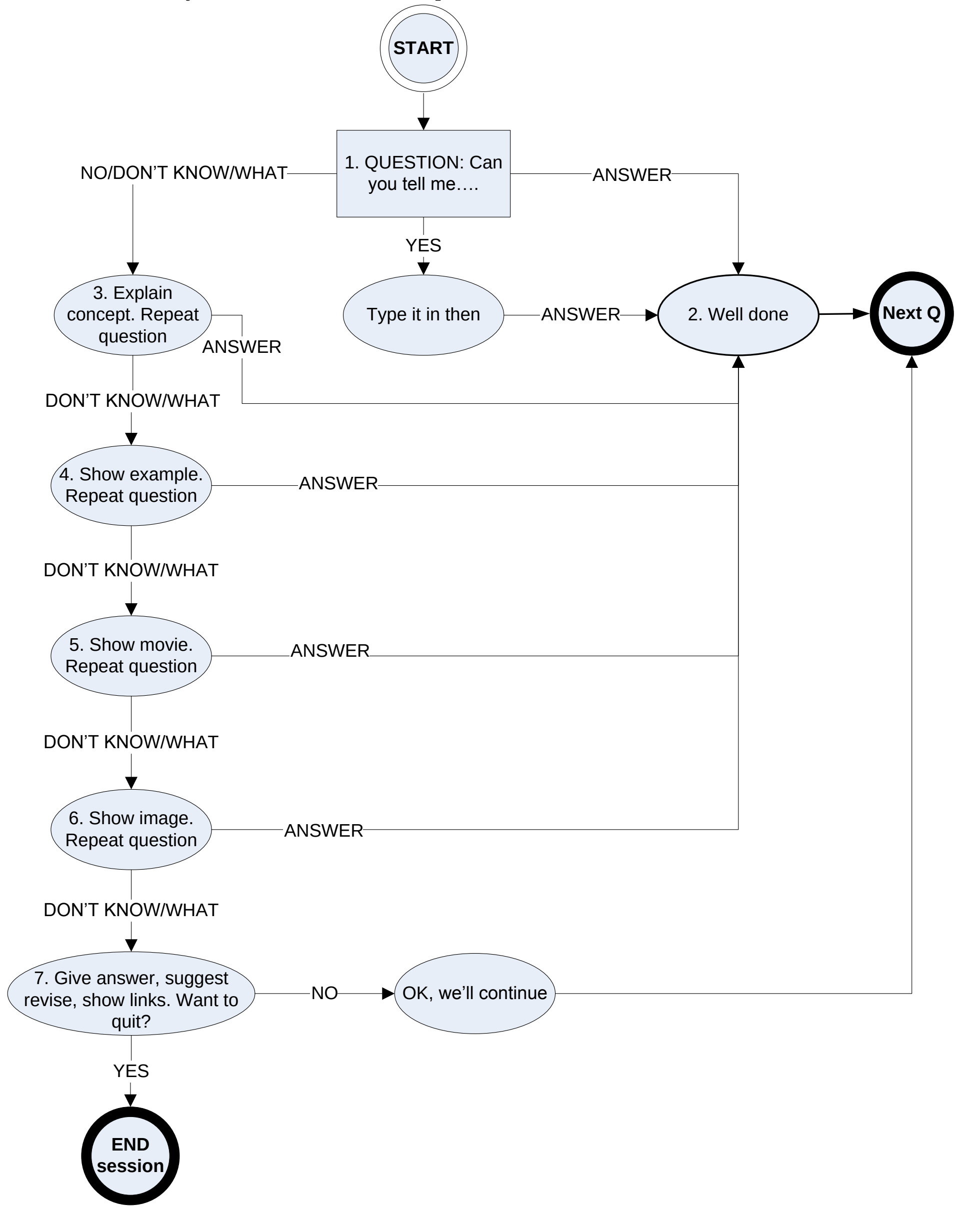


Fig. 3. Example generic question template with hints.

In this step, tutorial questions are mapped onto the generic styles and templates, with extra resources included as required, and the dialogue updated in the tutorial conversation document.

4.3.4 Step 2.4: script CA natural language dialogue for each tutorial question using the 3-level model

Step 2.4 of the methodology involves creating CA scripts to conduct the tutoring dialogue defined in steps 2.1, 2.2 and 2.3 (and recorded in the tutorial conversation document). This involves first adopting a CA that can capture and receive information using variables, then scripting the conversation using an appropriate scripting language. Convagent Ltd (2005) InfoChat CA was selected as it is a pattern matching CA that allows information to be captured using variables. CA scripts, organised into contexts, were developed for the tutorial based on the tutorial conversation document and applying the 3-level tutorial conversation model. Overall, there were 38 contexts containing around 400 rules written using the InfoChat PatternScript language (Michie & Sammut 2001). An example FAQ rule from one of the tutorial scripts is shown in Table 5. In the rule, *a* is the activation level used for conflict resolution (Michie 2001); *p* is the pattern strength followed by the pattern that is matched against the user utterance. *r* is the CA response. Also seen in the example is the wildcard (*) and macros (<explain-0>) containing a number of standard patterns that are each matched separately. When the rule fires, the variable *FAQ* is set to ‘true’ by the *<set> command.

Table 6.
Example CA script: FAQ rule.

```
<Rule-01>
a:0.01
p:50    *<explain-0> *select*
p:50    *select* <explain-0>*
p:50    *<remind-0> *select*
p:50    *select* <remind-0>*
p:50    *<confused-0> *select*
p:50    *select* <confused-0>*
r:      The SQL SELECT command is used to retrieve data from
        one or more database tables. *<set FAQ true>
```

4.3.5. Step 2.5: link tutorial dialogue to logic rules through CA variables

The final step in phase 2 of the methodology links the behaviour captured by the tutorial conversation to the set of logic rules (produced in phase 1) that predict learning styles. Moving through the tutorial conversation document, for each learner behaviour found, annotate the document with the learning style defined in the associated logic rule. The logic rules from Phase 1 (step 1.4) specify which learning styles are to be incremented when particular events occur (such as incrementing the Sensory learning style value after an example is shown). Next, the CA scripts must be updated to capture the behaviour by setting variable values when particular rules fire. Now that the tutorial conversation has been fully scripted for a CA it must be tested and verified by expert human tutors.

This section has described the steps of the generic methodology to design a tutoring conversation illustrated by the development of a tutorial for SQL using the InfoChat CA.

4.4. Methodology phase 3: construct the CITS architecture

Once the learning styles predictor module and the tutorial conversation have been designed, it is necessary to incorporate them into a CITS architecture. The CITS will require a CA that allows information to be passed in and out, a Graphical User Interface (GUI) and a Student Model. The next section will propose a standard Oscar CITS architecture that is generic and incorporates the required components.

5. Oscar CITS architecture

The proposed Oscar CITS architecture is shown in Fig. 4. The Oscar CITS is independent of the learning styles model adopted and the subject domain being taught. As such, the proposed Oscar CITS architecture is modular, allowing individual components to be reused or replaced as necessary. The proposed generic architecture allows alternative tutorial knowledge bases and CA scripts developed following phase 2 of the methodology to be simply ‘plugged in’ to adapt the tutoring to new subjects. Similarly, different learning styles models may be applied by replacing the Learning Styles Predictor component (created following the methodology phase 1).

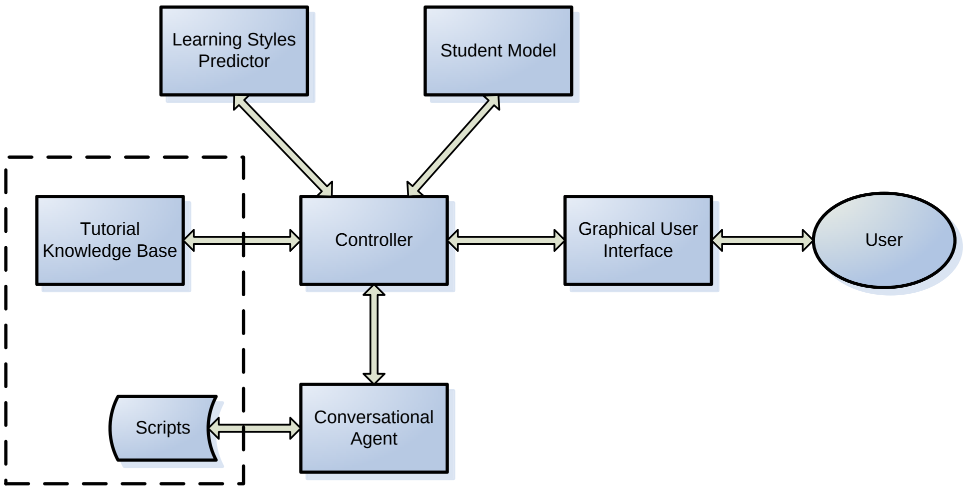


Fig. 4. Oscar CITS system architecture

Each component in the proposed architecture will now be briefly described.

- The *Controller* is the central manager of the system, responsible for instantiating objects and system variables, communicating with all components and managing the learner interaction.
- The *Learning Styles Predictor* component receives information from the CA, GUI and student model to predict a student's learning style, using information about learning styles held in a knowledge base. This module is developed following phase 1 of the Oscar CITS methodology.
- The *Student Model* component receives and sends information from and to the controller about the student, such as their level of knowledge, topics visited, test scores and learning style.
- The *Graphical User Interface (GUI)* component is responsible for display, managing events (such as clicking of buttons) and sending communication to and from the user. The display consists of a webpage that provides instructions, displays questionnaires, tests, images, documents, interactive movies and the chat area used to communicate with the user.
- The *Tutorial Knowledge Base* is responsible for managing course information, such as topics and their breakdowns, related tests and teaching material. The tutorial knowledge base receives information and instructions from the GUI, learning styles predictor and CA components via the controller, and sends information to the GUI and CA via the controller.
- The *Conversational Agent* component is responsible for accepting natural language text and information about topic and learning style from the GUI, tutorial knowledge base and learning styles components via the controller, and generating a natural language response. The CA accesses a database of tutorial conversation scripts (related to but not linked to the tutorial knowledge base) in order to match the input to rules that generate a response. The CA records the dialogue in log files that can be accessed by the controller.

A modular, generic architecture and an original, generic methodology have been proposed for creating an Oscar CITS. The Oscar CITS architecture has been designed with component reuse in mind, and can be adapted for different learning styles models by following phase 1 of the Oscar CITS Methodology to develop another learning styles predictor module. Similarly, different subject domains can be applied by following phase 2 of the Oscar CITS Methodology to develop the tutorial conversation. The next section will describe the experiments carried out to validate the proposed Oscar CITS methodology and architecture.

6. Experiments

The Oscar CITS was implemented and tested by real university students in a real teaching/learning environment in order to:

- validate the Oscar CITS prediction of learning styles from a natural language tutoring dialogue;
- analyse the effectiveness of Oscar CITS as a learning tool;
- study the impact of the Oscar CITS natural language tutoring on students.

Oscar CITS was implemented to deliver an SQL revision tutorial by applying the methodology and architecture proposed in sections 4 and 5. First, the ILS model was adopted and analysed following Phase 1 of the Methodology described in section 4 to develop the Learning Styles Predictor module. In the next phase of the Methodology (phase 2) a ten question SQL revision tutorial was captured from university lecturers and the generic tutorial question templates and styles were applied. A 12 question MCQ test was devised to assess the tutorial learning outcomes. The InfoChat pattern-matching CA (Convagent Ltd 2005) was adopted, and the tutorial conversation was scripted using its PatternScript language (Michie & Sammut 2001). The logic rules developed for the Learning Styles Predictor module were then mapped to the CA scripts to ensure that relevant behaviour was captured using variables. In Phase 3 of the methodology, the proposed Oscar CITS architecture was implemented using the .net framework and mySQL, and the Oscar CITS was installed onto a web server. The Oscar CITS is at present available via the Internet to Manchester Metropolitan University (MMU) students. Oscar CITS conducts its tutoring conversations in real time and is currently being used to support a number of undergraduate and postgraduate computing modules within MMU. The Oscar CITS GUI is shown in Fig. 5.

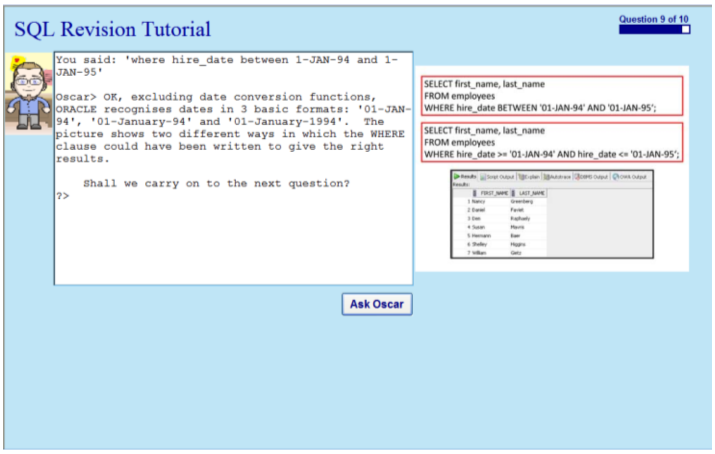


Fig. 5. Oscar CITS

The experiments described in this paper have been selected from a larger study to demonstrate how different types of behaviour may be used to predict learning styles.

6.1. Experimental design

As the aim of the experiments is threefold, the Oscar CITS will be evaluated on three levels:

1. Can Oscar CITS predict learning styles dynamically from a two-way tutoring discourse? How successful is the prediction of learning styles? The Oscar CITS prediction of learning styles will be measured against the results of the ILS questionnaire. The main hypothesis 'it is possible to predict learning style from a two-way tutoring conversation' was broken down into five hypotheses (H) as follows:
   - H1: the success of a learner after experiencing a particular style of tutoring is indicative of learning style.
   - H2: a lack of attention to detail in answering questions is indicative of learning style.
   - H3: choosing to be guided through a process (or not) is indicative of learning style.
   - H4: the success of a learner in a particular style of tutoring question (theoretical or practical) is indicative of learning style.
   - H5: a learner's reading time is indicative of learning style.
2. Does Oscar CITS successfully tutor learners, i.e. do they learn anything? Learning gain will be evaluated by comparing the MCQ pre-test score (completed before the tutoring conversation begins) to the MCQ post-test score (completed after the tutoring conversation ends) to see whether test scores have improved, as follows:

   Learning_gain = post-test_score – pre-test_score
3. How comfortable and confident do learners feel in using the tutoring system, and would they use Oscar CITS in practice? Satisfaction from the learners' perspective will be determined via a questionnaire using a set of subjective metrics. The design of the evaluation questionnaire was based on a user satisfaction questionnaire for rating dialogues with text-based CAs (O'Shea, Crockett & Bandar 2011). The questionnaire requires participants to rate aspects of the Oscar CITS tutorial using a six-point Likert scale (which forces participants to express a positive or negative opinion). Additionally, open questions were included to capture positive and negative comments.

Participants

This paper presents results collated from two studies and evaluated on all three levels. The studies had different participants who had no previous experience using Oscar CITS.

- *Study 1* – An initial pilot study was undertaken to explore whether the implementation of Oscar CITS was successful in tutoring and whether sufficient information was captured to predict learning styles. Ten participants were chosen whose first language was English and who had previous experience of an undergraduate ORACLE SQL course (but with various levels of expertise).
- *Study 2* – There were 104 participants who had previous experience of an undergraduate SQL course and various levels of SQL expertise. Participants were second and final year undergraduate students on a computer science degree at MMU. The Oscar CITS SQL revision tutorial was integrated into the first teaching week and during the timetabled classes, participants were asked to complete the revision tutorial. In order to promote full completions of the tutorial, participants who completed the Oscar CITS revision tutorial were awarded marks in recognition of their engagement.

6.2. Experimental methodology

Study 1 was a controlled study that took place in an office setting where participants could be unobtrusively observed during their Oscar CITS tutorial. Participants completed the tutorial individually in a single session. Study 2 was undertaken in several computer laboratories. Participants started the Oscar CITS revision tutorial in the laboratories, and those who did not complete the tutorial in a single session were able to continue the revision via the Internet at another time.

Each participant registered with the Oscar CITS anonymously, which involved being assigned a user ID and creating a password, that were recorded in the student model. Next, participants completed the formal ILS questionnaire, also recorded in the student model. Before starting the conversational tutorial, participants completed a pre-tutorial 12 question MCQ test, known as the pre-test, to assess their existing SQL knowledge. The pre-test results were stored in the student model. Next, Oscar CITS directed a two-way conversational SQL revision tutorial that took on average approximately 43 minutes, with each participant following an individual learning path depending on their existing knowledge and the dialogue. During the tutorial, the participant dialogue was recorded in log files along with captured aspects of participant behaviour. There were ten main SQL tutorial questions. At the end of the tutorial, participants completed the same MCQ test (known as the post-test) to assess their learning gain, with the results being stored in the student model. Next, Oscar CITS presented participants with a comparison of their test results (indicating their learning gain) and some feedback on their tutorial performance. Finally, participants were asked to complete a user evaluation questionnaire. For the purpose of the experiments, the participant behaviour data recorded during tutorial interactions was analysed to generate a learning styles prediction after all tutorials were complete (rather than during the tutorial conversation like the full working system). The next section will describe the analysis of participant behaviour for the five reported experiments.

