Improving User Modeling via the Integration of Learner Characteristics and Learner Behaviors

Kimiz L. Dalkir

A Thesis

in

The Department

of

Educational Technology

Presented in Partial Fulfillment of the Requirements

for the Degree of Doctor of Philosophy at

Concordia University

Montreal, Quebec, Canada

April, 1997

0-612-25914-5

ABSTRACT

Improving User Modeling via the Integration
of Learner Characteristics and Learner Behaviors

Kimiz L. Dalkir, Ph.D.
Concordia University, 1997

Three major disciplines: educational psychology, cognitive science and artificial intelligence, were critically surveyed to identify useful variables for learner modeling in order to identify the subset of variables that proved to be useful in modeling individual learners as they interacted with a computer-based learning environment. The research study first critically assessed the contextual validity and usefulness of *a priori* measures to provide initial or default values for a stereotypical learner model. These measures included a pretest, a questionnaire and two learning style inventories, the Kolb LSI and the Entwistle ASI. In addition, the utility of Artificial Neural Networks (ANNs) was assessed to establish whether they provide supplementary or complementary information for the objective of creating an adaptive learner model. Differences in interaction patterns with the learning environment were analyzed using ANNs and statistical analyses, to identify on-line learner behavioral variables that were valid and useful in updating the stereotype learner model.

The instructional validity of the learning environment was established as students were found to have spent time interacting with the system, they attended to the material presented and they were found to have learned the content. Significant learning was found, as assessed by pretest-posttest differences. Of the Educational Psychology variables, only the Entwistle ASI proved to be useful as an *a priori* measure in this context. Students with high scores on both the reproducing and meaning orientation dimensions performed better on the posttest. In addition, expected learner profiles, as extrapolated from the ASI, actually occurred as students interacted with the system.

Finally, Artificial Intelligence approaches, in the form of Artificial Neural Networks (ANNs) were superimposed on the *a priori* categorization established by the learning style categories.

Conventional statistical cluster analysis and ANN pattern recognition on learner trace data produced as students interacted with the learning materials both produced very similar classifications of students. It thus appears to be possible to obtain, effectively, the same data from an ongoing dynamic assessment of learners as it is from a priori measures, rendering the latter redundant in this context. Thus the use of ANNs can prove useful as a dynamic data gathering and analysis system in real time to make instructional adjustments and recommendations. The potential advantage of dynamic models over *a priori* measures is that they continue to evolve as learner needs change, continually updating the learner model and thus enabling the learner model to keep pace with an instructional system endowed with adaptive capabilities. Future research could build on the exploratory data generated by this study, examining both the variables which may inform the creation of an adaptive interface, as well as using the ANN-based methodology created here.

ACKNOWLEDGEMENTS

I would like to extend heartfelt thanks to the following individuals who made this study possible:

My thesis supervisor, Dr. Richard Schmid, for his inspiration, support and patience in setting high standards and providing me with the means to attain them;

My committee members, Dr. Gary Boyd and Dr. Steve Shaw, for their invaluable critiques;

Jean-Francois Arcand, for his incredible skill in rendering abstract ideas into elegant mathematical formalisms and then translating the whole lot into efficient, robust and ergonomically correct software modules;

Sophie-Julie Pelletier, Richard Trevail, Lyne Champagne, Talib Hussain, Martin Deveault, Hugues Belanger, Sonia Faremo and Alain Lamarche of the CITI PSS team,who provided enthusiastic assistance and moral support;

Dr. William Hodges and Dr. Jacques Ajenstat, both for their participation in the preparation of the study and in allowing me access to their classes to collect data;

Marcel Drouin, at CITI, for allowing me education leave to work on my thesis;

My husband, Peter, and son, Kemal, for their understanding and encouragement.

Table of Contents

List of Tables

List of Figures

CHAPTER 1. RATIONALE

Introduction

One-to-one tutoring has often been held up as the ultimate adaptive teaching and learning system (Bloom, 1984). With the advent of computer technology, researchers attempted to replicate this learning environment by increasing the intelligence of computerized systems (for example, Bork, 1981; Carr, 1977; Genesereth, 1978; Goodyear, 1991; Hartley and Sleeman, 1973; O'Shea and Self, 1983; and Wenger, 1987). One essential component of intelligence (machine and human-based) is adaptivity: the ability to respond differently to the same stimulus implying the ability to learn. In the context of machine-based adaptivity, the system would need to adapt to the particular goals, preferences and abilities of the users. Researchers discuss this capability in terms of user modeling (Kobsa and Wahlster, 1988). In educational applications, the term learner modeling is often used.

While real-time, dynamic adaptivity was never a problem in successful one-to-one tutoring, roadblocks were quickly encountered in trying to design computer systems that could behave in the same flexible manner (Hannafin, 1992; Hativa and Lesgold, 1991; Laurillard, 1988; Lesgold, 1994; and Park and Tennyson, 1983). Even when working with domains such as computer programming, researchers were unable to implement truly adaptive learning environments let alone trying to tackle more flexible and sophisticated learning objectives such as critical judgment or creativity-based problem solving.

It became increasingly clear that the success of individualized tutoring scenarios is due not only to the tutor's extensive knowledge of the subject matter and of pedagogical techniques but also to the ability to continually adapt the tutor's model of the individual learner to maximize learning success (Bennett, 1979). For example, the appropriate use of analogy depends sufficient knowledge of the student to pick the best analogy, on knowledge of more than one subject matter and the pedagogical rationale behind the use of analogies. More recent research on tutoring suggests that computerized tutors don't do much of this (e.g., Frasson and Gauthier, 1990). Furthermore, if one pursues the use of constructivist learning environments, the goal is not convergence and conformity but rather divergence, creativity and autonomy (Brown et al., 1990; Bruner, 1966; Collins *et al.*, 1989; Driscoll, 1994; Hill and Johnson, 1995; and von Glaserfield, 1988). In such cases, instead of always seeking to accommodate student learning styles and preferences, a learning system should also challenge students and help them to acquire new learning strategies.

In a computer-based system, adaptation is shared, such that the system adapts to the learner but learners adapt to the system (just as they would with a human tutor). One of the key challenges faced by designers of computer-based learning environments is to find a way in which to implement valid and useful models of the learner, one that can be continuously updated during the course of learning, in much the same way successful teachers and tutors do in learning interactions.

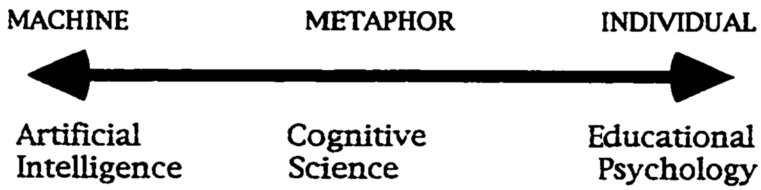
Disciplines Addressing Learner Modeling

Attempts to determine and exploit learner models have been addressed by three different disciplines: educational psychology, cognitive science and artificial intelligence. The educational psychology approach emphasizes individual propensities in information processing and looks at primarily stable or fixed predispositions in the form of learning styles (Allport, 1961; Ausburn and Ausburn, 1978; Davis and Schwimmer, 1981; Fischer and Fischer, 1979; Guilford, 1980; Keefe, 1979; Letteri, 1977; and Messick, 1984, 1994). Once such styles are measured, individual differences can then be accommodated by the learning environment. Although it will be an oversimplification, it may be useful to differentiate between the three disciplines in terms of the major focus of each. For educational psychology, this focus is on the inherent characteristics of both learners and teachers - the focus of these studies is on *people*.

In contrast, the information processing model of learning in cognitive science is one of contextual information processing variables as well as cognitive and meta-cognitive processes which are used by learners to adjust and modify their approach according to their motivation, background knowledge and prior experience (Anderson, 1984; Anderson *et al*, 1990; Ausubel *et al*, 1978; Bertel, 1994; Gagne *et al*, 1988; Orey and Nelson, 1992; and Simon, 1995). Cognitive science is a discipline which is concerned with the individual but at a more basic processing level. The metaphor of information processing to represent learning, used in the past thirty years, has been one that was heavily influenced by technology-driven environments. The focus of these studies is on the *metaphor.*

At the other end of the spectrum, the artificial intelligence approach has concentrated on developing explicit, primarily rule-based models of a domain of knowledge, with the model of the learner as a subset of this domain model. Any discrepancies between the learner state ("overlays") and the domain model are "corrected" through pedagogical interventions (e.g., Brown and Burton, 1978; Burns *et al,* 1991; Carr and Goldstein, 1977; Gentner, 1979; Goodyear, 1991; Hartley and Sleeman, 1973; Holt, 1990; Holt and Wood, 1990; Park and Tennyson, 1983; and Wenger, 1987). The artificial intelligence discipline has thus focused on the *machine* as a mechanical model of human learning. Figure 1 summarizes the different research goals that have been addressed by those three disciplines, showing a greater focus on the machine in Artificial Intelligence, a greater focus on the individual in Educational Psychology and a focus on the computer as a metaphor for learning in Cognitive Science.

Figure 1. The Three Disciplines Contributing to Learner Modeling



Learner Modeling Applications

Each of these three disciplines has in turn given rise to a number of different applications in the area of adaptive learning environments. Applications in educational psychology were rarely influenced by technological factors as most applications were not delivered technologically. Learning style inventories, for example, represent concrete applications or tools of educational psychology research. These were often administered, analyzed and acted upon without the use of any technologies. Teachers used the results of the inventories to "manually" adjust their teaching styles in order to better accommodate the needs of different categories of students. Learning environments were adaptive in as much as teachers adapted their pedagogical approaches. The role of the computer was for the most part non-existent.