6.2.1. Analysis of participant behaviour

Experiment 1: logic rules

This experiment relates to a participant's individual learning path during the tutorial. During the tutorial, logic rules increment associated learning style scores when particular behaviour occurs. For each ILS dimension the two related learning style scores were compared to give a prediction of learning style for that dimension. For example, for the processing dimension if the score for Active is higher than the score for Reflective, the participant is predicted to be Active. Where scores were equal, the learning style dimension remained unclassified and was excluded from the analysis. To calculate the prediction accuracy, the predicted learning style for each dimension was compared to the ILS questionnaire results. The number of correct predictions for each learning style was counted to produce an accuracy value that is the percentage of correct predictions for each learning style. This experiment tests the hypotheses H1, H2 and H3 and generated prediction accuracies for all learning style dimensions.

Experiment 2: tutorial question style

This experiment considered the style of tutorial questions where participants gave the correct answer by counting the number of correct theoretical and the number of correct practical questions. The number of correct answers of each style was compared, taking into consideration the possible number of correct answers for theoretical and practical questions, using the formula below:

$$\frac{\text{Correct practical questions}}{\text{Total practical questions}} \quad \text{compared to} \quad \frac{\text{Correct theoretical questions}}{\text{Total theoretical questions}}$$

Participants who performed equally well in both styles of question were unclassified and excluded from the analysis. Where participants performed better in practical questions, the Oscar CITS predicted their learning style to be Active and Sensory. Participants who performed better in theoretical questions were predicted to be Reflective and Intuitive. The Oscar CITS prediction was compared to the ILS questionnaire results and the number of correct predictions counted for each learning style, to produce a prediction accuracy percentage. This experiment tests the hypothesis H4 and generated prediction accuracies for the perception (Sensory/Intuitive) and processing (Active/Reflective) ILS dimensions.

Experiment 3: approach to queries

In Experiment 3, the learner's approach to writing queries was considered. Two questions in the tutorial applied a generic question template (methodology step 2.3) with a choice of approach to writing SQL queries to solve a problem. For each question, participants who attempted the query straight away were predicted to be Global learners whilst participants who asked for guidance were predicted to be Sequential learners. Each participant had two predictions, one for each question. The predicted learning style was compared to the ILS questionnaire results, and the number of correct predictions counted for each learning style to produce a prediction accuracy percentage. This experiment tests the hypothesis H3 and generated prediction accuracies for the perception (Sensory/Intuitive) and understanding (Sequential/Global) ILS dimensions.

Experiment 4: attention to detail

One tutorial question applied a generic 'trick question' style (methodology step 2.3), that includes the answer in the explanatory text to test the participant's attention to detail and reading skills. Participants who did not answer the question correctly were predicted to be Visual and Intuitive learners, whereas those who answered correctly were predicted to be Verbal and Sensory learners. The predicted learning style was compared to the ILS questionnaire results, and the correct predictions counted for each learning style to produce a prediction accuracy percentage. This experiment tests the hypothesis H2 and generated prediction accuracies for the perception (Sensory/Intuitive) and the input (Visual/Verbal) ILS dimensions.

Experiment 5: reading time

Experiment 5 considers a participant's aptitude with words by investigating their reading speed. As each learner follows an individual learning path, calculating reading time from the total number of words read over the duration of the tutorial would not produce a fair comparison. The only text common to all participant interactions is the introductory text for the first tutorial question, so reading time was defined as the time taken to read this text. Each participant's reading time was then compared to the average (both mean and median) reading time across the sample. Where a participant had an above average reading time, Oscar CITS predicted they were Sensory and Visual learners, and where they had a below average reading time, they were predicted to be Intuitive and Verbal learners. The predicted learning style was compared to the ILS questionnaire results, and the correct predictions counted for each learning style to produce a prediction accuracy percentage. This experiment tests the hypothesis H5 and generated prediction accuracies for the perception (Sensory/Intuitive) and the input (Visual/Verbal) ILS dimensions.

7. Results and discussion

There were 114 participants over both studies, with 75 participants completing the full revision tutorial. The distribution of learning styles across the 75 participants was approximately equal for all but the Visual/Verbal dimension, which contained many more Visual than Verbal learners. This finding is consistent with the ILS model, which states that "most people of college age and older are visual" (Felder & Silverman 1988). This has implications for the analysis of results for predicting the Visual/Verbal learning styles, as the dataset is so biased towards the Visual learning style. The distribution of the 75 participants is shown in the first row of Table 7 (prior probability). The experimental results will now be discussed.

7.1. Experimental results

Table 7 shows the prediction accuracy results, representing the ability of Oscar CITS to predict a participant's learning style for that experimental measure. Experiments 3, 4 and 5 did not require the completion of the entire tutorial and so the number of participants analysed is higher. The *prior probability* is the accuracy of predicting a learning style based on the distribution of learning styles across the sample. This is included as a fairer comparison than simply using 50% because the spread of learning styles across the sample is not exactly equal. This is particularly true for the Visual/Verbal dimension where 87% of participants are Visual. Each experiment's results will now be discussed.

Table 7.
Experimental results: accuracy of prediction of learning styles.

	n	Sensory	Intuitive	Visual	Verbal	Active	Reflective	Sequential	Global
Prior probability	*75*	*60%*	*40%*	*87%*	*13%*	*57%*	*43%*	*60%*	*40%*
Experiment 1 – logic rules	75	4%	80%	68%	10%	100%	0%	82%	33%
Experiment 2 – tutorial question style	75	36%	50%	-	-	53%	73%	-	-
Experiment 3a – approach to queries (Q5)	89	65%	38%	-	-	-	-	74%	48%
Experiment 3b – approach to queries (Q9)	76	70%	56%	-	-	-	-	70%	61%
Experiment 4 – attention to detail	94	59%	28%	94%	17%	-	-	-	-
Experiment 5 – reading time	95	51%	78%	47%	71%	-	-	-	-

Experiment 1: logic rules

Using this measure, Oscar CITS was able to predict three learning styles with higher accuracy than the prior probability – Intuitive (80%), Active (100%) and Sequential (82%). For the Visual learning style, even though Oscar CITS accurately predicts Visual participants in 68% of cases, the unequal spread of participants for this dimension means that this is not significant when compared to the prior probability of 87%. This measure was not able to predict the Reflective learning style, probably because Reflective learners spend time after the learning experience reflecting on what they know and put it together as knowledge. The results support hypotheses H1, H2 and H3 and show that logic rules are the most successful factor in predicting the Intuitive, Active and Sequential learning styles.

Experiment 2: tutorial question style

70 participants showed a preference for practical or theoretical tutorial questions; those participants whose success was the same for both question styles remained unclassified. Oscar CITS was able to predict the Intuitive (50%) and Reflective (73%) learning styles better than the prior probability. The results support hypothesis H4 and show that tutorial question style was the most successful factor in predicting the Reflective learning style, with the accuracy of 73% being far better than the prior probability of 43%.

Experiment 3: approach to queries

This experiment predicted learning styles depending on a participant's approach to writing queries. Table 7 reports results for two relevant tutorial questions as Experiments 3a and 3b. 89 participants completed question 5 (Experiment 3a) and 76 participants completed question 9 (Experiment 3b). Apart from the Sequential learning style, results for the second question were higher – probably because having experienced the style of question before, participants has a better idea of their preferred approach. All learning styles (except the Intuitive in experiment 3a) were predicted with higher accuracy than the prior probability. Experiment 3b was the most successful factor in predicting the Sensory (70%) and Global (61%) learning styles, and the results support hypothesis H3.

Experiment 4: attention to detail

94 participants had completed the 'trick question'. For the Sensory/Intuitive learning style dimension, the prediction accuracies of 59% and 28% are worse than the prior probability for the sample of 62% and 38% respectively. However, predictions for the Visual/Verbal learning style dimension were better than the prior probability at 94% and 17% respectively, with this measure producing the most accurate prediction overall for the Visual learning style. Therefore the results support hypothesis H6, a lack of attention to detail in answering questions is indicative of learning style.

Experiment 5: reading time

Reading time was calculated for 95 participants who had completed Question 1. The results were mixed, with poor predictions of Intuitive and Visual participants (those with a below average reading time) but good predictions of Sensory and Verbal participants (those with above average reading times). The prediction accuracies for the Intuitive (78%) and Verbal (71%) learning styles are much higher than the prior probabilities of 40% and 13% respectively. The results show that this measure is the best predictor of Verbal learning style, thus supporting the hypothesis H5. However, it must be borne in mind that the uneven spread of participants for the Visual/Verbal dimension prevents firm conclusions from being drawn.

7.2. Learning gain

Table 8 shows the participant learning gain results, with a total average test score improvement of 13%. Average learning gain was higher for study 1 (20%), which probably reflects the higher motivation of participants in completing the tutorial as this was a controlled setting. Study 2 involved real students completing the tutorial in a real educational environment, and so a lower learning gain was expected due to factors such as distractions. The results suggest that Oscar CITS did help learning as participants increased their learning of SQL and improved their test results.

Table 8.
Learning gain results.

Study	n	Learning gain Mean (/12)	Standard deviation	Mean %
Study 1	10	2.4	2.01	20%
Study 2	63	1.44	2.07	12%
Total	73	1.58	2.07	13%

7.3. Participant evaluation

In general, the participant feedback showed that Oscar CITS was well received, understandable and helpful. 46 participants completed the evaluation questionnaire. 87% of participants rated the tutoring highly, with 51% awarding the tutoring the highest rating. 94% of participants found the tutoring helpful, with 72% giving the highest rating. An astounding 35% of participants stated that they would use Oscar CITS tutorial *instead* of attending a face-to-face tutorial. Slightly more than half of the sample (52%) would use Oscar CITS instead of reading a book, and 85% of participants would use Oscar CITS to support classroom tutoring. Overall, 89% of participants would use a resource like Oscar CITS if it were available. When openly asked for comments about Oscar CITS, half of the participants remarked that Oscar was easy to use and 43% noted that Oscar CITS was helpful. One participant commented "*is like having your own friendly tutor*", and another "*it gives instant feedback unlike a traditional test*". From these results it can be concluded that most people found the Oscar CITS tutoring easy to use, helpful, and would use Oscar CITS to support their studies.

7.4. Results summary

The experiments were conducted using real university students in a real teaching/learning environment. The results support the hypotheses and show that by adopting the Oscar CITS methodology and architecture, it is possible to successfully predict learning styles from a two-way natural language tutoring conversation. Oscar CITS helped participants to increase their knowledge and participants valued the Oscar CITS learning experience and would use Oscar CITS to support learning. Table 9 summarises the best prediction accuracies resulting from the five experiments described. In a full Oscar CITS learning style values are adjusted dynamically throughout the tutorial conversation based on learner behaviour, apart from the Reflective learning style, where the preferred question style is tested periodically at the end of each tutorial.

Table 9.
Oscar CITS best prediction accuracy.

	n	Sensory	Intuitive	Visual	Verbal	Active	Reflective	Sequential	Global
Oscar CITS	75-95	70%	80%	94%	71%	100%	73%	82%	61%

The methodology and architecture for Oscar CITS are independent of the learning styles model and subject domain chosen. Although the results show the successful prediction of ILS learning styles, before conclusions may be drawn about non-computing subject domains it is necessary to implement Oscar CITS and empirically test its prediction of learning styles with different models.

A comparison of results with other CITS is not possible as no other CITS can predict learning styles. On a superficial level, the results compare favourably with menu-based ITS that predict ILS learning styles (Ozpolat & Akar 2009; Cha *et al* 2006; Garcia *et al* 2007). However it is inappropriate to compare prediction accuracies with these ITS because, despite adopting the ILS, they classify learning styles differently, introducing a third 'Neutral' class for each dimension which describes learners with low strength learning styles (i.e. those at the centre of the dimension). Also, the method of calculating prediction accuracy for these ITS uses different scoring, by awarding a 0.5 score if the learning style prediction is mismatched with a Neutral classification, rather than a zero score for all mismatches used by Oscar CITS.

8. Conclusions

This paper has presented the Oscar Conversational Intelligent Tutoring System, a novel CITS which implicitly predicts and adapts to learning styles whilst directing a tutorial conversation. Oscar CITS imitates a human tutor by incorporating the intelligent tutoring techniques of curriculum sequencing, intelligent solution analysis and problem solving support. A tutorial is directed by Oscar CITS, which detects behaviour cues from learners to present learning material suited to their knowledge and learning style. Learners can participate in a personalised tutorial via the Internet, learning at their own pace at a time and place to suit them. Oscar's conversational style is intuitive to use, helping to improve motivation and build confidence, with one user remarking "*it encouraged me to think rather than simply giving me the answer".*

An original methodology and architecture for creating the Oscar CITS were described, which are independent of the learning styles model and subject domain being taught. The 3-phase methodology describes the development of the Learning Styles Predictor, Tutorial Knowledge Base and CA components and includes a number of generic tools to aid development (behaviour variables, key words, logic rules, 3-level conversation model, question styles and templates). The generic architecture is modular, allowing different learning style models and subject domains to be applied whilst supporting the reuse of components.

Oscar CITS was implemented to deliver an SQL revision tutorial and evaluated empirically by real students in a real educational setting. The experimental results show that it is possible to predict learning styles from a two-way natural language tutoring conversation. Oscar CITS successfully predicted all learning styles in the Index of Learning Styles model, with accuracies ranging from 61-100%. Oscar CITS was well received by participants, who found it helpful, easy to use and successful in improving their knowledge.

Further work has been done in analysing different sorts of behaviour for predicting learning styles from natural language, including a preference for practical or theoretical questions, the number of words used, the amount of discussion, duration and vocabulary. An algorithm is now being developed to improve the accuracy predicting learning styles using a fuzzy set representation that combines different aspects of learner behaviour captured from a natural language tutorial. An Oscar CITS adaptation algorithm has been designed that selects the best fitting adaptation for each tutorial question by combining student learning styles with available teaching styles (Latham, Crockett, McLean & Edmonds, 2011). In future, a speech module could be incorporated into the Oscar CITS architecture to facilitate spoken tutorial conversations.

Acknowledgements
The research presented in this paper was funded by EPSRC. The authors thank ConvAgent Limited for the use of their InfoChat CA and PatternScript scripting language.

References

Ammar, M. B., Neji, M., Alimi, A. M. & Gouarderes, G. (2010). The Affective Tutoring System. *Expert Systems with Applications 37*, 3013-3023.

Arnott, E., Hastings, P. & Allbritton, D. (2008). Research Methods Tutor: Evaluation of a dialogue-based tutoring system in the classroom. *Behaviour Research Methods 40* (3), 694-698.

Brooks, C., Greer, J., Melis, E. and Ullrich, C. (2006). Combining ITS and eLearning Technologies: Opportunities and Challenges. *ITS 2006, LNCS 4053*, 278-287.

Brusilovsky, P. & Peylo, C. (2003). Adaptive and Intelligent Web-based Educational Systems. *Int. J. Artificial Intelligence in Education 13*, 156-169.

Cha, H. J., Kim, Y. S., Park, S. H., Yoon, T. B., Jung, Y. M. & Lee, J. H. (2006). Learning styles diagnosis based on user interface behaviours for the customization of learning interfaces in an intelligent tutoring system. *ITS 2006, LNCS 4053*, 513-524.

Chi, M.T.H., Siler, S., Jeong, H., Yamauchi, T. & Hausmann, R.G. (2001). Learning from human tutoring. *Cognitive Science 25*, 471-533.

Coffield F., Moseley D., Hall E. & Ecclestone K. (2004). *Learning Styles and Pedagogy in Post-16 Learning: A Systematic and Critical Review*. London: Learning and Skills Research Center.

Convagent Ltd (2005), *Convagent*, Available: http://www.convagent.com/

Crown copyright (2004). *Pedagogy and Practice: Teaching and Learning in Secondary Schools Unit 1: Structuring learning*, DfES Publications.

D'Mello, S., Lehman, B., Sullins, J., Daigle, R., Combs, R., Vogt, K., Perkins, L. & Graesser, A. (2010). A Time for Emoting: When Affect-Sensitivity Is and Isn't Effective at Promoting Deep Learning. *ITS 2010, LNCS 6094*, 245-254.

Felder, R. & Silverman, L.K. (1988). Learning and Teaching Styles in Engineering Education. *J. Engineering Education 78* (7), 674-681.

Garcia, P., Amandi, A., Schiaffino, S. & Campo, M. (2007). Evaluating Bayesian networks' precision for detecting students' learning styles. *Computers & Education 49*, pp. 794–808.

Honey, P. & Mumford, A. (1992). *The manual of learning styles*, Peter Honey, Maidenhead.

Hsieh, S. W., Jang, Y. R., Hwang, G. J., & Chen, N. S. (2011). Effects of teaching and learning styles on students' reflection levels for ubiquitous learning. *Computers & Education 57*, 1194-1201.

Kelly, D. & Tangney, B. (2006). Adapting to intelligence profile in an adaptive educational system. *Interacting with Computers 18*, pp. 385-409.

Khoury, R., Karray, F. & Kamel, M.S. (2008). Keyword extraction rules based on a part-of-speech hierarchy. *Int. J. Advanced Media and Communication 2* (2), 138-153.

Klasnja-Milicevic, A., Vesin, B., Ivanovic, M. & Budimac, Z. (2011). E-Learning personalization based on hybrid recommendation strategy and learning style identification. *Computers & Education 56*, 885--899.

Latham, A., Crockett, K. & Bandar, Z. (2010). A Conversational Expert System Supporting Bullying and Harassment Policies. *In: Proc. ICAART 2010*, 163-168.

Latham, A., Crockett, K., McLean, D. & Edmonds, B. (2009). Using Learning Styles to Enhance Computerised Learning Systems, in *Proc. 2009 Annual Research Student Conference, Manchester Metropolitan University*, Manchester, UK.

Latham, A.M., Crockett, K.A., McLean, D.A. & Edmonds, B. (2010). Predicting Learning Styles in a Conversational Intelligent Tutoring System. *ICWL 2010, LNCS 6483*, 131-140.

Latham, A.M., Crockett, K.A., McLean, D.A., Edmonds, B. & O'Shea, K. (2010). Oscar: An Intelligent Conversational Agent Tutor to Estimate Learning Styles. *In: Proc. IEEE WCCI 2010*, 2533-2540.

Latham, A.M., Crockett, K.A., McLean, D.A. & Edmonds, B. (2011). Oscar: An Intelligent Adaptive Conversational Agent Tutoring System. *LNAI 6682,* 563-572.

Leontidis, M. & Halatsis, C. (2009). Integrating Learning Styles and Personality Traits into an Affective Model to Support Learner's Learning. *ICWL 2009, LNCS 5686*, 225-234.

Li, Y., Bandar, Z., McLean, D. & O'Shea, J. (2004). A Method for Measuring Sentence Similarity and its Application to Conversational Agents. *In Proc. FLAIRS 2004*, 820-825.

Mairesse, F., Walker, M., Mehl, M. & Moore, M. (2007), Using Linguistic Cues for the Automatic Recognition of Personality in Conversation and Text, *J. Artificial Intelligence Research 30*, 457-501.

Melis, E., Andrès, E., Büdenbender, J., Frishauf, A., Goguadse, G., Libbrecht, P., Pollet, M. & Ullrich, C. (2001). ActiveMath: A web-based learning environment. *Int. J. Artificial Intelligence in Education 12* (4), pp. 385-407.

Michie, D. (2001). Return of the Imitation Game. *Electronic Transactions on Artificial Intelligence 6*, 203-221.

Michie, D. & Sammut, C. (2001). *Infochat Scripter's Manual*, ConvAgent Ltd, Manchester, UK.

Mitrovic, A. (2003). An Intelligent SQL Tutor on the Web. *Int. J. Artificial Intelligence in Education 13*, 171-195.

O'Shea, J., Bandar, Z. & Crockett, K. (2011), Systems Engineering and Conversational Agents, in Tolk, A. & Jain, L.C. (Eds), *Intelligence-Based Systems Engineering*, Intelligent Systems Reference Library 10, Springer-Verlag Berlin Heidelberg.

O'Shea, K., Crockett, K. & Bandar, Z. (2011), An Approach to Conversational Agent Design using Semantic Sentence Similarity, *IEEE Transactions on Systems, Man and Cybernetics, under review*.

Özpolat, E. & Akar, G.B. (2009). Automatic detection of learning styles for an e-learning system. *Computers & Education 53*, pp.355-367.

Papanikolaou, K.A., Grigoriadou, M., Kornilakis, H. & Magoulas, G.D. (2003). Personalizing the Interaction in a Web-based Educational Hypermedia System: the case of INSPIRE. *User Modeling and User-Adapted Interaction 13*, 213-267.

Popescu, E. (2010). Adaptation provisioning with respect to learning styles in a Web-based educational system: an experimental study. *Journal of Computer Assisted Learning 26*, 243-257.

Pudner, K., Crockett K.A. & Bandar, Z. (2007). An Intelligent Conversational Agent Approach to Extracting Queries from Natural Language. *In Proc. WCE Int. Conf. Data Mining and Knowledge Engineering 2007*, 305-310.

Sammut, C. (2001). Managing Context in a Conversational Agent. *Linkoping Electronic Articles in Computer & Information Science 3* (7). Linkoping University Electronic Press, Sweden.

Sarrafzadeh, A., Alexander, S., Dadgostar, F., Fan, C. & Bigdeli, A. (2008). "How do you know that I don't understand?" A look at the future of intelligent tutoring systems. *Computers in Human Behaviour 24*, 1342-1363.

Spallek, H. (2003). Adaptive Hypermedia: A New Paradigm for Educational Software. *Advances in Dental Research 17 (1)*, 38-42.

Wang, T., Wang, K. & Huang, Y. (2008). Using a Style-based Ant Colony System for Adaptive Learning. *Expert Systems with Applications 34*(4), 2449-2464.

Woo Woo, C., Evens, M.W., Freedman, R., Glass, M., Seop Shim, L., Zhang, Y., Zhou, Y. & Michael, J. (2006). An intelligent tutoring system that generates a natural language dialogue using dynamic multi-level planning. *Artificial Intelligence in Medicine 38*, pp. 25-46.

Yannibelli, V., Godoy, D. & Amandi, A. (2006). A genetic algorithm approach to recognise students' learning styles. *Interactive Learning Environments 14 (1)*, 55-78.

Oscar: An Intelligent Adaptive Conversational Agent Tutoring System

Annabel Latham[1], Keeley Crockett[1], David McLean[1], and Bruce Edmonds[2]

[1] Intelligent Systems Group, School of Computing, Mathematics & Digital Technology, Manchester Metropolitan University, Manchester M1 5GD, UK
[2] Centre for Policy Modelling, Manchester Metropolitan University, Aytoun Building, Aytoun Street, Manchester M1 3GH, UK
{A.Latham,K.Crockett,D.McLean,B.Edmonds}@mmu.ac.uk

Abstract. This paper presents an adaptive online intelligent tutoring system called Oscar which leads a tutoring conversation and dynamically predicts and adapts to a student's learning style. Oscar aims to mimic a human tutor by using knowledge of learning styles to adapt its tutoring style and improve the effectiveness of the learning experience. Learners can intuitively explore and discuss topics in natural language, helping to establish a deeper understanding of the topic and boost confidence. An initial study into the adaptation to learning styles is reported which produced encouraging results and positive test score improvements.

Keywords: Intelligent Tutoring System, Conversational Agent, Learning Style.

1 Introduction

The widespread use of the Internet has presented opportunities for the delivery of learning, both in terms of distance-learning and in supporting traditional classroom activities. Intelligent Tutoring Systems (ITS) extend traditional content-delivery computerised learning systems by adding intelligence which aims to improve the effectiveness of a learner's experience. This usually involves personalising the tutoring by adapting the learning material presented according to existing knowledge [1] or student affect such as emotion [2]. ITS which build in some social awareness, such as personalising tutoring to the individual, offer a more familiar and comfortable learning experience. Most ITS are menu-based and offer student-directed study and support at a time and pace to suit individuals, but offer an experience more akin to a computerised textbook than a classroom tutorial. Conversational Intelligent Tutoring Systems (CITS) incorporate more human-like natural language interfaces which allow learners to explore and discuss a topic, supporting the constructivist style of learning used by human tutors. However, creating a CITS which can converse naturally with a learner is a complex and time-consuming task, which is why only a few CITS exist [3][4]. Human tutors adapt their tutoring style and content based on cues they pick up

J. O'Shea et al. (Eds.): KES-AMSTA 2011, LNAI 6682, pp. 563–572, 2011.

from students, such as their level of existing knowledge and their learning styles. Learning styles describe the way groups of people prefer to learn, for example by trial and error or by observation [5]. A CITS which can mimic a human tutor by leading an adaptive tutorial conversation offers students a familiar format which can help improve confidence and motivation, leading to a better learning experience. There are no tutor-led CITS which can predict and adapt to learning styles during a tutoring conversation.

This paper describes a novel CITS which dynamically predicts and adapts to a student's learning style during a tutor-led conversation. The research focussed on mimicking a face-to-face tutorial and building in knowledge of learning styles rather than designing an interface specifically to pick up learning style behaviour, as in [6]. The adaptation algorithm employed recognises the importance of providing a coherent learning experience, and so considers both the student's learning style preferences and the opportunity for adaptation in tutoring questions.

In this paper, section 2 introduces the background concepts of Intelligent Tutoring Systems, the Index of Learning Styles and Conversational Agents. Section 3 describes the Oscar CITS and the methods used to incorporate adaptivity. Section 4 outlines the experimental methodology and two sample learner dialogues. Section 5 reports the results and discussion, and Section 6 describes the conclusions and future work.

2 Background

2.1 Intelligent Tutoring Systems

Intelligent tutoring systems (ITS) are computerised learning systems which adopt intelligent systems techniques to personalise the learning experience. ITS endeavour to improve the effectiveness of tutorials and boost learners' motivation and confidence by adapting to each individual's characteristics, such as existing knowledge. ITS are normally designed to be student-directed, with a system of menu choices or hyperlinks which are reordered or ranked to recommend a particular sequence to learners [7]. Whilst this design simplifies the analysis of student behaviour, it does not truly teach the students but rather assists in self-learning, and is little different to recommending chapters of a book. Although rarely employed, conversational interfaces allow a more natural, teacher-led learning experience which supports the construction of knowledge used by human tutors [8]. Examples of CITS are AutoTutor [3] and CIRCSIM-tutor [9] which both help students construct knowledge using conversational agent tutors, however neither consider learning styles during tutoring.

The three main approaches to intelligent tutoring [1] are curriculum sequencing (presenting material in a suitable sequence [7]), intelligent solution analysis (giving feedback on incomplete or erroneous solutions [10]) and problem solving support (offering intelligent assistance in finding solutions [11]). Most ITS employ curriculum sequencing based on student knowledge and also more recently user affect factors such as emotion [12], personality [13] and learning style [4]. Few ITS incorporate all three techniques as they are complex and time-consuming to develop, but the Oscar CITS presented in this paper will incorporate all three intelligent technologies by personalising learning material and discussing problems and solutions with students.

2.2 The Index of Learning Styles

The Index of Learning Styles (ILS) model [5] describes the learning styles in engineering education and their associated teaching styles. In the ILS model a student's learning styles are represented as points along four dimensions to indicate the strength as well as the nature of their learning style preference. Each learning style dimension describes a step in the process of receiving and processing of information, as shown in Fig. 1. The ILS model measures learning style with a 44-question self-assessment questionnaire. There are 16 (2^4) combinations of learning styles, for example intuitive/visual/active/global.

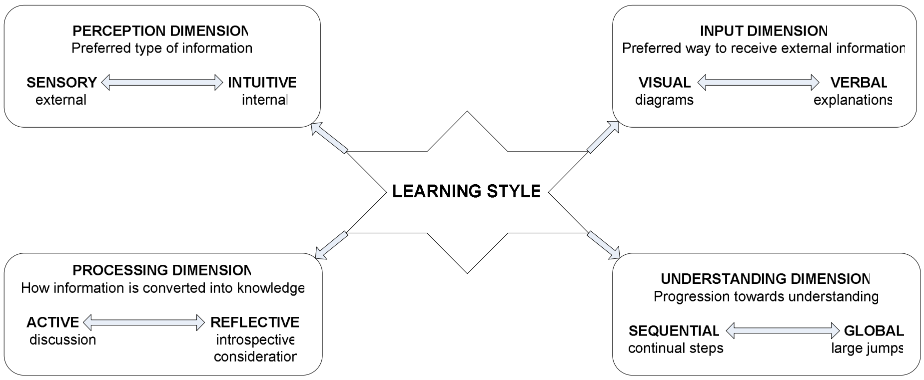


Fig. 1. ILS Dimensions

For each learning style, the ILS model details typical learner behaviours and teaching styles which address learner preferences. This information is beneficial for lecturers who informally group types of learners to adapt their teaching rather than using the formal assessment questionnaire. Knowledge of learner behaviours and teaching styles is also indispensable when developing a CITS which can adapt its teaching style to individual learner preferences.

The ILS model was incorporated into the Oscar CITS as engineering students will make up the initial experimental groups. However the flexible modular structure of the Oscar CITS does not restrict the choice of learning style model to the ILS.

2.3 Conversational Agents

Conversational agents (CAs) allow people to interact with computer systems intuitively using natural language dialogues. CA interfaces have been used effectively in many applications, such as web-based guidance [15], database interfaces [16] and tutoring [3]. CAs are complex and time-consuming to develop, requiring expertise in the scripting of conversations, and are therefore rarely found in ITS. Systems such as Oscar CITS which aim to mimic a human tutor need CA interfaces to support the construction of knowledge through discussion [8].

Textual CAs usually adopt a pattern matching [17] or semantic based [18],[19] approach. Semantic-based CAs seek to understand the meaning of the input by studying the constructs and meanings of natural language [19] or by comparing the semantic similarity of phrases [18]. Pattern-matching CAs rely on a knowledge base containing a set of pattern-based rules [16]. During a conversation user utterances are matched to rules in the knowledge base, with the best matching rule (selected by an algorithm) firing to produce a natural language response. In the case of Oscar CITS, a pattern matching approach was adopted as it can cope with grammatically incomplete or incorrect phrases, as are commonly found in text-based chat by students.

3 Oscar: An Adaptive Conversational Intelligent Tutoring System

Oscar is an online CITS which dynamically predicts and adapts to each individual student's learning style during a tutoring conversation. By adapting the tutoring style to suit individual learners, Oscar aims to provide the most appropriate learning material for their learning style, leading to a more effective learning experience and a deeper understanding of the topic. In addition to delivering tutor material suited to an individual's learning style (known as curriculum sequencing), Oscar provides intelligent solution analysis and conversational problem solving support. Like human tutors, Oscar CITS promotes a deeper understanding of the topic by using a constructivist style of tutoring, giving intelligent hints and discussing questions with learners rather than presenting the answer straight away. Oscar CITS imitates classroom tutorials with human tutors by using a natural language interface and tutor-led tutorial style which aims to help learners feel comfortable and confident during online tutorials.

The architecture and methodology for developing the original Oscar CITS is described in [20]. Results of two initial experiments which investigated the prediction of learning styles show that Oscar CITS was successful in dynamically predicting several learning styles [20],[14]. For the initial studies, Oscar delivers an online tutorial in the domain of the database Structured Query Language (SQL). Oscar draws on knowledge bases of learning styles (the ILS model), tutor material and conversation scripts to deliver a conversational tutorial to a student. To support the tutoring conversation, diagrams, images and interactive movies may be displayed. Aspects of the student's behaviour and understanding inform the dynamic prediction of learning style, allowing the tutoring style to be personalised to best suit the student.

Throughout tutoring the Oscar CITS records and logs information about the behaviour of the student, for example the timing of interactions and the type of tutor resource accessed. The tutoring conversation is also recorded, along with information about the student knowledge of the topic being discussed.

The first implementation of Oscar CITS successfully incorporated human-like intelligence into a conversational tutorial which improved student test results and dynamically predicted their learning styles. The next section will outline the extension of Oscar CITS to include the ability to adapt a tutorial to a student's learning styles.

3.1 Methods for Including Adaptivity

The Index of Learning Styles model [5] was analysed and a table of learner behaviour for each learning style drawn up. The characteristics were evaluated to establish whether they could be incorporated into a CITS. The subset of learner behaviour considered to be most important for an adaptive CITS was then assigned the appropriate teaching styles described in the ILS model. The breakdown of behaviour and teaching styles was examined further to develop several domain-independent categories of tutor material required for developing an adaptive CITS. Each tutor material category was mapped to the appropriate learning style, for example, Category 4: Practical Examples maps to the Sensor, Active and Sequential learning styles. The standard categories were designed from the point of view of the tutor and intend to make the development of tutoring material for an adaptive CITS as simple and consistent as possible. The standard organisation of tutor material also facilitates modular development, as additional materials can be expanded and added without the need for a total redesign of the tutoring session.