Applications derived from cognitive psychology research contributed to advances in instructional design (e.g., Gagne's prerequisite hierarchy, Gagne *et al*, 1988) and most computer-based training applications. Applications covered a wide range of subject matters in education and training. Although these applications were driven by user needs, they were heavily influenced by the computer metaphor. Adaptivity in these learning environments was much more automated, and dependent on the possible adaptive responses that the technological delivery system could provide. The role of the computer was to provide a metaphor for learning and, in some cases, to deliver the learning.

Artificial intelligence applications were almost always influenced by computer systems. These applications tended to concentrate on modeling human learning but only in order to then endow the computer with similar learning capabilities, either to provide working models of human cognition or to improve the efficiency and effectiveness of the application systems. Adaptive applications that resulted from this research include intelligent tutoring systems (ITS) and microworlds or intelligent learning environments (ILE). Most applications addressed highly constrained, formalized or easy to formalize domains with an emphasis on skill acquisition or procedural training. The role of the computer was eventually expanded to include pedagogical tasks.

Learning Theories in Learner Modeling

One can similarly compare and contrast the three disciplines with respect to the learning theories that the applications are based upon. Learning theories in the field of educational psychology are based on individual characteristics and consider such individual traits as learning styles, preferences, past experiences and current competencies. Early theories (Bruner, 1961, 1966; Piaget,1954) were based on the developmental stage of the learner, especially in terms of when they were ready and capable of learning.

Directing learning meant guiding process. Developmental readiness to learn can be contrasted with cognitive readiness which is often defined in terms of prerequisite knowledge successfully assimilated to date. This developmental approach was distinct from that of reception learning which drew upon information processing theories of learning (Ausubel *et al*, 1978; Gagne, 1988) in which instructional design was heavily influenced by hierarchies of prerequisite knowledge. In the developmental approach, there was extensive testing of the student in order to determine both their initial and subsequent stages of knowledge acquisition as they interacted with the learning materials. Emphasis was placed on organizing content.

On the other hand, researchers in cognitive science relied a great deal on the metaphor of the computer to model learning. Early theories were behaviorist in nature (Skinner, 1968) which led to a pre-programmed instructional design that consisted of small learning steps and immediate feedback and correction upon completion of each step. Unlike educational psychologists, cognitive psychologists studied the learning systems and how students interacted with them in order to investigate and ultimately assist human learning. These learning environments became increasingly based on information-processing learning theories (Atkinson and Schiffern, 1968). In this respect, the line between educational psychology and cognitive psychology becomes less distinct. There is a great deal of overlap as well as significant evolution in both fields which seems to blur the boundaries. In addition, the same researchers contributed to both fields and some researchers switched from one to the other. The same dichotomy does, however, exist in both disciplines, between developmental and information processing theories of learning. Developmental theories led to theories of discovery-based learning (Bruner, 1961, 1966) such as those that are possible when interacting with a simulation environment. Information-processing approaches led to computer-based training environments that were usually competency-based or mastery learning instructional designs.

In artificial intelligence, to the extent that learning theories were considered at all, they were addressed with the purpose of programming appropriate learning algorithms and not for the purpose of assisting users to learn via the medium of the computer. There are a number of different camps in

AI research as applied to teaching and learning environments. The very first application of AI to education was simply an extension of expert systems (Clancey, 1984). These researchers were primarily engineers and computer scientists. Most instructional designs were implicitly drawn from information-processing learning theories and based on the designed interaction with the computer system. These included a study of expert-novice differences and an implicit mastery-based instructional design through immediate correction of any learner deviations from the expert path (e.g.,Carr, 1977; Carr and Goldstein, 1977; Kearsley, 1987; London and Clancey, 1982; Rickert, 1987; Van Lehn, 1987). Emphasis was placed on the best possible executable model of the domain such that the computer system could approximate as closely as possible the knowledge and reasoning of human experts. In this way, learners could "apprentice" themselves to a computerized "master performer" and essentially learn through observation of expert behavior and via corrective feedback whenever they diverged from expert behavior.

Applications later evolved into rule-based ITS applications, which typically left little room for learner control as the general model was that of the learner as a clean slate and the computer as the repository of knowledge to be transferred to the learners. This approach can be characterized as a primarily domain-based approach to learner modeling.

A slightly different objective was addressed by researchers such as John Anderson who studied the interactions between learners and ITS-type learning environments in order to derive theories of human learning empirically (e.g., Anderson's ACT* theory, Anderson, 1984; Anderson *et al*, 1990, 1995; Corbett *et al*, 1990). This group of AI researchers was made up of psychologists and cognitive scientists. Their focus was on how to better model and ultimately improve the process of human learning.

Another group undertook the study of human learning theories not to improve human learning via computers but in order to improve upon machine learning capabilities (e.g., Allman, 1989; Levine, 1991; Michalski *et al*, 1986; Mingall, 1995; Pearl, 1988). These AI researchers were mostly mathematicians and neurobiologists who studied aspects of human learning such as concept association and memory storage and retrieval in order to replicate these processes in a computerized environment. The best known applications of these pattern recognition technologies have not been in education and training but in such fields as robotics, machine vision, voice technologies and image recognition.

A separate camp of AI researchers distinguished themselves in advocating computerized learning environments that they claimed relied on learning theories from the developmental learning research, where students are given maximum choice and flexibility in how they went about learning. In this approach, the discovery learning environment, or microworld, is designed in such a way as to not only impart facts to the learner but also to aid them in acquiring the appropriate learning processes. Students are thus encouraged to actively explore their environment rather than to conform to an explicit model based on expert knowledge (e.g., Papert, 1980; Pask *et al*, 1972; Pask, 1976, 1988; Schank, 1990). This approach can thus be characterized as more of a learner-based approach to learner modeling.

The microworld learning theories are more constructivist in nature and can be said to belong to the family of interactional theories of learning. Interactional theories of learning (Bruner, 1961, 1966; and Vygotsky, 1962) take an even larger unit as the focus of study, that of learners and learning systems that are interacting with one another. The focus is thus not on the individual as in educational psychology, not on the metaphor as in cognitive science, nor on the learning system, as in AI, but rather on the interactions between learners, and between learners and the learning system. This relegates the computer to the role of tool of mediation rather than that of a repository of knowledge or that of an automated teacher.

There has been renewed interest in the application of social constructivism to the instructional design of learning systems (Driscoll, 1994). This is due to a greater interest in not just increasing the individualization of instruction but also increasing the intelligence or adaptivity of environments in which groups of learners or trainees interact not only with the computerized system but also with one another. Such scenarios include distance education systems and, more recently, the World Wide Web. This has meant increasing the scope of learner modeling to include not only individual differences but also characterizations of groups of learners, team dynamics and the interrelationships and interdependencies of the various roles adopted by the participants.

In summary, approaches to learner modeling in these three different disciplines appear to differ in at least three major dimensions: (1) domain-based vs. learner-based learner modeling, (2) machine vs. person-based learner modeling, and (3) connectionist vs. constructivist-based approaches to learner modeling. Table 1 shows the characteristics of the major types of learning environments with respect to these three dimensions. Domain-based approaches are exemplified by expert tutoring systems that grew out of early AI systems. In contrast, learner-based models are best exemplified by discovery environments or microworlds, where the onus is on the learner to explore the learning environment in any way they wish, rather than to mimic an expert's performance.

Table 1. Learning System Features with respect to the Three Dimensions.

	First dimension		Second dimension		Third dimension	
Learning System	Machine-based	Person-Based	Domain-Based	Learner-Based	Connectionist	Constructivi
Machine learning	|——|		|——|		|——|	
ITS, expert tutors	|——	——|	|——	——|	|——|	
Anderson and others	|——	——|	|——	——|	|——|	
microworlds, ILEs	|—	——|	|—	——|	|—	——|
learning style instruments		|——|		|——|		|——|

The connectionist approaches to learner modeling emphasize finding the best possible model of the knowledge to be acquired, of pedagogical expertise and of the state of knowledge acquired by the learner in the form of a knowledge representation that can be encoded in the computer. In contrast, constructivist approaches place the emphasis on a larger unit, that of a group of learners whose learning is mediated by a computer. The constructivist model thus addresses both the products and processes of interacting and learning in a social context.

Machine-based approaches to learner modeling are found in the application of AI to education and training that try to endow computers with the ability to learn. Their goal is not to use computers to assist human learning but to study human learning in order to improve on how well machines can learn. In contrast, people-based approaches to learner modeling, as exemplified by researchers in educational psychology and cognitive science, seek to study human-computer interactions in order to better understand and better assist human learning.

The majority of learner modeling to date has been domain-based, machine-driven and connectionist in approach. Learner modeling using learner-based, people-driven and constructivist approaches has not been as popular in the design of computer-based learning environments.

A New Methodology for User Modeling

Adaptivity in computer-based learning environments becomes more feasible only if comprehensive, valid and useful learner models can be developed. Educators insist that any instructional system (human or machine-based) must have some form of understanding of the student if learning is to effectively take place. It is argued in this study that in order to realize this type of learner model, contributions from each of the three disciplines must be recognized and integrated in a meaningful fashion.

Previous research to develop increasingly sophisticated ways of modeling the learner has typically drawn only from one isolated discipline and as a result, have encountered numerous obstacles. As Self (1990) has pointed out, with respect to contributions from educational psychology, we have not been successful in identifying learning styles using intelligent tutoring systems nor have we been able to associate them with different ITS instructional treatments. In the past, designers of computerized instructional systems have come up against many limitations in trying to model the learner, based on a cognitive science approach (Lesgold, 1994).

Some AI researchers have declared learner modeling to be an intractable problem and tried to circumvent the impasse by focusing their efforts on the design of rich simulation-based environments which have no need for a learner model (Orey and Nelson, 1992). Yet educational technology research has shown the crucial role played by a model of the learner in computer-based teaching and learning environments (e.g., Goodyear, 1991), as explained in more detail in the literature review chapter.