The next stage was to consider how the Oscar CITS would adapt tutoring according to a student's learning style. The ILS model indicates that students who have no strong learning style preference in a dimension (i.e. they are placed at the centre of the ILS scale with a score of 1 or 3) should be given learning material including a mixture of styles. An additional *Neutral* learning style category was introduced to group those students and a Neutral adaptation style included.

There are a number of possible ways to adapt to learning styles, the simplest of which would be to adapt to the student's strongest learning style. However, a tutorial is made up of a number of tutorial questions, and this approach would require incorporating every category of tutor material into every tutorial question. This may not be possible in real life, as it is important to construct a coherent tutorial and learning experience. Consequently the adaptation strategy needed to consider not only the strength of the student's learning style but also the strength of adaptation available for each individual tutorial question. This method was adopted and a complex, domain-independent adaptation algorithm was developed which combined the strengths of the student's learning style with the tutorial adaptations to select the best fitting adaptation for each question in the student's learning path.

For the initial study an SQL revision tutorial was developed for the Oscar CITS. The adaptive SQL learning material extended the tutorial delivered in previous experiments [20],[14]. This was achieved by adding different resources covering the standard categories of tutoring material. This involved creating several versions of the learning material, each suited to a different learning style. Next, each tutorial question was assigned a score for every learning style which represented the number (or strength) of opportunities for adaptation to that learning style. Where no adaptation existed for a learning style (i.e. the question score was zero) the Neutral adaptation was assigned by the algorithm. The initial study will now be described.

4 Experimental Methodology

A controlled study was conducted to test the hypothesis that students who are presented with learning material matched to their learning styles perform better than students presented with learning material which is unsuited to their learning styles. 70 final year undergraduate science and engineering students were asked to refresh their SQL knowledge by completing the Oscar CITS SQL revision tutorial. This involved each student registering with the Oscar CITS anonymously and completing the formal ILS questionnaire before beginning the tutorial. Next, students completed a pre-tutorial multiple choice question (MCQ) test to assess existing knowledge before starting the conversational tutorial. The tutorial was led by the Oscar CITS tutor who conversed in natural language with students and guided them through the ten tutorial questions, showing images, movies and examples as necessary. The conversational SQL revision tutorial took on average approximately 43 minutes, with each learner following an individual learning path depending on their knowledge and learning styles (see section 4.1 for example dialogues). After the tutorial conversation, students completed the same MCQ test and were then presented with a comparison of their test results and some feedback from Oscar. Finally, students were asked to complete a user evaluation questionnaire.

After completing the ILS questionnaire, participants were unknowingly assigned to one of three experimental groups. Students whose learning styles were at the centre of all ILS scales (i.e. there was no strong preference) were assigned to the *Neutral* group. These students followed the neutral adaptation learning path, with tutor material including different aspects of all learning styles (e.g. describing theory as well as examples). Students who had at least one preferred learning style were randomly assigned to either the *Adapt* or *Mismatch* groups using a 2:1 ratio. These students followed an adaptive learning path assigned by the algorithm, and were given tutor material favouring particular learning styles (e.g. containing explanations of theory rather than practical examples). Students in the Mismatch group were deliberately presented with learning material unsuited to their learning styles, e.g. if a student was Sequential, they were presented with material suited to a Global learner.

For each completed tutorial, log files which recorded the dialogue, timings, knowledge and other behavioural factors were analysed to assess each learner's performance during the tutoring session. During the SQL Revision tutoring session, ten questions were posed, requiring eighteen answers (as some questions incorporated multiple steps or questions). Learning gain was calculated by counting the number of correct answers given to tutorial questions, and an average score was calculated for each experimental group. The experimental group averages were then compared to determine whether there was any difference in learning gain related to the type of adaptation to learning style. In addition, the learner experiences in using Oscar CITS were evaluated by analysing the user feedback questionnaires.

4.1 Examples of Adaptive Learner Dialogues

Examples of the Oscar CITS tutoring style adaptation can be seen in the dialogue snippets logged during the experiment (Tables 1 and 2). Both snippets are taken from the same tutorial question where students are asked to write an SQL query, and both students were assigned to the Adapt experimental group (i.e. the adaptation matched their learning style). The student dialogue in Table 1 is an example of the tutoring style adapted to a Sequential learner, where Oscar guides the student through the steps of the query. Table 2 in contrast shows a student dialogue with tutoring adapted for a Global learner, where the student writes a query without initial guidance and learns from the feedback given about errors in their solution.

Table 1. Dialogue snippet logged during the experiment adapting to a Sequential learner

Utterance	Dialogue
87. Oscar>	Now let's see if you can remember how to write a query which chooses values between a range. Look at the employees table. We need a query which displays the first and last names of all employees hired between 1 January 1994 and 1 January 1995. The first step in writing a query is to state the information you need. For our query to display the first and last names of all employees hired between 1 January 1994 and 1 January 1995, which columns do we need?
88. Student>	dont know
89. Oscar>	Look again at the columns in the EMPLOYEES table. If we want to display the first and last names of all employees hired between 1 January 1994 and 1 January 1995, which columns do we need?
90. Student>	first_name, last_name, hire_date
91. Oscar>	That's right, we need FIRST_NAME and LAST_NAME, so the first part of our query is shown in the picture. The next step in writing a query is to state which table the information is retrieved from. For our query to list the first and last names of all employees hired between 1 January 1994 and 1 January 1995, type in the name of the table our information will be taken from.
92. Student>	employees
93. Oscar>	That's right, the table is EMPLOYEES. The last part of writing a query is the WHERE clause, which lets us restrict the rows selected to only those satisfying one or more conditions. We want to restrict the employees displayed to only those hired between 1 January 1994 and 1 January 1995. There are a number of ways to write this restriction in ORACLE SQL as dates can be represented in different formats. Type in the WHERE clause which limits the employees in this way.
94. Student>	where hire_date in('01-01-1994' && '01-01-1995')
95. Oscar>	One way to tackle this query is to use the BETWEEN keyword to specify a range. Complete the last line of the query shown - type in the WHERE clause to limit the employees displayed to those hired between 1 January 1994 and 1 January 1995.
96. Student>	'01-01-1994' and '01-01-1995'
97. Oscar>	OK, excluding date conversion functions, ORACLE recognises dates in 3 basic formats: '01-JAN-94', '01-January-94' and '01-January-1994'. The picture shows two different ways in which the WHERE clause could have been written to give the right results.

Table 2. Dialogue snippet logged during the experiment adapting to a Global learner

Utterance	Dialogue
64. Oscar>	Now let's see if you can remember how to write a query which chooses values between a range. Look at the employees table. Please type in a query which displays the first and last names of all employees hired between 1 January 1994 and 1 January 1995.
65. Student>	select First_name, last_name from Employees where Hire_date between 1-JAN-1944 and 1-Jan-1995;
66. Oscar>	I'm afraid your WHERE clause is not quite right. Check your spelling and type in the last part of our query which displays the first and last names of all employees hired between 1 January 1994 and 1 January 1995.
67. Student>	select First_name, last_name from Employees where Hire_date between 1-JAN-1994 and 1-Jan-1995;
68. Oscar>	No. Excluding date conversion functions, ORACLE recognises dates in 3 basic formats: '01-JAN-94', '01-January-94' and '01-January-1994'. The picture shows two different ways in which the WHERE clause could have been written to give the right results.

5 Results and Discussion

Of the 70 student participants, 54 fully completed the tutoring session. Table 3 shows the results of the experiment. The group of 8 Neutral students had learning style results which showed no strong preference for a particular learning style (i.e. their styles were balanced in the centre of the scale), and followed a neutral adaptation learning path. The Adapt group contained 32 students who followed a learning path containing learning material in a style adapted to their individual learning styles. The Mismatch group of 14 students followed an adaptive learning path of tutor material which was mismatched to their learning styles.

Table 3. Experimental results

Experimental Group	No. Students	Average Correct Answers
Neutral	8	72%
Adapt	32	73%
Mismatch	14	61%
Total	54	70%

Students in the Neutral and Adapt groups have similar averages of correct answers given during the tutoring, of 72% and 73% respectively. However, the Mismatch group has a much lower average of only 61% correct answers, which is 12% less than the Adapt group average. The results support the hypothesis that students who are presented with learning material matched to their learning styles perform better than students presented with learning material which is not matched to their learning styles.

In general, the user evaluation results showed that the Oscar CITS was well received, with 95% of learners finding the tutoring helpful and 89% agreeing that Oscar helped them to revise. 91% of the sample said that they would use the Oscar CITS resource, with 86% stating they would use Oscar to support classroom tutoring and 77% saying they would use Oscar instead of learning from a book. A surprising 50% of the sample said they would use Oscar in place of attending face-to-face tutorials. There was a 21% mean test score improvement after tutoring for students who did not achieve full marks in the initial test.

6 Conclusions and Further Work

This paper has presented a novel conversational intelligent tutoring system called Oscar, which implicitly predicts and adapts to a student's learning style during a tutoring conversation. Oscar employs a conversational agent to intelligently lead an online tutorial, mimicking a human tutor in offering students learning material adapted to their learning styles, individualised problem solving support and intelligent solution analysis. A CITS which personalises tutoring by dynamically predicting and adapting to learning styles could improve the effectiveness of a student's learning experience and help to boost confidence. Effective, personalised online tutoring could support class-based courses and widen access through distance learning.

The results of the initial study showed that students whose learning path adapted to their learning styles achieved on average 12% more correct answers than those students presented with learning material not matched to their learning styles. With regards to Oscar's conversational tutoring, the results have shown that the subjects did value the online Oscar CITS and that Oscar's tutoring seemed to help learning and improved test scores by 21% on average. It can therefore be concluded that using Oscar has helped provide students with a positive learning experience.

In future, it is planned to incorporate the tutor material categories into a toolkit to speed up the development of an adaptive CITS.

Acknowledgement. The authors thank Convagent Ltd for the use of the InfoChat conversational agent and PatternScript scripting language.

References

[1] Brusilovsky, P., Peylo, C.: Adaptive and Intelligent Web-based Educational Systems. Int. J. Artificial Intelligence in Education 13, 156–169 (2003)

[2] D'Mello, S., Lehman, B., Sullins, J., Daigle, R., Combs, R., Vogt, K., Perkins, L., Graesser, A.: A Time for Emoting: When Affect-Sensitivity Is and Isn't Effective at Promoting Deep Learning. In: Aleven, V., Kay, J., Mostow, J. (eds.) ITS 2010. LNCS, vol. 6094, pp. 245–254. Springer, Heidelberg (2010)

[3] Graesser, A., Chipman, P., Haynes, B.C., Olney, A.: AutoTutor: An Intelligent Tutoring System With Mixed-Initiative Dialogue. IEEE Trans. Education 48(4), 612–618 (2005)

[4] Popescu, E.: Adaptation provisioning with respect to learning styles in a Web-based educational system: an experimental study. Journal of Computer Assisted Learning 26, 243–257 (2010)
[5] Felder, R., Silverman, L.K.: Learning and Teaching Styles in Engineering Education. J. Engineering Education 78(7), 674–681 (1988)
[6] Cha, H.J., Kim, Y.S., Park, S.-H., Yoon, T.-b., Jung, Y.M., Lee, J.-H.: Learning styles diagnosis based on user interface behaviors for the customization of learning interfaces in an intelligent tutoring system. In: Ikeda, M., Ashley, K.D., Chan, T.-W. (eds.) ITS 2006. LNCS, vol. 4053, pp. 513–524. Springer, Heidelberg (2006)
[7] Klasnja-Milicevic, A., Vesin, B., Ivanovic, M., Budimac, Z.: E-Learning personalization based on hybrid recommendation strategy and learning style identification. Computers & Education 56, 885–899 (2011)
[8] Chi, M.T.H., Siler, S., Jeong, H., Yamauchi, T., Hausmann, R.G.: Learning from human tutoring. Cognitive Science 25, 471–533 (2001)
[9] Woo Woo, C., Evens, M.W., Freedman, R., Glass, M., Seop Shim, L., Zhang, Y., Zhou, Y., Michael, J.: An intelligent tutoring system that generates a natural language dialogue using dynamic multi-level planning. Artificial Intelligence in Medicine 38, 25–46 (2006)
[10] Mitrovic, A.: An Intelligent SQL Tutor on the Web. Int. J. Artificial Intelligence in Education 13, 171–195 (2003)
[11] Melis, E., Andrès, E., Büdenbender, J., Frishauf, A., Goguadse, G., Libbrecht, P., Pollet, M., Ullrich, C.: ActiveMath: A web-based learning environment. Int. J. Artificial Intelligence in Education 12(4), 385–407 (2001)
[12] Ammar, M.B., Neji, M., Alimi, A.M., Gouarderes, G.: The Affective Tutoring System. Expert Systems with Applications 37, 3013–3023 (2010)
[13] Leontidis, M., Halatsis, C.: Integrating Learning Styles and Personality Traits into an Affective Model to Support Learner"s Learning. In: Spaniol, M., Li, Q., Klamma, R., Lau, R.W.H. (eds.) ICWL 2009. LNCS, vol. 5686, pp. 225–234. Springer, Heidelberg (2009)
[14] Latham, A.M., Crockett, K.A., McLean, D.A., Edmonds, B.: Predicting Learning Styles in a Conversational Intelligent Tutoring System. In: Luo, X., Spaniol, M., Wang, L., Li, Q., Nejdl, W., Zhang, W. (eds.) ICWL 2010. LNCS, vol. 6483, pp. 131–140. Springer, Heidelberg (2010)
[15] Latham, A., Crockett, K., Bandar, Z.: A Conversational Expert System Supporting Bullying and Harassment Policies. In: ICAART 2010, pp. 163–168 (2011)
[16] Pudner, K., Crockett, K.A., Bandar, Z.: An Intelligent Conversational Agent Approach to Extracting Queries from Natural Language. In: Proc. WCE Int. Conf. Data Mining and Knowledge Engineering, pp. 305–310 (2007)
[17] Michie, D.: Return of the Imitation Game. Electronic Transactions on Artificial Intelligence 6, 203–221 (2001)
[18] Li, Y., Bandar, Z., McLean, D., O'Shea, J.: A Method for Measuring Sentence Similarity and its Application to Conversational Agents. In: Proc. FLAIRS 2004, pp. 820–825 (2004)
[19] Khoury, R., Karray, F., Kamel, M.S.: Keyword extraction rules based on a part-of-speech hierarchy. Int. J. Advanced Media and Communication 2(2), 138–153 (2008)
[20] Latham, A.M., Crockett, K.A., McLean, D.A., Edmonds, B., O'Shea, K.: Oscar: An Intelligent Conversational Agent Tutor to Estimate Learning Styles. In: Proc. IEEE World Congress On Computational Intelligence 2010, pp. 2533–2540 (2010)

Oscar: A Personalised Online Conversational Intelligent Tutoring System

Annabel Latham[1], Keeley Crockett[1], David McLean[1] and Bruce Edmonds[2],

[1]Intelligent Systems Group, Department of Computing & Mathematics, Manchester Metropolitan University, Manchester M1 5GD, UK

[2]Centre for Policy Modelling, Manchester Metropolitan University, Aytoun Building, Aytoun Street, Manchester M1 3GH, UK.

{A.Latham, K.Crockett, D.McLean, B.Edmonds}@mmu.ac.uk

Introduction

Intelligent tutoring systems (ITS) are computer aided learning (CAL) systems which personalise their learning content for an individual based on learner characteristics such as existing knowledge [1]. A recent extension to ITS is to capture student learning styles using a questionnaire and adapt subject content accordingly [2], however students do not always take the time to complete questionnaires carefully, so may not be shown the most effective learning material. This paper describes a web-based conversational intelligent tutoring system (CITS) called Oscar which aims to mimic a human tutor by conducting a tutoring conversation whilst dynamically predicting and adapting to a student's learning style. By implicitly modelling the student's learning style during the tutoring conversation, Oscar can personalise the delivery of material for each individual learner which improves the effectiveness of the tutoring. An initial pilot study is presented using the domain of SQL database programming for undergraduate University students. The study produced encouraging results in predicting learning styles through conversational tutoring and improving student test scores.

Design

Oscar is a novel CITS which aims to imitate a human tutor by estimating and adapting to an individual student's learning style during a tutoring conversation. A detailed description and the methodology for constructing Oscar CITS is reported in [3] and [4]. An initial study was conducted, applying Oscar CITS to the tutoring of undergraduate Science and Engineering students using the Index of Learning Styles (ILS) model [5]. There were 17 hypotheses to be tested, considering learner behaviour and language during the tutorial in relation to the four ILS dimensions. This paper presents two hypotheses (H) which consider the processing[1] (Active/Reflective) and understanding[2] (Sequential/Global) ILS dimensions:

H1: a student's learning path through the tutorial is indicative of learning style.

H2: choosing to be guided through a process (or not) is indicative of learning style.

53 students were asked to complete the formal ILS questionnaire [5], followed by a test to assess their existing SQL knowledge. The students then engaged in a personal tutorial led by Oscar CITS, involving completing tasks and answering questions, being given hints and help as required. Finally students repeated the test to measure learning.

---

[1] learners process information Actively (discussion) or Reflectively (introspectively).
[2] learners progress towards understanding Sequentially (continual steps) or Globally (large jumps).

Results

Table 1: Experimental Results

Learning Style [5]	Hypothesis	Prediction Accuracy
Active	H1	100%
Reflective	H1	0%
Sequential	H1	76%
	H2	73%
Global	H1	31%
	H2	80%

23 students were excluded as they did not complete the entire tutoring session. Table 1 shows the results of 30 students. There was a mean 22% improvement in test scores over the group.

Discussion

The results of the initial study are promising, with Oscar predicting Sequential/Global learners with 76%/80% accuracy and Active learners with 100% accuracy. However Oscar was unable to predict Reflective learners, which may be due to the nature of reflective learners who examine and manipulate information introspectively. A larger study is required before firm conclusions may be drawn.

Overall, Oscar seemed to help students learn, with a mean 22% improvement in test scores. In using tutor-led conversation rather than a student-led CAL, Oscar CITS enables a constructivist style of tutoring to be employed and is a familiar format for students. Oscar CITS can assist in widening participation by offering students the flexibility (in terms of time and place) to attend a one-to-one online tutorial in support of or in place of classroom activities.

References

[1] Brusilovsky, P., Peylo, C.: Adaptive and Intelligent Web-based Educational Systems. Int. J. Artificial Intelligence in Education 13, pp. 156-–169 (2003)

[2] Wang, H.C., Li, T.Y., Chang, C.Y.: A web-based tutoring system with styles-matching strategy for spatial geometric transformation. Interacting with Computers 18, pp. 331--355 (2006)

[3] Latham, A.M., Crockett, K.A., McLean, D.A., Edmonds, B., O'Shea, K.: Oscar: An Intelligent Conversational Agent Tutor to Estimate Learning Styles. In Proc. IEEE World Congress On Computational Intelligence 2010, pp2533—2540 (2010)

[4] Latham, A.M., Crockett, K.A., McLean, D.A., Edmonds, B.: Predicting Learning Styles in a Conversational Intelligent tutoring System.To be presented at: International Conference on Web-Based Learning (ICWL) December 2010, Shanghai.

[5] Felder, R., Silverman, L.K.: Learning and Teaching Styles in Engineering Education. J. Engineering Education 78 (7), 674--681 (1988)

Predicting Learning Styles in a Conversational Intelligent Tutoring System

Annabel Latham[1], Keeley Crockett[1], David McLean[1], and Bruce Edmonds[2]

[1] Intelligent Systems Group, Department of Computing & Mathematics, Manchester Metropolitan University, Manchester M1 5GD, UK
[2] Centre for Policy Modelling, Manchester Metropolitan University, Aytoun Building, Aytoun Street, Manchester M1 3GH, UK
{A.Latham,K.Crockett,D.McLean,B.Edmonds}@mmu.ac.uk

Abstract. This paper presents Oscar, a conversational intelligent tutoring system (CITS) which dynamically predicts and adapts to a student's learning style throughout the tutoring conversation. Oscar aims to mimic a human tutor to improve the effectiveness of the learning experience by leading a natural language tutorial and modifying the tutoring style to suit an individual's learning style. Intelligent solution analysis and support have been incorporated to help students establish a deeper understanding of the topic and boost confidence. Oscar CITS with its natural dialogue interface and classroom tutorial style is more intuitive to learners than learning systems designed specifically to capture learning styles. An initial study is reported which produced encouraging results in predicting several learning styles and positive test score improvements in all students across the sample.

Keywords: Intelligent Tutoring System, Conversational Agent, Learning Style.

1 Introduction

Intelligent Tutoring Systems (ITS) use intelligent technologies to improve the effectiveness of the student learning experience [1]. ITS can help students by providing personalised tutoring at a time and a pace to suit the individual, and offering the facility to explore in depth topics which have not been fully understood. Such benefits may not be offered in a face-to-face class full of students with varying needs and levels of expertise. Most ITS present personalised content according to student knowledge or characteristics [2], but few attempt to truly mimic a human tutor by leading the tutorial and engaging the learner in discussion [3]. A conversational intelligent tutoring system (CITS) employs a conversational agent interface to allow discourse in natural language. Human tutors pick up cues from students which indicate their learning preferences, and adapt their tutoring style accordingly. Learning styles model the way in which groups of students learn most effectively, for example by trial and error or observation [4]. Some ITS adapt tutoring to an individual's learning style, determined by using a formal questionnaire [5] or analysing learner behaviour [6]. However, there are no tutor-led CITS which can predict and adapt to learning style during the tutoring session like a human tutor.

X. Luo et al. (Eds.): ICWL 2010, LNCS 6483, pp. 131–140, 2010.

The research presented in this paper aims to develop a CITS which can dynamically predict and adapt to a learner's learning style during the tutoring session. Rather than specifically designing a learning interface to capture the learning style of the user as in [6], the focus of the research has been on imitating a human tutor and determining if it is possible to predict learning style from the student's behaviour and interaction throughout the tutorial. Whilst this considerably increases the complexity of the task of predicting learning style, conversational interfaces are intuitive to use and an ability to discuss a problem can aid the deeper learning of a topic.

In this paper, section 2 introduces some background concepts of the Index of Learning Styles [7], intelligent tutoring systems and conversational agents. Section 3 describes the Oscar CITS and presents its architecture. Section 4 outlines the experimental methodology and a sample learner dialogue. Section 5 includes the results and discussion, and Section 6 describes the conclusions and future work.

2 Background

2.1 The Index of Learning Styles

In their Index of Learning Styles (ILS) model [7], Felder and Silverman described the learning styles in engineering education and suggested different teaching styles to address learners' needs. The ILS model defines four separate dimensions of preferred learning style, each relating to a step in the process of receiving and processing information as follows:

- *Perception* – learners are *sensory* or *intuitive* depending on the type of information they prefer to perceive (e.g. external (sensory) or internal (intuitive)).
- *Input* – learners are *visual* or *verbal* according to the way they prefer to receive external information (e.g. diagrams (visual) or explanations (verbal)).
- *Processing* – learners are *active* or *reflective* according to the way information is converted into knowledge (e.g. discussion (active) or introspective consideration).
- *Understanding* – learners are *sequential* or *global* depending on their progression towards understanding (e.g. continual steps (sequential) or large jumps (global)).

The ILS uses a self-assessment questionnaire with 11 questions per learning style dimension, resulting in a score for each dimension. Each learning style dimension may be thought of as an axis with the opposite learning styles at either end (e.g. Visual versus Verbal), and the ILS questionnaire score places each learner on the axis according to the strength of their preferred learning style. There are 16 (2^4) learning styles overall (an example being sensory/visual/active/sequential).

The ILS model was chosen for the Oscar CITS as it describes engineering students, who will make up the initial experimental groups. However, the Oscar CITS is not restricted to the ILS model and its modular structure allows Oscar to be adapted to incorporate other learning style models, such as Honey and Mumford [4].

2.1.1 ILS in Practice

Whilst the ILS defines a formal questionnaire for students to identify their learning style, in practice it is not common for lecturers to use a formal tool when planning to teach a course. A lecturer will typically use their knowledge and experience of different groups of learners to incorporate different types of material and activities. During tutorials, lecturers will intuitively pick up informal behavioural cues from students which indicate their level of understanding and their preferred learning style, and use these observations to adapt their teaching style accordingly.

Felder and Silverman described typical learner behaviours and associated teaching styles for each learning style in their model. This information is useful when informally grouping types of learners and also when building the ILS model into a CITS. A summary of the behaviour descriptions is given below:

- *Perception.* Sensing learners prefer facts and experimentation, are patient with detail, comfortable with symbols (e.g. words) and careful but slow. Intuitive learners prefer principles and theories, are bored by detail, uncomfortable with symbols and quick but careless.
- *Input.* Visual learners remember what they see, like pictures and diagrams and prefer visual demonstration. Verbal learners remember what they hear, like discussion and prefer verbal explanation.
- *Processing.* Active learners like to do something with information (discuss or test), they are experimentalists and process information by testing an idea. Reflective learners like to examine and manipulate information internally, are theoreticians and process information by postulating explanations and drawing analogies.
- *Understanding.* Sequential learners like to follow a linear reasoning process, can work with partially understood material and prefer information presented in a steady progression of complexity. Global learners make intuitive leaps, have difficulty working with material they have not understood and prefer to jump directly to complex material.

The Oscar CITS was designed to imitate a human tutor-led tutorial rather than being developed specifically to predict learning styles. Therefore Oscar requires knowledge of the theory of learning styles and their associated behaviours rather than the diagnostic questionnaire in order to imitate the practice of a human tutor.

2.2 Intelligent Tutoring Systems

Intelligent Tutoring Systems (ITS) are computer-based educational systems which employ intelligent technologies to provide individualised instruction. There are three main approaches to intelligent tutoring: curriculum sequencing, intelligent solution analysis and problem solving support [1]. Curriculum sequencing systems present students with learning material in a sequence and style best suited to their needs [2]. Intelligent solution analysis gives detailed feedback to the student on incomplete or erroneous solutions [8], and problem solving support techniques present intelligent assistance to reach a solution [9]. Curriculum sequencing alone is little better than selecting chapters from a book, but by including intelligent solution analysis and problem solving support an ITS can get close to offering support available from a human tutor. Although combining these three technologies adds benefits such as a more effective learning experience and improved student confidence and motivation, few ITS incorporate all three approaches as they are complex and time-consuming to develop. The Oscar CITS presented in this paper will include all three intelligent technologies by personalising learning material and conversing with the student, helping them to construct knowledge and learn from their mistakes.

Personalisation in ITS was traditionally based on student knowledge but has now been extended to include user affect, such as emotion [10], personality [11] and learning style [12]. Some ITS capture learning styles using a formal questionnaire [5], [13] whilst others analyse a student's behaviour within the ITS [6], [14]. Completing questionnaires is onerous for students, who do not always lend enough attention to complete them accurately. Predicting learning style using a history of student behaviour means adaptation is delayed until several modules have been completed, and also a student's learning style may change over time or for different topics. EDUCE [15] and WELSA [16] both estimate learning style dynamically for curriculum sequencing, but do not include a conversational interface or incorporate other intelligent tutoring technologies. The Oscar CITS will dynamically predict learning style throughout the tutoring conversation and adapt its intelligent tutoring style to suit the learning style.

2.3 Conversational Agents

Conversational agents (CAs) allow people to interact with computer systems using natural language dialogues. CA interfaces are intuitive to use and have been engaged effectively in many applications, such as web-based guidance [17], database interfaces [18] and intelligent tutoring systems [19]. Most ITS do not have a natural language interface as CAs are complex and time-consuming to develop, however to adequately mimic a human tutor an ITS should support the construction of knowledge through discussion [20]. The complexity of developing conversational tutors means that CAs are often included in ITS to help with the learning management system (e.g. how to use the system) [21] rather than conduct the tutoring. Two ITS with CA tutors are AutoTutor [3], which helps students construct knowledge about computer literacy and physics, and CIRCSIM-tutor [22], which engages students in discussion to solve physiology problems. Unlike the Oscar CITS, neither AutoTutor or CIRCSIM-tutor consider learning styles during tutoring.

There are three main approaches to developing CAs: using natural language processing [23], pattern matching [24] or artificial intelligence [25] methods. The Oscar CITS adopts a pattern matching CA, which is most reliable in coping with student utterances including grammatically incorrect or incomplete language (as commonly found in student communications such as chat programs and SMS text messaging). Pattern matching CA systems use an algorithm to match key words and phrases within a user utterance to a set of pattern-based rules. A rule normally consists of an identification, a set of stimulus patterns, the rule's current status and a response pattern. The algorithm decides the best fitting rule to fire, thus producing the CA response. There are usually numerous patterns in a given context, leading to many hundreds of rules in the CA's knowledge base, which demonstrates the complexity and time required to script rules for a CA (and the reason CAs are rare in ITS).

3 Oscar Conversational Intelligent Tutoring System

The Oscar CITS is a conversational intelligent tutoring system designed to dynamically predict a student's learning style during a tutoring conversation, and to adapt the tutoring style to suit the individual learner. Oscar's pedagogical aim is to provide the learner with the most appropriate learning material for their learning style leading to a more effective learning experience and a deeper understanding of the topic. Rather than being designed with the purpose of picking up learning styles (such as [6]), the Oscar CITS attempts to mimic a human tutor by leading a two-way discussion and using cues from the student dialogue and behaviour to predict and adapt to their learning style. Oscar's natural dialogue interface and classroom tutorial style are intuitive to learners, enabling them to draw on experience to feel more comfortable and confident in using the CITS. Oscar CITS is a personal tutor which can answer questions, provide hints and assistance using natural dialogue, and which favours learning material to suit each individual's learning style. The Oscar CITS offers 24-hour personalised learning support at a fixed cost. Oscar's intelligent approach includes presenting learning material in the sequence and style most suited to the individual's learning style (curriculum sequencing), analysing and giving feedback on incomplete and erroneous solutions (intelligent solution analysis) and giving intelligent hints and discussing questions (problem solving support). By combining all three intelligent technologies with a conversational interface, Oscar's intelligent support aims to build the confidence of the learner and improve motivation and deep understanding of the subject.

3.1 Oscar CITS Architecture

Fig. 1 shows the modular structure of the Oscar CITS, which has been designed with component reuse in mind. This structure allows alternative knowledge bases and conversational agent scripts to be simply 'plugged in' to the system to adapt the tutoring to new subjects.

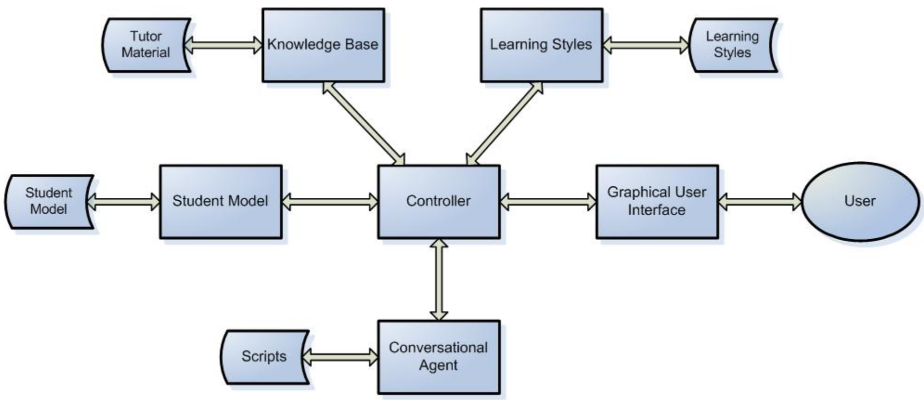


Fig. 1. Oscar CITS structure

The central *controller* manages communication between all components and the user interaction. The *graphical user interface* (GUI) displays a webpage which provides instructions, displays questionnaires, tests, images, documents, interactive movies and the chat area used to send communication to and from the user. The *CA* receives natural language text and information about the topic and learning style and generates a natural language response using a database of scripts. The *student model* holds information about the student, such as their identifier and password, level of knowledge, topics visited, test scores and learning style. The *knowledge base* component manages course information, such as syllabus, related tests and categorised teaching material, which is accessed from a tutor material database. Teaching material is categorised according to teaching style, which is related to learning style. Finally, the *learning styles* component receives information from the CA, GUI, knowledge base and student model, and accesses the learning styles database to predict a student's learning style. Further details of the development of the Oscar CITS can be found in [26].

4 Experimental Methodology

An initial study was conducted to investigate the Oscar CITS prediction of learning style. For the study, the Oscar CITS was scripted to deliver a revision tutorial for an undergraduate Sequential Query Language (SQL) course. There were 17 hypotheses to be tested, covering the learner's behaviour and use of language during the tutorial.

The results covering three hypotheses for the perception (Sensor/Intuitor) and input (Visual/Verbal) ILS dimensions are reported in [26]. These experiments considered the student learning path (accuracy of 70% for Sensor/Intuitor, 50% for Visual/Verbal), the number of interactions (accuracy of 70% for Visual/Verbal) and reading time (accuracy of 70% for Visual/Verbal).

This paper will focus on two hypotheses (H) which relate to the processing (Active/Reflective) and understanding (Sequential/Global) ILS learning dimensions:

H1: a student's learning path through the tutorial is indicative of learning style.
H2: choosing to be guided through a process (or not) is indicative of learning style.