The following section describes the subset of variables that were selected from each of these disciplines.

Learner Model Variables from Educational Psychology

Learning style refers to the characteristics of the person rather than the environment. A learning style is a learner's preferred mode of processing (Keefe, 1979). The learning style perspective assumes that *a priori* assessment and categorization through the use of learning style instruments will serve to predict learner behavior in any given learning context as learning style is treated as a stable trait rather than a variable state. The learning style perspective is one which emphasizes individual differences and the key role these differences play in cognitive processes and outcomes. Characteristics possessed by individuals are usually assessed by some quantitative or qualitative means in order to assign the individual to one of a finite number of "types."

Two learning style instruments were included in this study: the Kolb Learning Style Inventory (LSI) and the Entwistle Approaches to Studying instrument (ASI). The Kolb LSI ostensibly measures learning style constructs that are more closely linked to underlying personality structures. Others view learning styles as more contextual or task-dependent features than fixed personality traits of individuals. The Entwistle ASI is an example of a learning style typology that is more directly linked to learning tasks and achievement.

Learner Model Variables from Cognitive Science

Cognitive psychology has studied the question of individual differences in how learners perceive and process information. Learners differ profoundly in what they do in learning, their success in any particular learning situation and in the stability of their behaviour patterns. A large part of the problem is understanding what differences learners bring psychologically to the learning situation that confronts them. This requires an assessment of the "initial state" of the learner, that is, postulating critical properties of the learner which interact with learning. This type of research led to aptitude-treatment-interaction (ATI) research (Cronbach and Snow, 1977).

The major finding from over two decades of ATI research is that orderly ATI patterns can be obtained and explained and that they involve prior knowledge, ability and personality variables in some cases (Snow, 1989). Both learner and task characteristics were found to affect the outcome of learning processes. Four major variables which contribute to learner achievement have been extensively documented in the cognitive science literature: prior knowledge of the content area, task perception and motivation variables, time spent on the task (or time spent learning) and meta-cognitive processes brought to bear on the learning session such as the level of learner control. These four variables are repeatedly found to be the most important moderators of ATI effects and they have therefore been selected for inclusion in this study.

<u>Learner Model Variables from Artificial Intelligence</u>

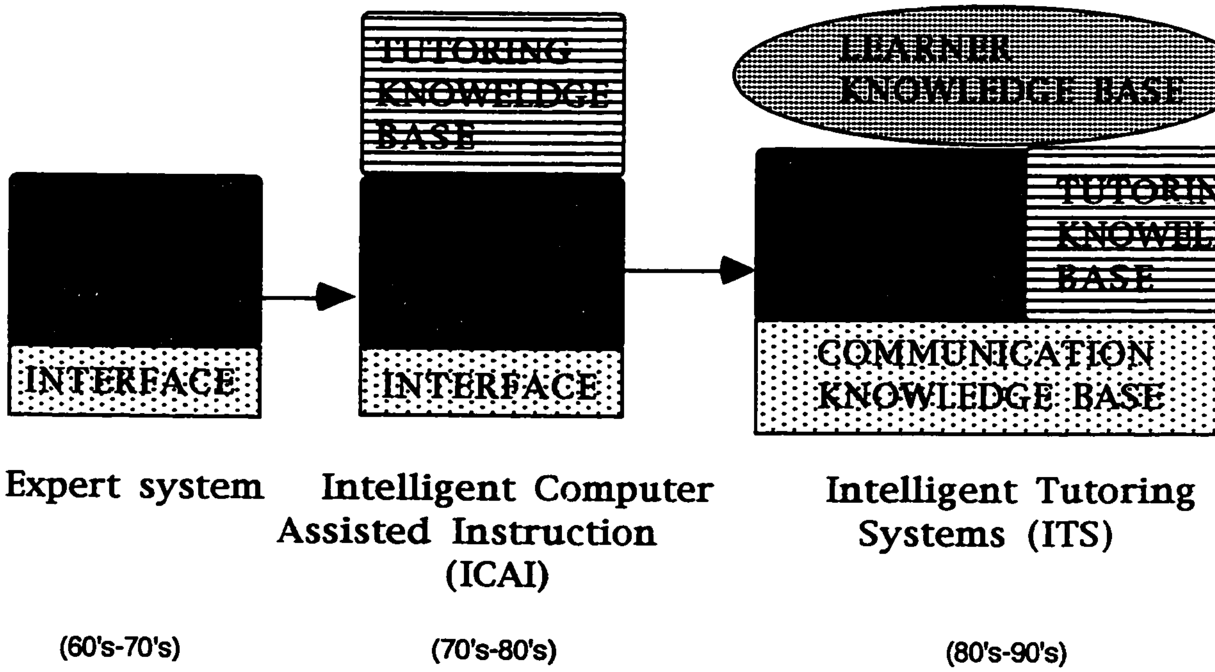
Learner modeling is a fairly late development in AI research. Early systems, such as expert systems, contained only a model of the domain. The interface with the user was usually a canned one (provided along with the expert system development tool). No accommodation was made for individual differences other than the optional explanations available on request for those users requiring an explanation of the recommendations made by the system. The overall emphasis was on a knowledge engineering approach to extract and rationalize expert knowledge and know-how on a given subject. This expertise was then made available in an executable or interactive mode to less knowledgeable users seeking assistance on a particular decision task. In order to make use of the same knowledge base for teaching, it was initially felt that all that was required was to include sufficiently explanatory texts accompanying each one of the possible decision paths.

When expert systems with extended or deep explanations were tested with novice users, it was found that another type of knowledge base was required: one that formalized the pedagogical expertise required to teach users. The same knowledge engineering approach was conducted, this time not with subject matter experts but experts in teaching that particular subject matter (e.g., Kearsley, 1987). However, it was soon realized that even with the addition of a pedagogical knowledge base, the system was still not sufficiently knowledgeable to teach. Thus a third component, called the learner model, was added. This was initially little more than a record keeping system that kept track of which lessons the student visited and how well they did on various tests. Eventually, the learner model was implemented, which kept track not only of student achievement in the form of test results, but also of student performance, in terms of how they solved a given problem. This approach was compared and contrasted with the way the expert model solved the same problem and any deviations were identified, corrected and explained. Thus, in addition to assessing the products of learning, the process of learning (in terms of problem solving) was also analyzed. This also added a new source of data about the learner : actual learner behavior as detected in real-time or during the learning process. The underlying approach was a knowledge-based bottom up approach that began with a model of the domain and added on missing elements (see Figure 2).

A different, parallel AI approach was to eliminate the need for learner models through the use of unguided, unstructured open-ended discovery learning environments. This approach was championed early on by Pask (1976) within the framework of Conversation Theory and Papert (1980) in the design of the LOGO learning environment and continues to have its proponents (e.g., Schank, 1990).

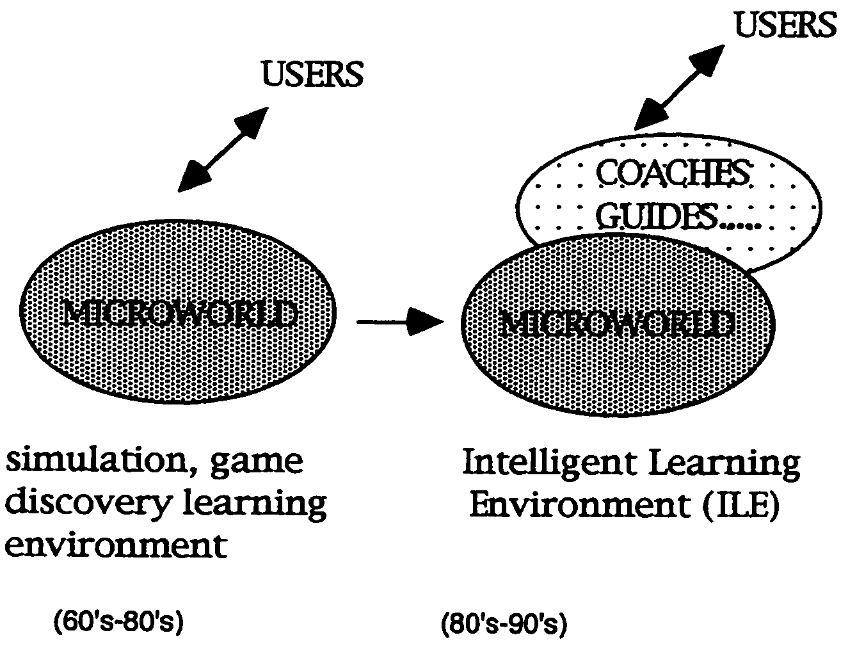
The development of Performance Support Systems (Gery, 1991) represents an extension of the ILE model, where an intelligent environment is designed to support both task performance and on the job learning of users. The PSS approach attempted to add more structured forms of job aids and learning aids to provide not only assistance but also proactive guidance throughout the interaction.

Figure 2. Evolution of Intelligent Tutoring Systems (ITS)



Each approach has its advantages and shortcomings. The ITS approach has a high degree of success in ensuring learning objectives are mastered but these systems tend to have difficulty handling the diversity and dynamic nature of learners. ILEs are quite successful at accommodating changing individual differences as learners are not restricted in their learning paths. However, they place the decisional responsibility on the student - the intelligence of these environments largely resides in the students and not the system.

Figure 3. Evolution of Intelligent Learning Environments (ILEs)



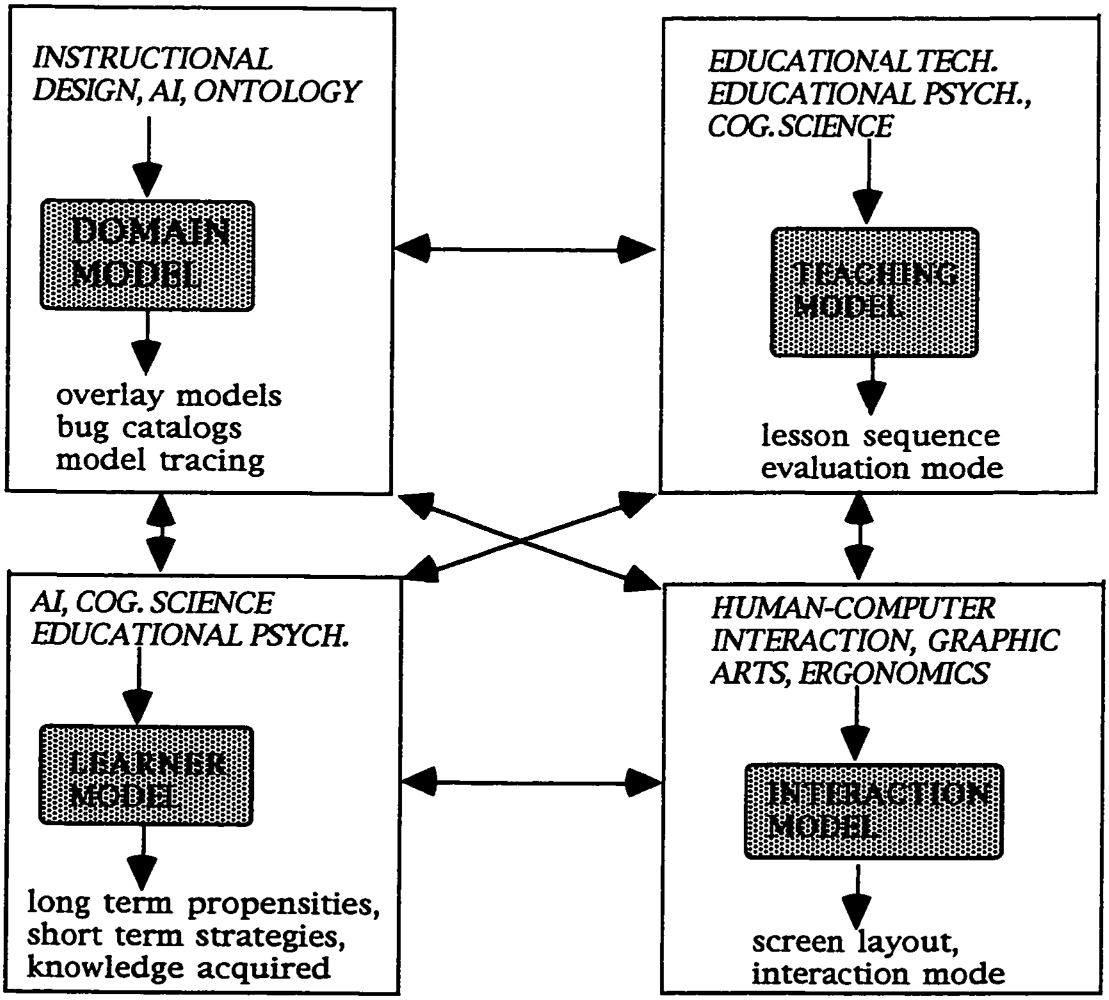
Learner Modeling Approach

An alternative response to the problem of learner modeling is to design and develop adaptive learner models and to do so in an integrative or holistic fashion. In this way, the advantages offered by both ITS and ILE approaches can be kept and at the same time the shortcomings of each can be compensated for to some extent. This approach to learner modeling would increase the "intelligence" of the system by increasing its repertoire of potential responses to the enormous variety of student actions.

Figure 4 summarizes the four major components to be considered in such an approach to the design of intelligent learning environments: a domain model, a teaching model, a learner model and an interaction model. Of the four, the first three are found in the conventional ITS approaches and the latter model is found in conventional ILE approaches. One of the major advantages offered by this approach to learner modeling is that no 'blinders' are placed on designers, i.e., they are not restricted to a single perspective based on a single discipline. Instead, one can integrate a number of variables and approaches from diverse but relevant disciplines into a single coherent system design. This is the approach that was applied to the problem of learner modeling in this study.

One of the ways to undertake such an approach would be to conduct learner modeling throughout the interaction in order to dynamically update the model. The system could adapt to the user based on the dynamic model, which would always be up to date. This would still allow learners full freedom of choice in navigating through the system since learner data would not be used to "correct" learners by minimized deviations from an ideal or expert model. Rather, learner data would be added to what is already known, inferred or guessed at about the learner. The addition of a dynamic dimension to the learner model could thus serve to complement, rather than replace, existing approaches to learner modeling .

Figure 4. Integrated Approach to the Design of Intelligent Learning Environments



A recent development in AI research may prove useful in carrying out more sophisticated analysis of learner behavior. Pattern recognition techniques, of which neural networks are the best known, were originally developed in order to have computers mimic some human pattern recognition tasks such as vision and voice recognition. These techniques enable computers to "learn" ("machine learning") by generalizing patterns from a large set of examples. Instead of programming the computer, the computer is "trained" to behave in a certain way by recognizing patterns in data sets. Machine learning techniques have been used in a limited fashion in learner modeling applications.

Machine learning techniques appear to represent the most promising new development in the field of learner modeling (Self, 1990), as explained further in the literature review chapter. These techniques appear to hold promise not only for updating or initializing existing student models, but as a means of deriving them in real-time, as students interact with the learning materials and data are generated on how and what they are learning. In other words, data can be obtained on both the products and processes of learning, including the navigational processes involved in interacting with the software environment that serves as the delivery vehicle for the learning materials.

A higher level learner model might also make use of keystrokes as the initial data source. Keystroke level models are often used in human-computer interaction or ergonomic studies to study and evaluate the usability of interfaces. Machine learning techniques have also been applied to the formative evaluation of user interfaces based on keystroke-level user models (Carey and Edwards, 1991; Carey, 1995; Stacey *et al*, 1991). In addition, machine learning techniques have also been used to observe on-line learner behaviors in order to deduce what type of learner they are and to diagnose student errors in an ITS (Woolf and Murray, 1992). Since there can be more than one explanation for a given student error, competing interpretations are tested out and the most likely explanation is selected by the machine learning system. Woolf and Murray thus made use of machine learning mechanisms to inform and update a learner model.

Studies at the Armstrong Lab (Sorensen, 1993) and by Beale and Finlay (1989, 1992; Finlay and Beale, 1991) used neural networks to determine novice-expert categories. Novice users tended to have a high number of help requests, conceptual errors and definition requests when usage was logged over a three-month period. In this way, the machine learning system was able to distinguish between novices and experts with statistically significant differences based on an analysis of the trace or cognitive audit trail left by students as they interacted with the system. However, finding expert-novice differences is fairly easy to ascertain using more conventional methods. A pretest on the course materials together with a questionnaire on experience using computers serve to establish the degree of familiarity learners have, both with the content and the delivery system, respectively. Machine learning techniques are not necessary to make such an initial determination. They become more useful if finer grained analyses are required; for example, to establish the degree of mastery at the level of a concept, topic or lesson module. The use of real-time assessments and updates of the learner model would be much less cumbersome than interrupting the learning process to administer more tests. In addition, the machine learning approach would be better suited to detecting changes in learner states over time, and detecting them as the change occurs. The latter is of particular importance to the design of adaptive systems.

Self (1987) was one of the first to point out the potential of applying machine learning to student modeling. He felt that the availability of large databases of information that a student could browse through at will (today's hypertext environments) would provide a wealth of information about the on-line learning behavior of a student. This coupled with new technologies to monitor a student's exploration of learning will enable us to design learning environments that can intelligently intervene to enhance the learning experience. A trace facility that can automatically track student actions and feed this as input data to a neural network should be able to provide the type of additional information that is required to render the learner model dynamic. Such up-to-date contextual information can then be added to existing information on the learner and be used to make instructional decisions about that particular learner. This study makes use of such a trace facility in a machine learning approach to learner modeling.

In summary, a novel methodology for dynamic learner modeling in adaptive learning environments was developed and tested in a specific context of a computerized course on neural networks. A subset of learner variables was incorporated into this learner modeling facility in order to assess their contextual usefulness. The Kolb LSI and the Entwistle ASI were selected from educational psychology. Prior knowledge, motivation, learner control and time on task variables were selected from cognitive science. The machine learning approach was selected from artificial intelligence. This study evaluated the usefulness of both the new approach to learner modeling and the particular subset of learner variables included in the learner model for this specific learning environment.

Description of Study

This research represents preliminary work in addressing new ways to attack the adaptive learner modeling problem. Part of the difficulty in learner modeling research lies in the complex nature of learners and of learning processes. Another source of complexity has been the fragmented nature and relative ineffectiveness of different disciplines to inform one another. The problem is analogous to that of the three blind men and the elephant: restricted types of information about a learner have been used to derive entire learner models.

This study does not represent a comprehensive analysis of all learner characteristics and learner behavior. Instead, representative variables from each discipline were selected and integrated in a new framework which brings together different types of information derived from different sources using different methods. These variables included the more stable learning predispositions from educational psychology research, the contextual learning processes employed by the learner from cognitive science research and the ongoing collection and analysis of learner behavior using machine learning techniques derived from AI research.