Twenty people were chosen whose first language was English and who had previous experience of an undergraduate SQL course and various levels of SQL expertise. Each person registered anonymously for the Oscar CITS and was then asked to complete the formal ILS questionnaire followed by a multiple choice test to assess existing SQL knowledge. Next, each person engaged in a personalised tutoring conversation led by Oscar. During the tutoring, each learner answered questions and completed various tasks in SQL. Depending on their level of knowledge, students were exposed to various resources and given hints and help as required. At the end of the tutorial, each person was asked to complete the same multiple choice test to measure their learning over the session. Various log files recorded the dialogue, timings, knowledge and other factors throughout the tutorial.

The tutorial log files were analysed and compared to the results of the ILS questionnaire to establish whether the information was indicative of learning style. For H1, aspects of the student's behaviour had been mapped to the ILS model (see Table 1) and learning style scores were incremented depending on the student's interaction during the tutorial. The final scores for each end of the ILS dimension were compared to give the overall learning style tendency for that dimension, e.g. for the processing dimension if the score for Active was higher than the score for Reflective, the learner was considered to be Active. For H2, the learner's approach to writing queries was considered. Two questions involved writing complex SQL

Table 1. Aspects of ILS behaviour built into the Oscar CITS

Active	**Reflective**
Do something with information: discuss/ explain/ test.	Examine and manipulate information introspectively.
Experimentalists.	Theoreticians.
Process information by setting up an experiment to test an idea, or try out on a colleague.	Process information by postulating explanations/ interpretations, drawing analogies, formulating models.
Sequential	**Global**
Follow linear reasoning processes.	Make intuitive leaps.
Can work with material they have only partially or superficially understood.	Difficulty working with material not understood.
Learn best when information is presented in a steady progression of complexity and difficulty.	Sometimes better to jump directly to more complex and difficult material.

queries to solve a problem, and learners who attempted the query straight away were classed as Global learners whilst students who asked for guidance were classed as Sequential learners. Section 5 presents and discusses the results of the experiments.

4.1 Example Learner Dialogue

Table 2 shows part of a student's tutoring dialogue logged during this experiment. In utterance 33, Oscar continues a question about datatypes. The student gives the wrong datatype in utterance 34, and next Oscar indicates that the answer is wrong and gives the student a second chance to answer in case of spelling errors, repeating the question. In utterance 36 the student tries again, still with the wrong answer. Oscar gives feedback that the answer is wrong (utterance 37), and helps by showing an image of the most common datatypes to see if the student is able to pick out the required answer. In this action, Oscar is helping the student learn by reminding them of the common datatypes and asking them to think about the correct datatype rather

Table 2. Example tutoring dialogue snippet logged during the experiment

Utterance	**Dialogue**
33. Oscar>	Now, in our books table suppose we want to store the price of the book where the maximum price of a book is 99.99. What would be the datatype?
34. Student>	VARCHAR2
35. Oscar>	I'm sorry, that's wrong. Check your spelling and try again - type in the datatype for book price (with a maximum of 99.99).
36. Student>	VARCHAR2 MAX 99.99
37. Oscar>	That's still wrong. Take a look at the table of common datatypes - can you now type in the datatype for book-price where the maximum price is 99.99?
38. Student>	NUMBER(2,2)
39. Oscar>	That's correct - we need a datatype of NUMBER(2,2) in order to show the decimal point in book-price, with a maximum of 99.99.

than simply telling them the answer. In utterance 38 the student enters the correct datatype, extended to include the decimal places, and Oscar then confirms that they have the right answer and repeats the solution to reinforce the knowledge.

5 Results and Discussion

Table 3 shows the results of the experiments. The ILS questionnaire results showed that the split of learning styles for each dimension across the sample was nearly equal, with 9 Active and 11 Reflective learners and 10 Sequential and 10 Global learners.

Table 3. Experimental results

Hypothesis	Learning Style	Prediction Accuracy
H1 – learning path	Active	100%
	Reflective	0%
	Sequential	80%
	Global	40%
H2 – approach to queries	Sequential	80%
	Global	75%

For H1, the prediction of learning style by Oscar CITS depended on the learner's path through the tutorial. When compared to the ILS questionnaire results, Oscar accurately predicted an Active learning style in 100% of cases, however it was not possible to predict a Reflective learning style. The characteristics of reflective learners described in the ILS model suggest that they spend time after learning to reflect on what they know and put it together as knowledge. As this activity happens after learning, it may not be possible to predict a reflective learning style during a tutorial. However, these results are not intended to be taken in isolation, and the development of an algorithm to combine different analyses may improve accuracy. Sequential learners were predicted with an accuracy of 80%, however Oscar was not able to predict Global learners using this method, with an accuracy of only 40%.

H2 relates to the Sequential/Global learning style dimension, as it considers a student's approach to writing complex queries. The results of this measure were better than H1, with Oscar's prediction of Global learners the same at 80% accuracy, but of Sequential learners much improved at 75% accuracy. Overall, user feedback after completing the tutorial indicated that Oscar was well received, understandable and helpful. Of the 18 students who did not achieve full marks in the pre-test, all of them improved their test scores, with an average improvement of 25%.

6 Conclusions and Further Work

This paper has presented Oscar, a novel CITS which implicitly predicts and adapts to a student's learning style during a tutoring conversation. Oscar CITS imitates a human tutor by leading a tutorial in natural language, intelligently analysing solutions and offering problem solving support rather than simply presenting the answers. In providing the learner with the most appropriate learning material for their learning style, Oscar CITS aims to improve the effectiveness of the learning experience and provoke a deeper understanding of the topic, and thus improve confidence. An effective, personalised online tutor such as Oscar CITS could support class-based courses and help to widen access through distance learning.

The results of the initial study are promising, with an accuracy of predicting learning style on the Sequential/Global dimension of 75-80%. The Oscar CITS performance on the Active/Reflective dimension was interesting, with a 100% accuracy in predicting Active learners, but no ability to predict Reflective learners at all. When considering the ILS description of reflective learners this is perhaps not surprising – reflective learners prefer to examine and manipulate information introspectively, behaviour that would be most difficult to capture during a tutorial over a short period of time. It may therefore be concluded from this experiment that it is not possible to predict reflective learners in the Oscar CITS, however a further study with a larger sample size is required before drawing any firm conclusions. Further experiments with a larger group are currently being undertaken, and an algorithm is being developed to combine several of the 17 aspects of behaviour to improve the accuracy of learning style prediction. Overall, the results have shown that the Oscar CITS tutoring seemed to help learning as all students who did not initially achieve full marks improved their test scores by an average of 25%.

Acknowledgement. The authors thank Convagent Ltd for the use of the InfoChat conversational agent and PatternScript scripting language.

References

[1] Brusilovsky, P., Peylo, C.: Adaptive and Intelligent Web-based Educational Systems. Int. J. Artificial Intelligence in Education 13, 156–169 (2003)

[2] Weber, G., Brusilovsky, P.: ELM-ART: An adaptive versatile system for Web-based instruction Int. J. Artificial Intelligence in Education 12(4), 351–384 (2001)

[3] Graesser, A., Chipman, P., Haynes, B.C., Olney, A.: AutoTutor: An Intelligent Tutoring System With Mixed-Initiative Dialogue. IEEE Trans. Education 48(4), 612–618 (2005)

[4] Honey, P., Mumford, A.: The Manual of Learning Styles. Peter Honey, Maidenhead (1992)

[5] Wang, H.C., Li, T.Y., Chang, C.Y.: A web-based tutoring system with styles-matching strategy for spatial geometric transformation. Interacting with Computers 18, 331–355 (2006)

[6] Cha, H.J., Kim, Y.S., Park, S.H., Yoon, T.B., Jung, Y.M., Lee, J.H.: Learning styles diagnosis based on user interface behaviours for the customization of learning interfaces in an intelligent tutoring system. In: Ikeda, M., Ashley, K.D., Chan, T.-W. (eds.) ITS 2006. LNCS, vol. 4053, pp. 513–524. Springer, Heidelberg (2006)

[7] Felder, R., Silverman, L.K.: Learning and Teaching Styles in Engineering Education. J. Engineering Education 78(7), 674–681 (1988)

[8] Mitrovic, A.: An Intelligent SQL Tutor on the Web. Int. J. Artificial Intelligence in Education 13, 171–195 (2003)

[9] Melis, E., Andrès, E., Büdenbender, J., Frishauf, A., Goguadse, G., Libbrecht, P., Pollet, M., Ullrich, C.: ActiveMath: A web-based learning environment. Int. J. Artificial Intelligence in Education 12(4), 385–407 (2001)
[10] Ammar, M.B., Neji, M., Alimi, A.M., Gouarderes, G.: The Affective Tutoring System. Expert Systems with Applications 37, 3013–3023 (2010)
[11] Leontidis, M., Halatsis, C.: Integrating Learning Styles and Personality Traits into an Affective Model to Support Learner's Learning. In: Spaniol, M., et al. (eds.) Advances in Web Based Learning – ICWL 2009. LNCS, vol. 5686, pp. 225–234. Springer, Heidelberg (2009)
[12] Papanikolaou, K.A., Grigoriadou, M., Kornilakis, H., Magoulas, G.D.: Personalizing the Interaction in a Web-based Educational Hypermedia System: the case of INSPIRE. User Modeling and User-Adapted Interaction 13, 213–267 (2003)
[13] Spallek, H.: Adaptive Hypermedia: A New Paradigm for Educational Software. Advances in Dental Research 17(1), 38–42 (2003)
[14] Garcia, P., Amandi, A., Schiaffino, S., Campo, M.: Evaluating Bayesian networks' precision for detecting students' learning styles. Computers & Education 49, 794–808 (2007)
[15] Kelly, D., Tangney, B.: Adapting to intelligence profile in an adaptive educational system. Interacting with Computers 18, 385–409 (2006)
[16] Popescu, E.: An Artificial Intelligence Course Used to Investigate Students' Learning Style. In: Li, F., Zhao, J., Shih, T.K., Lau, R., Li, Q., McLeod, D. (eds.) ICWL 2008. LNCS, vol. 5145, pp. 122–131. Springer, Heidelberg (2008)
[17] Latham, A., Crockett, K., Bandar, Z.: A Conversational Expert System Supporting Bullying and Harassment Policies. In: Proc. ICAART 2010, pp. 163–168 (2010)
[18] Owda, M., Bandar, Z., Crockett, K.: Conversation-Based Natural Language Interface to Relational Databases. In: Proc. WI/IAT 2007, pp. 363–367 (2007)
[19] Litman, D.J., Silliman, S.: ITSPOKE: An intelligent tutoring spoken dialogue system. In: Proc. HLT/NAACL 2004, pp. 52–54 (2004)
[20] Chi, M.T.H., Siler, S., Jeong, H., Yamauchi, T., Hausmann, R.G.: Learning from human tutoring. Cognitive Science 25, 471–533 (2001)
[21] De Carolis, B., Pizzutilo, S., Cozzolongo, G., Drozda, P., Muci, F.: Supporting Students with a Personal Advisor. Educational Technology & Society 9(4), 27–41 (2006)
[22] Woo Woo, C., Evens, M.W., Freedman, R., Glass, M., Seop Shim, L., Zhang, Y., Zhou, Y., Michael, J.: An intelligent tutoring system that generates a natural language dialogue using dynamic multi-level planning. Artificial Intelligence in Medicine 38, 25–46 (2006)
[23] Khoury, R., Karray, F., Kamel, M.S.: Keyword extraction rules based on a part-of-speech hierarchy. Int. J. Advanced Media and Communication 2(2), 138–153 (2008)
[24] Michie, D.: Return of the Imitation Game. Electronic Transactions on Artificial Intelligence 6, 203–221 (2001)
[25] Li, Y., Bandar, Z., McLean, D., O'Shea, J.: A Method for Measuring Sentence Similarity and its Application to Conversational Agents. In: Proc. FLAIRS 2004, pp. 820–825 (2004)
[26] Latham, A.M., Crockett, K.A., McLean, D.A., Edmonds, B., O'Shea, K.: Oscar: An Intelligent Conversational Agent Tutor to Estimate Learning Styles. In: Proc. IEEE World Congress On Computational Intelligence 2010, pp. 2533–2540 (2010)

Oscar: An Intelligent Conversational Agent Tutor to Estimate Learning Styles

Annabel M. Latham[1], *Member, IEEE,* Keeley A. Crockett[1], *Member, IEEE,* David A. McLean[1], Bruce Edmonds[2] and Karen O'Shea[1]

Abstract—**Intelligent tutoring systems are computer learning systems which personalise their learning content for an individual, based on learner characteristics such as existing knowledge. A recent extension to ITS is to capture student learning styles using a questionnaire and adapt subject content accordingly, however students do not always take the time to complete questionnaires carefully. This paper describes Oscar, a conversational intelligent tutoring system (CITS) which utilises a conversational agent to conduct the tutoring. The CITS aims to mimic a human tutor by dynamically estimating and adapting to a student's learning style during a tutoring conversation. Oscar also offers intelligent solution analysis and problem support for learners. By implicitly modelling the student's learning style during tutoring, Oscar can personalise tutoring to each individual learner to improve the effectiveness of the tutoring. The paper presents the novel methodology and architecture for constructing a CITS. An initial pilot study has been conducted in the domain of tutoring of undergraduate Science and Engineering students using the Index of Learning Styles ILS) model. The experiments to investigate the estimation of learning style have produced encouraging results in the estimation of learning style through a tutoring conversation.**

I. Introduction

INTELLIGENT Tutoring Systems (ITS) are computerised learning systems which attempt to imitate human tutors to provide more personalised learning than previous content delivery systems [1]. If human tutors could be mimicked adequately, the effectiveness of online learning would be improved and access to learning widened. The availability of an effective computer tutor would have a positive impact on distance learning as well as offering support for traditional class-based courses. Students attending an online tutoring session are able to learn at their own pace and at a time suited to other commitments. Students could also benefit from personalised learning, with the ability to revisit and delve further into topics they have not fully understood, which cannot be offered in a class of many students. For education establishments, online tutorials are a cost-effective way of offering flexible courses, with the cost fixed and borne at the time of development regardless of the number of students.

ITS are generally designed with a menu-style user interface [2], but a conversational interface would be a more natural mimic of human tutoring, offering constructivist styles of learning as used by human tutors [3]. Only a small number of ITS allow discussion with the tutor [4] due to the time and complexity of development. Like human tutors, ITS adapt the tutorial content for each individual student. Adaptation is normally based on a student's level of knowledge, but a recent enhancement is to present content suitable to a student's learning style [5], [6]. Learning styles describe the way in which groups of people learn most effectively, and are normally assessed by questionnaire [7]. Human tutors often informally pick up cues from students which indicate their understanding of a topic, and adapt the tutoring to aid learning, for example by drawing a diagram or giving a practical example. By assessing the student's reaction to particular styles of tutoring, human tutors then favour the more successful styles in future tutorials. ITS adaptation to learning style normally requires the student to complete a formal questionnaire [5], however students do not always take the time to answer questionnaires accurately, leading to incorrect results and less effective learning. Some ITS model learning style based on historical learning behaviour, however adaptation to learning style cannot then be offered initially. If an ITS could learn and adapt to a student's learning style during a tutoring conversation, such personalised, conversational tutoring would improve the student's learning experience. The novel conversational ITS described in this paper aims to mimic a human tutor by learning and adapting to a student's learning style during the tutoring conversation.

Conversational agents (CAs) are computer programs which interact with users by natural language [8]. There are three main approaches to developing CAs – using natural language processing (NLP) [9], pattern matching [10] or artificial intelligence (AI) techniques [11], which will be outlined in section II. The Oscar CITS presented in this paper adopts the pattern matching approach, which may be more reliable as patterns can cope with grammatically incorrect user utterances [11], as often used by students. Conversational agents require scripting for particular domains, a time-consuming and complex task, however to replicate human tutoring, a conversational interface is important.

This paper describes Oscar, a novel CITS which estimates

Manuscript received February 4, 2010.

[1]A. M. Latham, K. A. Crockett, D. A. McLean and Karen O'Shea are with the Intelligent Systems Group, Department of Computing & Mathematics, Manchester Metropolitan University, Manchester M1 5GD, UK (phone: (44) 0161-247-2000; fax: (44) 0161-247-1483; e-mail: A.Latham@ mmu.ac.uk).

[2]B. Edmonds is with the Centre for Policy Modelling, Manchester Metropolitan University, Aytoun Building, Aytoun Street, Manchester M1 3GH, UK.

student learning styles by picking up cues from students during tutoring conversations. By learning and adapting to a student's learning style during a tutoring conversation, Oscar can intelligently personalise tutoring at an early stage and without additional burden on the learner. Oscar is a web-based CITS with a CA interface which leads the tutoring session, asking questions, showing visuals and movies and offering intelligent feedback to students. A novel architecture has been designed which will facilitate the development of a CITS in any domain. For the purpose of this paper Oscar has initially been developed to offer online SQL revision tutorials. The results of the initial pilot study are presented which analysed student interaction with Oscar during an SQL revision tutorial to assess the accuracy of learning style predictions.