This dissertation made use of a novel methodology to look at the problem of learner modeling: i.e., I took the perspective of a system design based on theory rather than a bottom-up approach based on the addition of more technological components to the model. One of the objectives of this study was to select a useful subset of learner model variables from the three disciplines of educational psychology, cognitive psychology and artificial intelligence to form a coherent methodology for learner modeling. Previous work in learner modeling has been primarily from a bottom-up perspective where a learner model was added to an existing system rather than forming an integral part of the initial system design. The bottom-up approach has often been branded a "bandaid" approach that fixes some superficial problems or limitations of the initial system only (McTear, 1993). One of the major shortcomings of such bottom-up learner models is that they are often driven by the underlying technology. Thus, if the initial system is a knowledge-based system (e.g. rule-based expert system), then a learner model will be added on as an additional knowledge base. The objectives of learner modeling in the bottom-up approaches described were primarily to improve the functioning of an existing system. More often than not, the initial systems were not designed as learning tools (e.g., Clancey, 1986).

In contrast, the major objective of learner modeling advocated in this approach is to maximize adaptivity by increasing the variety of the system so that it can match the variety of learners using the system. In such an approach one can begin with a theoretical foundation that encompasses the three disciplines of cognitive science, educational psychology and artificial intelligence. Learner modeling can then be undertaken through a systematic selection and integration of key knowledge contributed by each discipline to the model. Once the comprehensive system design has been finalized, then the system can be developed and implemented. The learner modeling framework must therefore be designed using the system approach. A bottom-up or data-driven, empirical method was then used to instantiate the learner model with the data collected.

This study began with a theoretical analysis of the three disciplines and the development of a tool to accommodate this new systems approach to learner modeling. In this way, the contribution of each discipline to each of the major components of a computerized instructional system: the learner model, domain model, pedagogical model, and communication model (Wenger, 1987) can be evaluated (in terms of what each has contributed to date). More importantly, selected variables that have already been identified as being important can be put into this integrated framework. The relative usefulness of each source of information about a learner will vary. Thus their evaluation will necessarily always be relative or contextual. However, the objective of this study is not to compare one with the other. Rather, the objective is to undertake a holistic approach to learner modeling, one in which a number of key sources of information about a learner are integrated. The development of such a general framework for learner modeling will make possible synergy between the variety of literature addressing this problem.

This study was an initial attempt to examine simultaneously the relative contribution of a number of individual traits and contextual learner states. Learner variables must prove to be measurable and meaningful in order to be included in learner models for a computer-based learning environment. The computerized tool designed and developed for this purpose collected information on each of the categories discussed above. The relative contributions of each type of learner variable were assessed independently and in combination with others to identify those variables which can account for appreciable amounts of the variance in learner achievement.

The computerized learning environment consisted of a microcomputer-based learning module on neural networks. The course material was implemented in hypertext with an easy-to-use interface. There were five lessons spanning introductory concepts, the structure of neural networks, how they function, how to design one and their applications. Students interacted with the learning materials in a very flexible manner. They could spend as much time as they liked. They were also free to choose to be guided by the system or select lessons on their own. They could choose a number of options within each lesson such as help, examples, elaborate information, condense information or self-test. This environment thus allowed students to interact in diverse ways based on their learning propensities, goals and interests.

The ability of *a priori* measures to reliably account for a significant portion of the variance in learning outcomes appears to be in doubt. In this study the relative contributions of stable trait measures and ongoing learner state assessments were evaluated. The Kolb and Entwistle learning style instruments were selected from the Educational Psychology literature. The Kolb instrument is one of the most widely used tools of this type. The Kolb LSI represents an assessment of learner traits. The Entwistle instrument, on the other hand, is less commonly used. The Entwistle ASI represents the assessment of more context-dependent learner variables. Both instruments are purported to predict learner success and generic learner behaviors in learning environments. This was assessed using posttest achievement scores of learners in order to determine what, if any, additional posttest variance could be accounted for by these variables, beyond what was already accounted for by the cognitive science literature variables. Similarly, on-line variables, such as the frequency with which they selected various options and how long they spent in each selected option, were assessed with respect to the amount of posttest variance accounted for after contributions by the educational psychology and cognitive science variables.

Thus the theoretical descriptions of the different types of learners as assessed by both the Kolb and Entwistle instruments were used to establish general expected learner profiles. These profiles were then compared to actual learner behaviors exhibited by participants in order to evaluate the predictive value of such profiles. This represented a contextualization of the generic profiles through the mapping of general learner characteristics onto specific learning behaviors that could be manifested in the particular learning environment that was used.

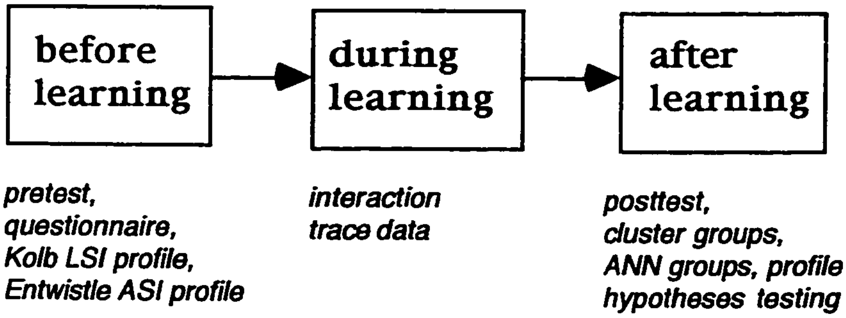
One advantage offered by the learning environment used in this study over more traditional testing scenarios is that it allowed actual learner behaviors to be observed. Self-report instruments such as the Kolb LSI and the Entwistle ASI address past or predicted learner behaviors, with questions such as "Do you prefer to study ...." or , "When you study, do you prefer...'" The learner trace variables represent tangible learner behaviors that took place during the process of learning.

The prior knowledge, task perception, time on task, learner control and posttest variables represent variables derived from the cognitive science literature. They represent well-researched, reliable and valid measures of domain-dependent learner variables. Prior knowledge was assessed through a pretest as well as questions on a questionnaire relating to familiarity with the field of neural networks. The related variable of prior experience was assessed through questionnaire items. Motivation was assessed by a questionnaire item. The time on task variable was assessed through an automated elapsed time feature of the learning environment. This variable is known to be a powerful predictor of learning performance and it is likely to be an indirect measure of other variables such as motivation and task perception. For example, students would be expected to interact longer with materials they find interesting or on tasks they perceive to be more important or more demanding. Learner control was assessed through the analysis of learner guidance mode selection. Pretest-posttest differences were evaluated as a measure of the learning that occurred with the system.

Finally, variables derived from the artificial intelligence literature were studied by analyzing learner trace data: the selections made, the amount of time spent in each selection, and the sequence in which the options were selected. These data were analyzed in two ways: conventional statistical procedures (cluster analysis) and neural network groupings were used to group together learners who exhibited common patterns of learner behavior. These learner data were analyzed independent of the previous types of data in order to ascertain whether or not any additional posttest variance could be accounted for. Any variables found to show a meaningful relationship with achievement data were then assessed to determine their potential contribution to data used to derive learner models. Groupings generated by cluster analysis and the neural network were then compared to determine whether group membership was similar and if not, to identify the source of the differences. This helped to evaluate the neural network methodology in terms of its usefulness as a real-time learner modeling tool. Figure 5 summarizes the types of data that were collected.

This study contributes an additional way of assessing differences in learning behavior as postulated by Kolb and Entwistle - in particular, whether expected or self-reported behaviors occur as learners interact with computer-based instructional systems and whether there are any links to other learner characteristics and/or learning outcomes. This information is collected through an automated trace facility that monitors student interactions with the computerized learning environment. The learning environment can be used with any type of self-report instrument in this fashion in order to quickly assess the predictive value of the instrument with a particular course.

Figure 5. Summary of Data Collected



The technology of neural networks provided a novel means of analyzing these qualitatively different data, which can be termed 'interaction trace data.' These trace data on learners can then be analyzed to assess their usefulness in prescribing for learning. It may be that some variables will show no contribution, others may contribute to a model of how people learn, and others still may prove useful in adapting instruction to individual learners. The learning environment designed for this study was evaluated with respect to its potential usefulness in assessing the local validity of variables derived from three domains (i.e., for the particular learning context), both as predictors of learning success and as useful components of learner models. This type of contextual learner modeling data is expected to complement existing domain-based and learner-based models.

In summary, this study looked at an integrated approach to learner modeling by selecting and assessing the contextual usefulness of variables from Educational Psychology, Cognitive Science and Artificial Intelligence. In addition, the usefulness of ANNs to add a dynamic component to learner modeling was explored. The approach to learner modeling thus consisted of a learner-based, constructivist approach.

CHAPTER 2. LITERATURE REVIEW

Overview

The literature review will first address the background history of adaptive learning systems an how the field of user and learner modeling evolved. Next, key studies from each of the three disciplines that this study drew upon will be summarized: educational psychology, cognitive science and artificial intelligence. The strengths and limitations of each field will be discussed. In the final section, related studies will be described in order to set a context for the proposed systems approach to learner modeling.

Adaptive Computer-Based Learning Environments

Adaptive education, according to Glaser (1984) is one which matches the developmental leve of an individual with respect to a skill with alternative environments. Landa (1976) has defined adaptiv instruction as a diagnostic and prescriptive process to adjust the learning environment to each learne McTear (1993, p.159) defines an adaptive system as "a system which automatically acquires knowledge about its users, updates this knowledge over time, and uses the knowledge to adapt to th user's requirements."

There is little consensus as to what constitutes individualized instruction. Perhaps the only principle that consistently has guided attempts to individualize instruction or accommodate individual differences has been to let students work at their own pace (Carrier and Jonassen, 1988). Developer of models tend to advocate the collection of information about general characteristics such as general ability, attitudes, or prior knowledge. The widespread use of microcomputers in teaching has opened up new possibilities for the individualization of instruction. These environments are flexible, often multimedia and they can monitor student progress as they progress through the course.

The quest for adaptation arose because it is not possible to anticipate the needs of each potential user given an indeterminate number of situations. In the context of computerized instructional systems, there is a need to identify which learner characteristics instructional adaptation should be based on , i.e., those which account for the most variability on a criterion task and which can be practically assessed on-line. This knowledge needs to go beyond the learner's preferences and beyond what the learner currently knows in the domain. A truly adaptive learning environment is one where all its components could adapt dynamically to the student by taking into account personal factors, cognitive styles, strategies and prior knowledge (van der Veer, 1990) as well as the learner's mental model of the adaptive system (Gentner and Stevens, 1983).

In order to be adaptive, a system must have the fundamental ability to "learn" the relevant characteristics of different users in order to provide a personalized learning environment, as well as to be able to evolve as the needs of a given user change over time. We need to know not only what to adapt but we also need a method for adapting (Benyon, 1993). In order for a system to continuously adapt to a user's needs, some form of dynamic assessment is first required (Lajoie and Lesgold, 1992) Realizing adaptation automatically requires mechanisms to observe the user while they are working and to record these observations, as well as mechanisms to exploit these records to build a user mode (Self, 1990). In this way, the system should be able to detect not only a learner's initial state but also conduct a sampling of states throughout the learning process. This process of user modeling is further described below, beginning with a general overview of the field, followed by a more detailed survey of learning modeling in Educational Psychology, Cognitive Science and Artificial Intelligence.

User and Learner Modeling

There are three major types of models in computerized learning environments: conceptual models represent the model of a system presented to the user by someone else, usually the designer of the system. A mental model is part of the thought process of the user when interacting with a system. People develop mental models internally as opposed to having models presented to them (Norman and Draper, 1986). Conceptual and mental models are models of the system, in contrast to the user model, which describes the user of the system. Daniels (1986) defines the user model as the model held by the system of a user. User modeling may in fact require both types of models based on the notion that any time two individuals interact, they each have a model or knowledge of the other. The assumptions each makes about the other are key elements when attempting to create a system that mimics a human intermediary in the process of interacting with a user. The ability of the system to behave in an interactive way allows the computer to "get to know" the user. This section will focus on the user model built by the system.

A sharp distinction between student model and learner model is necessary. A student model is a series of snapshots of the students' cognitive states. The learner model describes how students learn (Bierman, 1991). The natural temporal order, where diagnosis precedes treatment, is not the only reason that explains why research has focused on student models. The fact that there are few good theories of individual learning that can be implemented as models of the learner is another one. A student model could never be assessed with enough detail due to the limited bandwidth in the communication between the student and the system (Goldstein, 1979).

A distinction needs to be made between student model and student history as well: a student history is a record of interaction between the student and the system (e.g., the number of times that a task or problem has been presented to the learner, the number of times the learner has asked for help). There is still no universal agreement as to what should be stored within a student history (Dubs and Jones, 1991). Duchastel (1992, p.200) points out that the process of student modeling involves capturing the flow of interaction during a session, and interpreting that flow for adaptive purposes: "A trace of where the student has been is easily captured by a computerized learning environment. However, it is the interpretation of this trace that is difficult for a system to perform."

Kearsley (1987) discussed the need for sophisticated systems for individualizing instruction, with an increased degree of adaptation to the learner which may include: an initial assessment of a variety of students' aptitudes, personality factors, learning styles and interests. All these may serve to construct a learner model that interacts with the instructional decisions. The effectiveness of any adaptive instructional system depends on its ability to individualize instruction: capitalize on cognitive strengths, remediate cognitive deficiencies. Thus the system must 'know' a lot about the learner (Shute, 1993).

One of the challenges in building adaptive or user-responsive systems is to accurately model the student's state of understanding. Such a model must be constructed if the system is to be able to give the student sensible feedback (Bertel, 1994). This classic view of the student model is that it represents student understanding of the material to be taught. The purpose of a student model is to make hypotheses about student misconceptions and suboptimal performance strategies so that the tutoring module can point them out, indicate why they are wrong and suggest corrections (Brusilovsky 1994b). These will be referred to as domain-based learner models.

Some definitions of domain-based student models are: a data structure that reflects the assumed state of knowledge of the student concerning the target domain (Winkels, 1990); procedural knowledge, conceptual knowledge, individual traits and history (Self, 1987); all aspects of the student's behaviour and knowledge (Wenger, 1987); ideal student model (Anderson, 1984); artificial student simulating the development of student knowledge (Self, 1987).

Learner-based models focus on characteristics of individual learners rather than their knowledge state (i.e., how much of the target knowledge has been successfully acquired to date). Clowes *et al* (1985) call the user model a collection of observed and inferred abilities, beliefs, goals, attitudes and emotions. The user model serves as a means of distinguishing the user's needs and beliefs from those of the intermediary or system. In human-human interaction, the model can be derived from stereotypes, implicit knowledge, extralinguistic cues, nonverbal communication, the user's situation or a problem description. Characteristics of a user model can vary according to the system, the user and the task being performed. Daniels (1986) compiles a list of characteristics to be included in the user model: user status, goals, knowledge of the field, experience with the field, user background (employment, residence, etc.)

Rich (1983) identifies three dimensions helpful to organize the numerous descriptions of user models: static vs. dynamic, explicit vs. implicit and long-term vs. short-term. A static model is an unchanging model that is embedded in the system. A dynamic model is different for individual users and changes throughout the session. An explicit model is a stated model - that is, it is obtained through direct questioning of the user. An implicit model is inferred from the actions or responses of the user. Short term user models are concerned with what the user is doing at the time of the session, the goals the user has and what is being input by the user. A long term user model has information on expertise in a knowledge domain, which can be stored and updated in future sessions. This type of model would be applied to users who interact with the system consistently, where over time a model would be tailored to the individual user. This study attempted to design such a learner modeling environment, where a priori data from Educational Psychology and Cognitive Science measures can be used to initially populate a learner model (default values) and an ANN can be used to collect and incorporate additional information about learners as they interact with the learning materials.

User modeling is becoming increasingly important in a number of commercial environments such as: customized documentation, teaching systems, information-filtering and other tailored interfaces. For example, the field of Intelligent Help Systems (IHS) (Breuker, 1990) focuses on adaptive manuals that will support a user working with an application system. Student models also appear in the domain of adaptive user interfaces (Hayes-Roth, 1995). This is a relatively new field that studies interfaces that can adapt to suit the characteristics of users. User models can be designed in such a way that they are inspectable by users and depending on who the users are, this could prove to be quite beneficial to the learning process. This suggests an enormous potential for the model of the learner to serve as a learning tool in itself.

User models clearly have many potential applications in interactive systems but they may not be suitable for all tasks and all domains. Most of the environments in which user models have been applied have been structured ones such as computer-assisted instruction or medical advising systems. Potential applications of user modeling exist in hypertext systems (adapt the output text to the knowledge level of the user), databases (provide navigation aids which take into account users' interests, plans and goals), information filtering systems (which take information needs and interests of users into account), natural language systems (to tailor interpretation to user idiosyncrasies, knowledge, level of expertise and goals), tutorial systems (to adjust to the knowledge and abilities of the student), on-line help systems (to adjust advice to user goals and level of expertise and to adjust explanations so that they are tailored to the user's knowledge and they address the user's misconceptions) as well as user interfaces (to adapt the layout, interaction options and interaction modes to the user's tasks, abilities, and preferences) and adaptive testing.

The current state of the research on user modeling is that there has been a tremendous amount of research done in the last four years, the field is becoming scientifically established, researcl is spreading from universities to industrial labs, and there are many possible applications. Empirical evaluations have only recently started and there are, as yet, no commercial products equipped with user models. In addition, the three fields of inquiry, educational psychology, cognitive science and artificial intelligence have not resulted in much cross fertilization. The contributions of each of these fields to the learner modeling problem are discussed below.

Learner Modeling in Educational Psychology

Educational psychology is not a field of study characterized by a body of theory that is internall consistent and accepted by all psychologists. Rather, it is an area of knowledge characterized by the presence of several schools of thought. In some instances these may supplement one another, but a other times they are in open disagreement. As a result, there are no final answers to questions concerning learning and no theory can be found to be absolutely superior to all others. This is mirrore in the diverse approaches to learner modeling in Educational Psychology.

Educational Psychology approaches to learner modeling have emphasized relating user characteristics to very general stable learner attributes or traits. This is the major focus of learning style research: "...by individualizing education in terms of cognitive and learning styles, we can optimize instructional methods tailored to learner characteristics, thereby enriching teacher behaviour and beliefs, as well as enhancing student learning and thinking strategies" (Messick, 1984, p.69). While it is generally accepted that individual differences do exist and that it is necessary to be aware of them, little has been done to date to integrate the educational psychology approach in learner modeling. We need more sophisticated ways of modeling the learner in order to include all relevant learner information.

Learning style refers to the characteristics of the person rather than the environment. A learning style is a learner's preferred mode of processing. Individuals differ in the way they process information. Learning styles deals specifically with the organization and control of strategies for learning and knowledge acquisition. Some authors define learning styles as fixed patterns for viewing the world. Styles are characterized as "self-consistent regularities in the manner or form of human activity" by Messick (1987, p. 37). Allport (1961) defines a style as a mirroring of personal traits. They are inferred from consistent individual differences in the ways of organizing and processing informatio (Messick, 1984). Learners transform or process learning stimuli or information in ways influenced by their learning styles and then use these transformed stimuli to generate solutions to learning problem (Ausburn and Ausburn, 1978).

The learning style perspective assumes that *a priori* assessment and categorization through the use of learning style instruments will serve to predict learner behaviour in any given learning context as learning style is treated as a stable trait rather than a variable state. The learning style perspective is one which emphasizes individual differences and the key role these differences play in cognitive processes and outcomes. Characteristics possessed by individuals are usually assessed by some quantitative or qualitative means in order to assign the individual to one of a finite number of "types."