This paper is organised as follows: Section II will describe conversational agents, Section III introduces learning styles, Section IV will outline ITS, Section V introduces the Oscar CITS, Section VI describes the experimental methodology of the pilot study and Sections VII and VIII include the results, discussions and conclusions.

II. Conversational Agents

Conversational agents allow people to interact with computer systems using natural language dialogues. There are three main approaches to developing CAs. The natural language processing approach [9] seeks to understand the user input by studying the constructs and meaning of natural language, applying rules to process important parts of sentences. Pattern matching systems [10] use an algorithm to match key words and phrases within an utterance, and so do not require grammatically correct or complete input. The AI method [11] compares the semantic similarity of phrases to decide on the meaning of the input.

CAs usually rely on a knowledge base containing a set of rules. User utterances are matched to pattern-based rules in the knowledge base and an algorithm decides which is the best fitting rule to fire, producing the CA response. A rule normally consists of an identification, a set of stimulus patterns, the rule's current status and a response pattern [11]. The Oscar CITS uses a pattern matching CA, which is most reliable in coping with student utterances including grammatically incorrect or incomplete language.

III. Learning Styles

Learning styles describe the way in which groups of people learn most effectively, for example by trial and error, or by observation [12]. There are numerous models of learning styles, which are generally assessed using self-assessment questionnaires. Most models of learning style describe dimensions along which a value is placed to represent the tendency for learning style. For example, on a visual/verbal dimension, learners who are more comfortable with discussion and verbal explanation tend towards the verbal end, whereas learners who prefer to study diagrams and pictures would tend towards the visual end.

Learning styles are thought to be a subset of personality [12] and there is much discussion in the literature about whether and how learning styles can be of use to teachers and learners [13], [14]. Pask concludes "It seems evident that distinctive learning strategies exist. ... There are also certain distinct styles, or dispositions to adopt classes of strategy" [14].

An early learning styles model was Kolb's Experiential Learning Model (ELM) [7], which is a four-stage learning cycle which can be entered at any point. ELM was developed further to produce the *Learning Style Inventory* (LSI), a 12 item questionnaire requiring the ranking of sentence endings [7]. Honey and Mumford's *Learning Style Questionnaire* (LSQ) was developed for management trainees and defines four learning styles which are similar to the stages of learning in ELM [15]. The LSQ model has been used in some ITS [1].

The *Index of Learning Styles* (ILS) [16], [17] was developed to describe the learning styles in engineering education and suggest different pedagogical styles to address learners' needs. The ILS model defines four dimensions of preferred learning style: perception (Sensor/Intuitor), input (Visual/Verbal), processing (Active/Reflective) and understanding (Sequential/Global). The ILS uses a self-assessment questionnaire with 11 questions per learning style dimension, which results in a score for each of the four dimensions. Each learning style dimension represents an axis with the opposite learning styles at each end (e.g. Visual versus Verbal), and the ILS questionnaire score places each learner on the axis according to the strength of their preferred learning style. The ILS has been adopted by a number of ITS [18], [5], [19], [20]. The ILS model was chosen for the Oscar CITS as it was designed for engineering students, who will make up the initial experimental groups. However, Oscar's modular structure means it is not restricted to ILS and can be adapted to use other learning style models.

IV. Intelligent Tutoring Systems

Computer-assisted learning systems were traditionally information-delivery systems developed by converting tutor or distance-learning material into a computerised format. The popularity of the Internet has enhanced the opportunities for e-learning, however most online systems are still teacher-centred and take little account of learner needs [21]. [22] identified two main groups of adaptive and intelligent web-based educational systems - Adaptive Hypermedia Systems (AHS) and Intelligent Tutoring Systems (ITS). AHS are akin to interactive books which adapt the navigation and content of hyperlinks to the knowledge of the user [1], [6]. ITS personalise teaching according to individual student characteristics, such as knowledge of the subject. A student model is built, including personal details and learning history, and teaching is adapted to the student. Such systems are now being extended to adapt to other student information, such as mood and emotion [23], [24] and

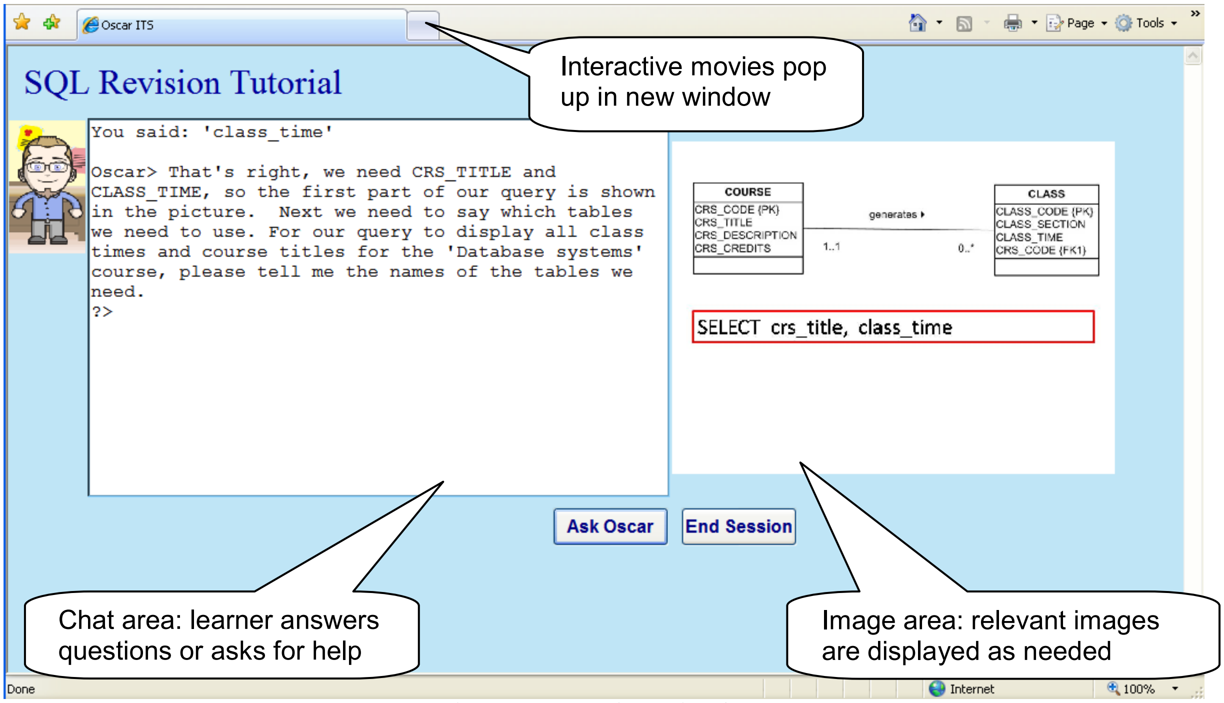


Fig. 1. Oscar CITS learner interface.

learning style [19].

Three approaches to intelligent tutoring are curriculum sequencing, intelligent solution analysis, and problem solving support [22]. Curriculum sequencing involves presenting each student with learning material in a sequence and style best suited to their needs [25]. Intelligent solution analysis aims to provide detailed feedback to the student on incomplete or erroneous solutions [2], and problem solving support techniques offer intelligent help in arriving at a solution [26]. Curriculum sequencing alone is little better than personalising a book, but by incorporating intelligent solution analysis and problem solving support an ITS can get close to mimicking a human tutorial. Few ITS incorporate all three intelligent approaches as they are complex and time-consuming to develop. However, combining all three technologies adds benefits by offering a more effective learning experience and intelligent support which can help to build confidence and motivation. The Oscar CITS presented in this paper will include all three intelligent technologies by personalising the learning material and helping the student to construct knowledge and learn from their mistakes.

A. Conversational ITS

Conversational interfaces have rarely been incorporated into teaching and learning systems, however the benefits of constructivist styles of learning (as used by human tutors) are widely accepted [27]. To mimic a human tutor, ITS should support the construction of knowledge: "*it seems necessary for future generations of ITSs to incorporate natural language capabilities.*" [31]. The complexity of developing conversational tutors means where CAs are included in ITS, it is often to interact or help with the learning management system (e.g. how to use the system) rather than conduct the tutoring [28], [29]. Two conversational ITS which do adopt CA tutors are AutoTutor [4] and CIRCSIM-tutor [30]. AutoTutor allows students to construct knowledge about computer literacy and physics through conversations. CIRCSIM-tutor incorporates a CA to allow students to solve physiology problems by discussion. Neither of these CITS take learning styles into consideration during tutoring.

B. Adaptation to Learning Style

Most ITS personalise learning by adapting to a student's existing knowledge of the subject. The extension of ITS to adapt to other student characteristics, such as learning style, is a new area of research. A small number of ITS which adapt to learning style use formal questionnaires completed by students during registration [5], [21]. However, students may not complete the questionnaire accurately as it is time consuming, therefore producing an unreliable student model [19]. There have been some attempts to detect learning style by analysing a student's behaviour history within the ITS [19], [20], [32], [33]. Whilst removing the need to complete a questionnaire, such ITS are not able to adapt to learning style until a number of learning modules are complete. Estimating learning style dynamically and continually updating the student model allows an ITS to adapt to changes in learning style over time. The EDUCE [34] and WELSA [35] adaptive educational systems both dynamically estimate learning style for curriculum sequencing, however they do not include a conversational interface or incorporate other intelligent tutoring technologies. The Oscar CITS reported in this paper will dynamically estimate learning style during a tutoring conversation, and then adapt the tutoring to suit that learning

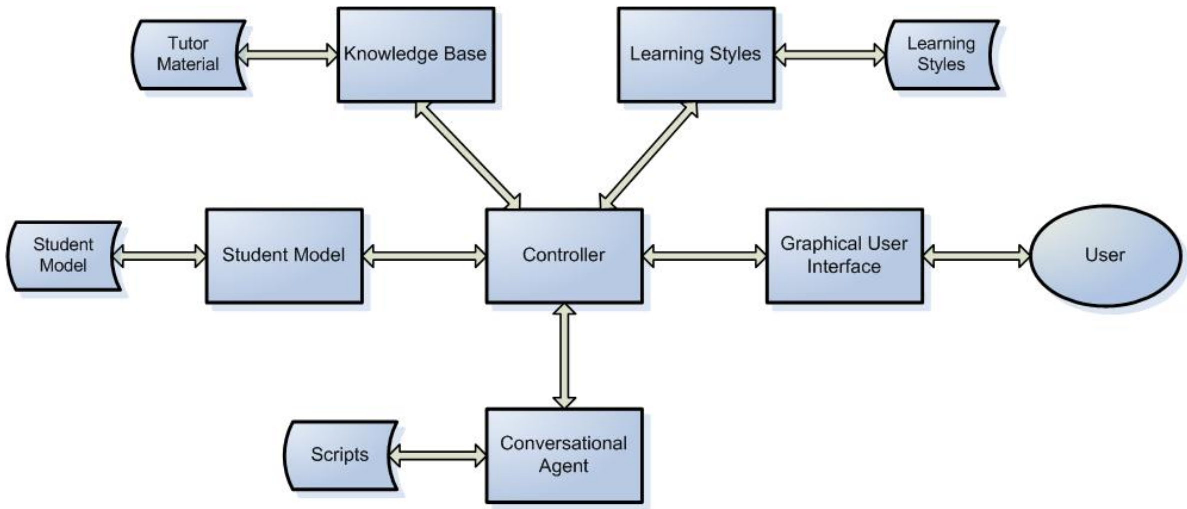


Fig. 2. Oscar CITS structure.

style whilst offering intelligent solution analysis and problem solving support.

V. Oscar CITS

The Oscar CITS is a conversational intelligent tutoring system which can dynamically estimate and adapt to a student's learning style during a tutoring conversation. In addition to curriculum sequencing, Oscar aims to mimic a human tutor in offering intelligent solution analysis and conversational problem solving support in the domain of the database Structured Query Language (SQL). The ILS model was adopted, which describes different learning characteristics and identifies associated pedagogical styles for engineering students. Oscar draws on a knowledge base of tutor material and conversation scripts to deliver a conversational tutorial to a student. To support the tutoring conversation, diagrams, images and interactive movies may be displayed. Aspects of the student's behaviour and understanding inform the dynamic estimation of learning style, allowing the tutoring style to be personalised to best suit the student.

Fig. 1 shows the Oscar CITS graphical user interface (GUI) during a tutoring session, where a diagram is visible and a tutoring conversation is taking place. The student is being asked to write a query with four main parts, and has chosen to be guided through each step by Oscar. The image shows a Unified Modelling Language (UML) diagram of the relevant database tables and the first part of the query written so far. In the chat area, Oscar has responded to confirm that the learner's previous answer was correct and has stated the next step in writing the query. Oscar then reminds the learner of the main query question and asks for information required for the next stage.

A. *Oscar Architecture*

Fig. 2 shows the overall structure of the Oscar CITS. A *central controller* communicates with all components to manage the user interaction. The *knowledge base* manages course information, such as topics and their breakdowns, related tests and teaching material, which is accessed from a Tutor Material database. All tutor information is categorised according to teaching style (related to learning style). The *learning styles* component receives information from the CA, GUI, knowledge base and student model and accesses the Learning Styles database, to estimate a learning style. The *student model* holds information about the student, such as name, level of knowledge, topics visited, test scores and learning style. The *GUI* (Fig. 1) displays a webpage showing questionnaires, tests, images, documents and interactive movies and sends communication to and from the user. The *conversational agent* receives natural language text and information about topic and learning style from the GUI, knowledge base and learning styles components, and generates a natural language response. The CA accesses a database of scripts in order to match the input and generate a response.

B. *Methodology*

Learning styles are central to the Oscar CITS, so development started by considering the ILS model. The ILS questionnaire contains 44 questions – too many to incorporate into a tutoring session, so a pilot study was done of 103 completed ILS questionnaires to investigate which were the best predictor questions [36]. The study found that 17 questions predicted the overall result in at least 75% of cases, with the top three questions predicting the result in 84% of cases. The subset of the best ILS predictor questions for each learning style dimension was then considered during the development of the Oscar CITS.

The domain of SQL was selected as the target audience for the pilot study would be undergraduate computing students, for whom a Databases course including SQL is compulsory. The ILS model, which was designed to describe engineering students' learning styles, is appropriate to this target group. Several interviews with undergraduate level database course tutors were undertaken. In consultation with database course lecturers, several SQL concepts were identified from an undergraduate Databases course syllabus.

Tutoring revision scenarios were designed, based around the syllabus and the database lecturers' experience of revision tutorials. Each revision question was mapped to the ILS model by incorporating questions from the questionnaire and using the model's descriptions of indicative behaviour, such as a preference for theoretical questions. Table I shows two examples of logic rules used by the system to increment learning style values during tutoring. Learning styles are held in eight values within the student model, representing each pole of the four dimensions. The logic rules are incremental, increasing learning style values where particular behaviour is evident. At the start of the first tutoring session, no initial learning style values exist for a student. During the tutoring conversation, learning style values are incremented depending on the student's tutoring conversation. At the end of the tutoring session, the value pairs of each learning style dimension are compared to reveal the student's overall learning style tendency for that dimension (i.e. the greater value). Learning style values depend on an individual's unique tutoring session, and if no evidence is gathered to suggest a particular learning style dimension, that learning style will remain unclassified. For example, student *x* attended a tutoring session on SQL during which their behaviour was analysed to uncover evidence suggesting a particular learning style. At the end of the tutoring discussion, the student's learning style was estimated to be Intuitor and Verbal, but no evidence was found to categorise the student for the remaining two learning style dimensions (Active/Reflective and Sequential/Global). Student *x* next completes a follow-up tutorial session which favours content to match an Intuitor/Verbal learning style. Incremental evidence from both tutoring conversations estimated the student's learning style to be Intuitor/Verbal/Active but there was no evidence to indicate a value for the Sequential/Global dimension.

TABLE I
EXAMPLE OF LOGIC RULES USED TO ADJUST STUDENT LEARNING STYLE VALUES BASED ON TUTORING CONVERSATION

1. Example rule to test whether presenting information visually helps the student's information perception:

```
IF    student does not know the answer
THEN  show student image/diagram;
IF    student shown image/diagram
AND   student gives correct answer
THEN  increase VISUAL;
```

2. Example rule to test how comfortable the student is with words and with detail:

```
IF    answer is given in the explanation text
AND   student does not know the answer
THEN  increase INTUITOR
AND   increase VISUAL;
```

Tutoring conversations were written based on the SQL revision scenarios, including numerous possible student responses. Additional material such as images, diagrams and movies was incorporated into the tutoring conversations and mapped to learning styles. The resulting tutorial walkthroughs indicated which learning style should be incremented at which point, based on the student's learning path.