Guild and Garger (1985) list five basic ways to assess styles: self report, the most common method, is often indirect and may represent wishful thinking (e.g., Kolb, Entwistle); tests of a particular skill or task, are more objective but limited (e.g., Witkin's field independence); interview, can be seen as writing your own profile, and may be influenced by perspectives of the participants (e.g., learner autobiographies); observation, is carried out through checklists, anecdotal records and use of a computer trace; and analysis, is performed on the products of learning such as errors, or achievement tests. The Kolb Learning Style Inventory (LSI) is an example of learning style constructs that are more closely linked to underlying personality structures. Others view learning styles as more contextual or task-dependent features than fixed personality traits of individuals. The Entwistle Approaches to Studying Inventories (ASI) is an example of a learning style typology that is more directly linked to learning tasks and achievement.

<u>Entwistle's ASI</u>

Entwistle (1981, 1987, 1988; Entwistle *et al*, 1975, 1979, 1983) worked at measuring approaches to and styles of studying. He explored cognitive skills, cognitive styles and personality characteristics underlying different approaches to studying. Entwistle labels three learning styles based on how students approached the task of reading a scientific article: meaning oriented, reproducing oriented and achievement oriented. Meaning orientation entails a search for personal understanding, reproduction orientation is memorizing and achievement orientation is doing whatev will work to obtain high grades. Students with a meaning orientation are intrinsically motivated by personal academic interests, students with reproducing orientations are motivated by a fear of failure and those with an achieving orientation are extrinsically motivated by hope for success.

Entwistle used large sample sizes and a wide range of disciplines in order to evaluate his instrument and to establish norms for the dimensions measured. The Entwistle instrument has moderate reliability (Newstead, 1992) in assessing how a learner is most likely to tackle a learning task The ASI can be used to calculate scores on nine dimensions and to characterize users as favoring th meaning, reproducing or achieving orientations. Newstead finds:

> the Approaches to Learning Inventory was found to be a potentially useful measure: the predicted factors emerged, the scales were moderately reliable, and those students adopting a deep approach to learning were more likely to be successful in their exams..... the scale has been found to be fairly reliable, with reliability typically well in excess of 0.5.....there has, however, been considerable debate about the number and meaning of factors that can be extracted...there is almost universal agreement that the meaning and reproducing orientations are robust and genuine factors... (p. 92)

Boyd and Mitchell (1991) were among the first to advocate the use of the Entwistle ASI in the design and implementation of Intelligent Tutoring Systems. Allinson (1991) investigated the relationship of user behaviour in a typical computer assisted learning environment to individual learnir style, as assessed by Entwistle's Approaches to Study Inventory. She used Entwistle's ASI on 310 first year university students. Care was taken to ensure that the two groups were well balanced in terms of male/female and arts/sciences splits. To deliver instruction, a computerized hypertext environment, the Hitchhiker's Guide, was used. This environment had two navigational aids: a tour facility and an index which could be used by students. Students completed an interactive course on

Physiological Feedback Mechanisms in Humans which represented a realistic and demanding task. A students were asked to complete a questionnaire asking them to report on how they learned as well a a posttest to measure learning of the concepts presented.

The system automatically generated a log-file of each subject's navigation throughout the thir minute interaction with the learning material. From these log-files, a significant difference was found in the total number of screens viewed by the two groups (those who scored higher on the reproducing orientation and those who scored higher on the meaning orientation). Subjects in the high reproducing group looked at significantly fewer screens. If we measure coverage in terms of available information screens as a percentage of the total information screens seen, a significant difference is found: students who scored high on the reproducing orientation dimension exhibited greater coverage of the materials. They also showed significantly greater use of the tour facility whereas students who scored high on the meaning orientation showed significantly greater index usage. A longitudinal analysis of the log-files showed that the initial number of interactions (i.e., within the first 7.5 minutes) are the same for both groups. For each of the subsequent time slots however, the high meaning subjects showed a consistently increased rate of activity over the high reproducing group. There were no differences in the learning outcome between the two groups, which seems to support the view that differences in the responses to the inventory are not concerned with simply a general level of ability nor that the evaluation measures had some interaction.

Kolb LSI

Kolb (1971, 1976, 1977; Kolb, Rubin and McIntyre, 1979) describes four learning modes: thinking or abstract conceptualization (AC), feeling or concrete experience (CE), watching or reflective observation (RO) , and doing or active experimentation (AE). Two dimensions ranging from concrete to abstract and from reflective to active cross to generate four learning styles: thinker-doers are convergers, feeler-watchers are divergers, thinker-watchers are assimilators and feeler-doers are accommodators. Kolb viewed each state as a transitory one, with a transition through all four modes the most effective way to learn in most situations.

The four Kolb stages are: concrete experience, one actually experiences something such as a field trip where learners are involved in the experience making it possible to feel the situation and to become aware of the problems involved; reflective observation, one observes something that one wishes to be able to do and reflects on what has been observed; abstract conceptualization, experience through paper cases, models or computer-based instruction simulation; and active experimentation, which is trial and error experimentation conducted as formal experiments with planned procedures under controlled conditions

The Kolb learning modes and styles can be identified through use of the Kolb Learning Style Inventory, a nine-item self-report questionnaire or the revised 12-item Learning Style Inventory (Kolb, 1985). The latter version has improvements in reliability over the earlier version (Sims *et al*, 1986 and Veres *et al*, 1991). The Kolb inventory is brief and straightforward, with individuals responding to twelve learning situation questions. The instrument requires people to resolve tensions between abstract-concrete and active-reflective forms of learning by rank-ordering preferences for each of these forms. This instrument was developed primarily as a tool for career guidance.

The Kolb LSI has been extensively criticized as having questionable psychometric quality (Cornwell and Dunlap, 1994; Freedman and Stumpf, 1978; Newby, 1994; Sewall, 1986; Sims *et al*, 1986; Stumpf and Freedman, 1981). Internal consistency reliability has been estimated to be from 0.29 to 0.81 for the individual scales, with an overall average of 0.58. However, the two learning mode dimensions represented by the combination scales of AC-CE and AE-RO have consistently exhibited higher reliability ranging from 0.66 to 0.86 with an overall average of 0.78. The AC-CE scale represents the abstract vs. concrete learning mode dimension while the AE-RO scale represents the active vs. reflective mode dimension. The use of the LSI is appropriate in this study as the principal variables of interest are the differences in these learning modes.

Atkinson (1988) states that the revised LSI not only did not improve the test-retest reliability but may have weakened it further. On the other hand, Marshall and Merritt (1985) found the alternate form had moderate reliability and construct validity. The brunt of the attacks on the LSI centers on the instrument, not the experiential theory (Curry, 1987). Cornwell and Manfredo (1994) state that it may be time to revisit the Kolb LSI. As the general critique is against the use of the LSI instrument and the method of scoring results the authors suggest that the Kolb learning theory may be valid and the LSI may be useful to discriminate different types of learners if the data collected are analyzed in a different fashion. They advocate looking at the LSI but deriving different constructs than the original four types of learners.

There has been extensive use of Kolb LSI in computerized learning environments. Esichaiku (Esischaikul *et al.*, 1994) investigated whether or not individuals who exhibit certain learning styles, as assessed by the Kolb LSI, are more successful in using a hypermedia problem solving system. The learning environment consisted of the HyperSolver system, a hypermedia system that provides users with a number of project management tools such as PERT charts or histograms that they can use to solve a given problem. Each tool is defined, advice is given as to when to use the tool and how to use the tool and examples of its application are provided. Students were first classified using the Kolb inventory and then assessed with respect to the time it took to solve problems and to the quality of their solutions. No significant differences in time were found. Convergers and assimilators performed significantly higher quality work than their counterpart accommodators and divergers. This research supports previous findings by Stanton and Stammers (1990) who concluded that hypermedia creates an environment that allows for different levels of prior knowledge, encourages exploration and permits individuals to adapt material to their learning styles as well as findings by Bostrom, Olfman and Sein (1990) where individual differences were found to affect how people learn to use new software.

Bostrom *et al* (1990) report the findings of a series of studies that examined the influence of a novice's learning style, as measured by the Kolb LSI, in learning typical tools such as spreadsheets and electronic mail. A consistent pattern of findings emerges that indicates that learning modes is an important indicator of learning performance, both by itself and in interaction with training materials. They developed and used a research framework to study the computer learning process that integrated research from cognitive psychology (mental models, Gentner and Stevens, 1983), and educational psychology (ATI paradigm, Cronbach and Snow, 1977), and information systems and computer science. Bostrom's team chose Kolb on the basis that the theory is widely used in research and in practical applications such as the formation of project teams. It is assumed that even if learning style varies with situations, it will remain constant within a particular context.

It was expected that abstract learners would do better than concrete learners when studying a new software package because concrete learners must rely on prior referent experiences, which they do not have in this novel situation. It was also thought that active learners would do better in hands-on training because the emphasis was on learning by doing. They conducted four studies, with a total of 373 subjects. They found that abstract trainees consistently performed better than concrete trainees but not significantly so. Abstracts also took less time than concretes to complete the tasks, but not significantly so. Abstracts scored higher in the comprehension test and the difference was significant in one out of the four studies. This study was the third study, and represented the most tightly controlled setting (laboratory setting). No help was given to trainees in this group and subjects were randomly assigned to one of the two treatments (analogical vs. concrete models). While some of the non-significant findings are likely due to low sample size and lack of experimental control, in other cases it can be concluded that learning style as measured by the Kolb LSI did not explain the training outcomes measured in these studies.

Clariana and Smith (1988) found that students who were high concrete experience and active experimentation performed better in computer assisted learning environments when learning math. McNeal (1986) used the Kolb LSI with 173 students and found that convergers did the best for all instructional treatments, which consisted of matching teaching style to learning style, deliberately mis-matching teaching style to learning style, and no use made of learning style at all.