TABLE II
EXAMPLE OF INFOCHAT PATTERNSCRIPT SCRIPTING SHOWING AN FAQ RULE

```
<Rule-01>
a:0.01
p:50  *<explain-0> *select*
p:50  *select* <explain-0>*
p:50  *<remind-0> *select*
p:50  *select* <remind-0>*
p:50  *<confused-0> *select*
p:50  *select* <confused-0>*
r:    The SQL SELECT command is used to retrieve data from
      one or more database tables.
```

Several styles of question were included, for example practical problems to create queries and theoretical questions to test knowledge. Standard question formats were represented diagrammatically to speed up development by reuse of the logic and CITS scripts. A list of frequently asked questions (FAQs) about SQL was compiled and an existing multiple choice test was adapted to cover the revision syllabus.

Next began the time consuming and complex task of scripting the CA component. Convagent's InfoChat CA [37], a CA employing natural language pattern matching, was chosen. CA scripts, organised into contexts, were developed to manage the tutorial conversation and respond to student inputs. Overall, there were 38 contexts containing around 400 rules which demonstrates the complexity of developing a CITS. A frequently asked questions (FAQ) layer of scripts was developed to deal with student responses which did not directly relate to the current question. Additionally a lower layer of scripts was designed to pick up abusive language (sessions are ended at this point). An example FAQ rule from one of the InfoChat scripts is shown in Table II. In the rule, *a* is the activation level used for conflict resolution [38]; *p* is the pattern strength followed by the pattern and *r* is the response. Also seen in the example is the wildcard (*) and macros (<explain-0>) containing a number of standard patterns which are each matched separately. Further information about the PatternScript language and InfoChat algorithm is available by contacting http://www.convagent.com.

The student model was designed, which holds the student name and password, level of knowledge, test scores and learning style values. The Oscar CITS components were then developed, producing a framework system which draws on various resources (the conversational agent scripts, tutor material, student model and learning styles) to present an adaptive tutoring session. The CITS student registration includes the completion of the ILS questionnaire and the completion of a multiple choice question (MCQ) pre-test to assess existing student knowledge. The same MCQ test is presented at the end of the tutoring session in order to assess learning.

During tutoring, the CITS records and logs information about the behaviour of the student, such as timing of interactions, the number of words used, the number of times FAQs are asked and the type of tutor resource accessed. The tutoring conversation is also recorded, along with information about the student knowledge of the topic being discussed.

VI. Experimental Methodology

An initial pilot study was conducted to assess the Oscar CITS in two ways – firstly Oscar's estimation of learning style and secondly the acceptance of the Oscar tutor by users. Three experiments were conducted, focusing on the perception (Sensor/Intuitor) and input (Visual/Verbal) ILS learning style dimensions. Experiment 1 explored the student's path through the learning material, Experiment 2 examined the number of discourse interactions during tutoring and Experiment 3 investigated reading time.

TABLE III
EXPERIMENTAL RESULTS

	Learning Style	Comparison	Accuracy of Estimation
Experiment 1 – learning path			
	SNS/INT[a]	-	70%
	VIS/VRB[b]	-	50%
Experiment 2 – number of interactions			
	VIS/VRB	Mean	70%
	VIS/VRB	Median	70%
Experiment 3 – reading time			
	VIS/VRB	Mean	60%
	VIS/VRB	Median	70%

[a]SNS/INT is the Sensor/Intuitor dimension
[b]VIS/VRB is the Visual/Verbal dimension

Ten people were chosen whose first language was English and who had previous experience of an undergraduate ORACLE SQL course (but with various levels of expertise). Each person registered for the Oscar CITS, completing the ILS questionnaire and pre-test, and then went through the SQL revision tutorial. Finally each person completed the post-test. At the end of the tutoring session, each person was informally interviewed and asked to complete a feedback questionnaire.

The log files recorded by the Oscar CITS for each person were analysed and compared to the results of the formal ILS questionnaire to assess whether the information being collected could be used to indicate learning style, and whether Oscar had accurately estimated learning style.

For Experiment 1, depending on the student's answers to tutoring questions, learning styles were incremented according to the mappings made to the ILS model which were documented in the tutoring conversation walkthrough. The final learning style scores were then converted into an overall learning style for each dimension, e.g. for the VIS/VRB (Visual/Verbal) dimension if the score for Visual was higher than that for Verbal, the student was considered to be Visual. The learning style result was compared to the ILS questionnaire results for each student.

For Experiment 2, the number of discourse interactions during the tutoring session was counted and compared to the mean and median values across the sample group. The hypothesis was that the more discursive a student is (i.e. the more interactions), the more they tend towards the verbal learning style.

For Experiment 3, the mean time taken to read 10 Oscar words was calculated for each student and compared to the mean and median values across the sample group. The hypothesis was that the longer a student takes to read instructions (i.e. the less comfortable the student is with words), the more they tend towards the visual learning style.

The next section presents and discusses the results of these experiments.

VII. Results and Discussion

Table III summarises the results of the three experiments. It should be noted that, as expected, the split of learning styles assessed by the ILS questionnaire was not equal across the sample. For the SNS/INT (Sensor/Intuitor) dimension, 20% of the sample was Sensory and 80% Intuitive learners. For the VIS/VRB dimension, 80% of the group was Visual and 20% was Verbal learners. Each experiment will now be discussed separately, and then the learner feedback on Oscar CITS will be summarised.

A. *Experiment 1 - Learning Path*

In experiment 1, the estimation of learning style depended on the learner's path through the tutoring material. For the perception dimension (Sensor/Intuitor), Oscar's result agreed with the ILS questionnaire result in 70% of cases. For the input dimension (Visual/Verbal), Oscar's results agreed with the ILS questionnaire in only 50% of cases. Clearly, further work and consideration needs to be given to the effect of visual material (images) versus discussion and explanation on the learner's understanding. As the tutorial is tutor-led rather than student-led, this dimension may be more difficult to estimate by conversation than in, for example, a hyperlink system [35]. However, the results of each experiment are not intended to be taken in isolation, and the development of an algorithm to combine different types of analysis may offer better accuracy.

B. *Experiment 2 – Number of Interactions*

Experiment 2 relates to the input dimension (Visual/Verbal) with the hypothesis that the students who enter into most discussion with Oscar are Verbal learners. The students were categorised as Visual or Verbal learners by comparing the number of discourse interactions to the mean and the median for the sample, and this was compared to the ILS questionnaire result. In 70% of cases for both the mean and median comparisons, there was agreement in the learning style assessment.

C. *Experiment 3 – Reading Time*

The hypothesis for Experiment 3 was that Visual learners take longer to read than Verbal learners. Students were categorised as Visual or Verbal learners by comparing their mean reading time for 10 Oscar words over the whole tutoring session with the group mean and median. Compared to the group mean, Oscar agreed with the ILS in 60% of cases, rising to 70% of cases when compared to the sample median. The mean differed considerably from the median, by 2 seconds, as the duration of the tutoring session also differed substantially, by 37 minutes, 7 seconds. As each individual's learning path is different, different numbers of Oscar words will be presented, however the indication is that the median is the most appropriate measure for comparison in this case.

D. *Using Oscar CITS*

In general, the user feedback from the initial pilot study showed that Oscar was well received, understandable and helpful. All students showed an improvement in their test scores after the revision tutorial, with the average improvement across the sample of 21%. 90% of the group would use Oscar to support classroom tutoring, with a surprising 20% stating they would use Oscar *instead* of face-to-face tutoring. Only 40% of learners agreed that they would use the Oscar CITS instead of reading a book. When openly asked for comments, half of the group commented that the conversational interface was natural and easy to understand, with one learner remarking "*it encouraged me to think rather than simply giving me the answer".*

VIII. CONCLUSION

This paper has presented the novel architecture and methodology for developing Oscar, a CITS which implicitly estimates and adapts to a student's learning style. Oscar employs a CA to intelligently lead an online tutorial, mimicking a human tutor in offering students individualised problem solving support and intelligent solution analysis. A CITS which personalises tutoring by dynamically estimating and adapting to learning style could improve the effectiveness of a student's learning experience and help to boost confidence. Effective, personalised online tutoring could offer support for class-based courses and widen access with distance learning.

The results of the initial pilot study are promising, with an accuracy of estimating learning style of 70% in three cases but 50% in the worst case. It is not appropriate to draw firm conclusions with a small initial sample size, and an unequal spread of learning style. Further experiments with a larger group are currently being undertaken. In addition, an algorithm using a fuzzy set representation of learning styles is currently being developed to combine different aspects of behaviour to improve the accuracy of learning style estimation. With regards to Oscar's conversational tutoring, the results have shown that the subjects did value the online Oscar CITS in supporting classroom lessons, and that Oscar's tutoring seemed to help learning and improved test scores in every case. It can therefore be concluded that using Oscar has helped give students a positive learning experience.

ACKNOWLEDGMENT

The authors thank Convagent Ltd for the use of the InfoChat conversational agent and PatternScript scripting language.

REFERENCES

[1] K. A. Papanikolaou, M. Grigoriadou, H. Kornilakis and G. D. Magoulas, "Personalizing the Interaction in a Web-based Educational Hypermedia System: the case of INSPIRE", *User Modeling and User-Adapted Interaction*, vol. 13, pp. 213-267, 2003.

[2] A. Mitrovic, "An Intelligent SQL Tutor on the Web", *International Journal of Artificial Intelligence in Education*, vol. 13, pp. 171-195, 2003.

[3] M. Sharples, "Learning As Conversation: Transforming Education in the Mobile Age", in *Proc. Conf. on Seeing, Understanding, Learning in the Mobile Age*, Budapest, Hungary, April 2005.

[4] A. Graesser, P. Chipman, B. C. Haynes and A. Olney, "AutoTutor: An Intelligent Tutoring System With Mixed-Initiative Dialogue", *IEEE Trans. Education*, vol. 48 (4), pp. 612-618, 2005.

[5] H. C. Wang, T. Y. Li and C. Y. Chang, "A web-based tutoring system with styles-matching strategy for spatial geometric transformation", *Interacting with Computers*, vol. 18, pp. 331-355, 2006.

[6] E. Triantafillou, A. Pomportsis, S. Demetriadis and E. Georgiadou, "The value of adaptivity based on cognitive style: an empirical study", *British J. Educational Technology*, vol. 35 (1), pp. 95-106, 2004.

[7] D. A. Kolb, *Learning style inventory: technical manual*, McBer & Company, Boston MA, 1976.

[8] J. Cassell, *Embodied Conversational Agents*, MIT Press, Cambridge MA, 2000.

[9] R. Khoury, F. Karray and M. S. Kamel, "Keyword extraction rules based on a part-of-speech hierarchy", *Int. J. Advanced Media and Communication*, vol. 2 (2), pp. 138-153, 2008.

[10] K. Pudner, K. Crockett and Z. Bandar, "An Intelligent Conversational Agent Approach to Extracting Queries from Natural Language", in *Proc.2007 WCE Int. Conf. Data Mining and Knowledge Engineering*, pp. 305-310.

[11] Y.Li, Z. Bandar, D.McLean and J. O'Shea, "A Method for Measuring Sentence Similarity and its Application to Conversational Agents", in *Proc.17th Int. Florida AI Research Society Conf. (FLAIRS 2004)*, pp. 820-825.

[12] A. Furnham, "Personality and learning style: a study of three instruments", *Personality and Individual Differences*, vol. 13, pp. 429-438, 1992.

[13] F. Coffield, D. Moseley, E. Hall and K. Ecclestone, "Learning styles and pedagogy in post-16 learning: A systematic and critical review", Learning and Research Skills Centre, UK, 2004.

[14] G. Pask, "Learning strategies, teaching strategies, and conceptual or learning style" in *Learning strategies and learning styles,* R.R.Schmeck, Ed. New York: Plenum, 1988, ch.4 pp.83-100.

[15] P. Honey and A. Mumford, *The manual of learning styles*, Peter Honey, Maidenhead, 1992.

[16] R. Felder and L. K. Silverman, "Learning and Teaching Styles in Engineering Education", *J. Engineering Education*, vol. 78 (7), pp. 674-681, 1988.

[17] R. Felder and E. R. Henriques, ER, "Learning and Teaching Styles in Foreign and Second Language Education", *Foreign Language Annals*, vol. 28 (1), pp. 21-31, 1995.

[18] P. Sancho, I. Martinez and F. M. Baltasar, "Sematic Web Technologies Applied to e-Learning Personalization in <e-aula>", *J. Universal Computer Science*, vol. 11 (9), pp 1470-1481, 2005.

[19] V. Yannibelli, D. Godoy and A. Amandi, "A genetic algorithm approach to recognise students' learning styles", *Interactive Learning Environments*, vol. 14 (1), pp 55-78, 2006.

[20] J. E. Villaverde, D. Godoy and A. Amandi, "Learning styles' recognition in e-learning environments with feed-forward neural networks", *J. Computer-Assisted Learning*, vol. 22, pp 197-206, 2006.

[21] H. Spallek, "Adaptive Hypermedia: A New Paradigm for Educational Software", *Advances in Dental Research*, vol. 17 (1), pp. 38-42, 2003.

[22] P. Brusilovsky and C. Peylo, “Adaptive and Intelligent Web-based Educational Systems”, *Int. J. Artificial Intelligence in Education*, vol. 13, pp. 156–169, 2003.

[23] S. K. D'Mello, S. D. Craig, B. Gholson, S. Franklin, R. Picard and A. C. Graesser, “Integrating affect sensors in an intelligent tutoring system”, in *Affective Interactions: The Computer in the Affective Loop Workshop at 2005 Int. Conf. Intelligent User Interfaces,* New York: AMC Press, pp. 7-13.

[24] A. C. Graesser, S. K. D'Mello, S. D. Craig, A. Witherspoon, J. Sullin, B. McDaniel and B. Gholson, “The Relationship Between Affective States and Dialog Patterns During Interactions With AutoTutor”, *J. Interactive Learning Research*, vol. 19 (2), pp. 293-312, 2008.

[25] G. Weber and P. Brusilovsky, “ELM-ART: An adaptive versatile system for Web-based instruction”, *Int. J. Artificial Intelligence in Education,* vol. 12 (4), pp. 351-384, 2001.

[26] E. Melis, E. Andrès, J. Büdenbender, A. Frishauf, G. Goguadse, P. Libbrecht, M. Pollet, and C. Ullrich, “ActiveMath: A web-based learning environment”, *Int. J. Artificial Intelligence in Education*, vol. 12 (4), pp. 385-407, 2001.

[27] M. Sharples, “Learning As Conversation: Transforming Education in the Mobile Age”, *Conf. on Seeing, Understanding, Learning in the Mobile Age*, Budapest, Hungary, April 2005

[28] B. De Carolis, S. Pizzutilo,G. Cozzolongo, P. Drozda and F. Muci, “Supporting Students with a Personal Advisor”, *Educational Technology & Society*, vol. 9 (4), pp. 27-41, 2006.

[29] A. Kerly, R. Ellis and S. Bull, “CALMsystem: A Conversational Agent for Learner Modelling”, in R. Ellis, T. Allen and M. Petridis (eds), *Applications and Innovations in Intelligent Systems XV – Proc. SGAI-2007*, pp. 89-102, 2007.

[30] C. Woo Woo, M. W. Evens, R. Freedman, M. Glass, L. Seop Shim, Y. Zhang, Y. Zhou and J. Michael, “An intelligent tutoring system that generates a natural language dialogue using dynamic multi-level planning”, *Artificial Intelligence in Medicine*, vol. 38, pp. 25-46, 2006.

[31] M. T. H. Chi, S. Siler, H. Jeong, T. Yamauchi and R. G. Hausmann, “Learning from human tutoring”, *Cognitive Science*, vol. 25, pp. 471-533, 2001.

[32] N. Stash and P. De Brau, “Incorporating cognitive styles in AHA! (The adaptive hypermedia architecture)”, in *Proc. 2004 IASTED Int. Conf.*, pp. 378-383, ACTA Press, Calgary, Canada.

[33] P. Garcia, A. Amandi, S. Schiaffino and M. Campo, “Evaluating Bayesian networks’ precision for detecting students’ learning styles”, *Computers & Education*, vol. 49, pp. 794–808, 2007.

[34] D. Kelly and B. Tangney, “Adapting to intelligence profile in an adaptive educational system”, *Interacting with Computers*, vol. 18, pp. 385-409, 2006.

[35] E. Popescu, “An Artificial Intelligence Course Used to Investigate Students’ Learning Style”, in *ICWL 2008, Lecture Notes in Computer Science*, vol. 5145, pp. 122-131, Springer Heidelburg.

[36] A. Latham, K. Crockett, D. McLean and B. Edmonds, “Using Learning Styles to Enhance Computerised Learning Systems”, in *Proc. of the 2009 Annual Research Student Conference, Manchester Metropolitan University*, Manchester, UK.

[37] Convagent Ltd 2005, *Convagent*, Available: http://www.convagent.com/

[38] D. Michie and C. Sammut, *Infochat Scripter’s Manual*, ConvAgent Ltd, Manchester, UK, 2001.