Logan (1990) studied on-line search behaviours and outcome variables among individual searchers to determine relationships between cognitive styles as measured by the Kolb LSI and five measures of on-line behaviour: the number of iterations (cycles), the number of directives issued (commands), the number of actual terms the system was asked to search for (descriptors), the total amount of time spent on-line (connect time) and the number of records printed out (references). Results indicated a consistent relationship between placement in quadrants of the LSI and high and low mean group scores. Assimilators showed higher mean scores on all five searching measures;

accomodators showed lower mean scores on four of the five. Accomodators and assimilators thus demonstrated opposing modes of on-line behaviour.

Summary

Bonham (1988) presents a comprehensive critique of learning style instruments. A problem that exists with learning styles is that one pair of bipolar traits for each theory probably is not complex enough to capture the essence of individual differences among human beings. Oversimplification is also evident in the failure to control for moderator variables such as sex or cultural background. The general view of learning styles is one of thinly developed theory and weak instruments, supported by fragmented research, often in settings not typical of adult education. One area of research needing continued attention is that of exploring links between style information use and either learner satisfaction or learning outcome.

Education researchers are searching for a theory that explains how students' learning styles vary. There are a number of competing models but no one model has been exclusively accepted by the research community. The single learning continuum model argues that each individual can be placed somewhere on a bipolar scale. The definite learning style model proposes that each person has one of a finite number of learning styles (e.g., serialist vs. holist). The situational learning style model proposes that each person is able to select from a number of possible learning styles, depending on the learning task at hand (e.g., surface vs. deep processing). Finally, the multidimensional learning style model specifies that each person has a different combination of styles. The current state of this theoretical development is that there is no clear-cut agreement on a universal learning style theory or measurement.

As a result, those dimensions demonstrated as having the highest discrimination amongst different types of learners were selected from each of the learning style inventories for inclusion in this study. The two Entwistle dimensions of meaning orientation and reproducing orientation were selected. In addition, students were assigned to one of four quadrants based on their scores on these two dimensions. From the Kolb LSI, it is not expected that the four learner types will prove to be significant. For this study, the two dimensions, AE-RO and AC-CE were selected for analysis. This partly replicates the Newstead (1992, p. 311) study, which looked at both the Entwistle ASI and the Kolb LSI in order to identify which, if any, had "some potential in assessing the learning styles of students."

Educational psychology focuses more on states that are more general influences on performance such as traits based on processing preferences whereas cognitive science research mostly investigates specific and contextual mental operations. The cognitive science approaches to learner modeling are discussed in the next section.

Learner Modeling in Cognitive Science

ATI Research

Cognitive psychology has studied the question of individual differences in how learners perceive and process information. Learners differ profoundly in what they do in learning, their success in any particular learning situation and in the stability of their behaviour patterns. Not all strategies are appropriate for all content. A large part of the problem is understanding what differences learners bring psychologically to the learning situation that confronts them. This requires an assessment of the "initial state" of the learner i.e., postulating critical properties of the learner which interact with learning. This type of research led to aptitude-treatment-interaction (ATI) research (Cronbach and Snow, 1977) that attempts to predict outcomes from combinations of aptitude and treatment variables, in particular, the behaviour of the individual *in-situ*. The major finding from over two decades of ATI research is that orderly ATI patterns can be obtained and explained and that they involve prior knowledge, ability and in some cases personality variables (Snow, 1989). This led to work on a generalized matrix framework that combined cognitive, conative and affective individual characteristics in order to conduct detailed multivariate analyses of instructional treatment variations. Both learner and task characteristics were found to affect the outcome of learning processes. Aptitude-treatment-interactions were found to exist but there are many complex combinations possible. Conventional research design and statistical significance testing are limited in providing interpretations of findings. Generic ATI hypotheses that can be used as a basis for instructional practice have yet to emerge. What is needed is a theory of the initial properties of the learner which interact with learning - the complex of personal characteristics th accounts for an individual's end state after a particular educational treatment.

Four major variables which contribute to learner achievement have been extensively documented in the cognitive science literature: prior knowledge of the content area, task perception and motivation variables, time spent on the task (or time spent learning) and meta-cognitive processe brought to bear on the learning session such as the level of learner control. Specialized prior knowledge and prior experience relevant to the instructional treatment are often found to be the mos important moderators of ATI effects (Carrier and Jonassen, 1988; Glaser, 1984).

From ATI research, the variables of prior knowledge, prior experience, and time on task were selected for inclusion in the systems approach to learner modeling that was used in this study.

Learner Control

Another important variable in the search to improve learning and thus the instructional system has been the degree to which learners had some flexibility or control over the learning environment. This can be related to the degree of structure or guidance provided by the learning environment. There is a range of possible instructional treatments, starting with rote learning, didactic, drill and practice, deduction, analogy, induction and discovery. Rote learning has no learner control whereas discovery learning has the most learner control (Shute, 1993).

High structure treatments are typically environments with direct instruction, whereas low structure environments are learner-controlled, discovery oriented environments. Discovery environments require planning and decision making functions to be carried out primarily, if not exclusively, by the learner. This approach is present in cognitive apprenticeship (Collins *et al*, 1989), situated cognition (Brown *et al*, 1990; CTGV, 1992) and social constructivist models of instruction (Vo Glaserfeld, 1988). The discovery learning approach is based on social cognitive and developmental psychology which emphasize the learners' development of metacognitive and self-regulatory skills superimposed on the process of rote learning and problem-solving.

Some interaction has been found in treatments involving differences in the structure of the instruction. In a highly structured treatment (with typically high external control and instruction chunked in small units) low ability learners perform better relative to low structure environments wher learners are expected to act much more independently. In low structure environments, high ability students do better and low ability learners do poorly (Snow, 1989). This evidence has been interpreted in terms of meta-cognitive strategies or self-regulatory skills associated with high ability learners. However, many other learner and task conditions can moderate these ATI effects.

Hannafin (1984) found students who perceive themselves as internally governed (i.e.,they assume personal responsibility for their performance and behaviour) perform best under internally controlled CAI. Students who perceive themselves as externally governed (i.e., they respond to imposed instructional demands) respond best to externally controlled CAI. Hannum (1990) found tha students with high prior knowledge make good choices about some but not all aspects of learning: pace, the amount of practice and whether to see an overview. High prior knowledge students made good choices on instructional strategy. In general, students made poor choices about topic sequencing, whether to practice and the difficulty level of practice. Experience with task sequencing (Brusilovsky, 1992) showed that novices tend to agree with system choices while experienced students prefer to make their own choice from a list of system options. Learners with good meta-cognitive skills also prefer and do better in relatively unstructured environments (Freitag and Sullivan, 1995).

Fogarty and Goldwater (1994) explored the use of an expert system as a means of increasing student control in accounting education but were unable to show any gains beyond that which could be attributed to increased effort. Fogarty and Goldwater studied the effects of varying levels of learne control. An increase in student control would seem to be a worthwhile objective; however, student control may not always be desirable. They used an expert system to enhance student control by providing the student not only with a highly available and robust study facilitator but also with the means to determine the examination process. Students were able to select particular sections of the course as well as the type of question in exams. The expert system allowed students to retake exams as often as they chose.

It is useful to distinguish local control from global or metalevel control. Collaborative ILEs for example, offer the learner a large measure of global control. A plausible scenario would be a design where the system is initially largely in control and as learners acquire expertise, they are encouraged t take more initiative and more control. In the learning environment used for this study, local learner control was offered as a choice in the selection of lesson topic sequence. This variable was then included in the subset of learner parameters for learner modeling.

Personality and Motivation Variables

Personality and motivation aptitudes also enter a wide variety of ATI patterns. Motivation plays a key role in cognitive theory as the expectant cognition of future outcomes provides the largest singl source of motivations for human action. Many behavioral psychologists consider all motivation to arise either directly from one's organic drives or basic emotions or from a tendency to respond that has bee established by prior conditioning of the drives and emotions. The key element of motivation is the paying of attention to one thing rather than to another. According to this viewpoint, learners do not have to want to learn in order to learn - they only have to be persuaded to study, which is in contrast to the view held by many behavioural psychologists.

Present day cognitive interactionists tend to avoid terms such as drive, effect and reinforcement. For them, some key concepts in motivation are goal, expectancy, intention and purpose. A person's psychological field includes purposes and goals, interpretation of relevant physical objects and events, and memories and anticipation. Accordingly, motivation cannot be described as merely an impulse to act triggered by a stimulus. Rather, it emerges from a dynamic psychological situation, characterized by a person's desire to do something.

The strongest result in ATI research appears to involve the state of anxiety and aspects of the trait of achievement motivation. Anxious or conforming students will do better with high structure sinc they are primarily motivated by extrinsic factors such as fulfilling course requirements or succeeding or a test. Non-anxious or independent students do better in a low structure environment as they are motivated by intrinsic factors such as personal interest in the content and goals for their personal futur (Snow, 1989). This is in turn linked to the time on task variable - those students with high intrinsic motivation tend to spend more time interacting with the learning content. The motivation and task perception variable was included in the subset of learner variables to be addressed by the systems model of the learner in this study.

Summary

With the powerful opportunities for individualization present in computerized learning environments, there has been an increased concern to model the student in order to have a deep basis for individualization of instruction. It is important to emphasize that concern with individualization is by no means restricted to computer-based education. Over the past decade, there has been an intensive effort by leading cognitive scientists to identify strong effects of aptitude-treatment interaction. What is meant by this is the attempt to show that, by appropriate adaptation of instruction to the aptitude of a particular student, measurable gains of learning can be obtained. One of the striking features of CAI work has been the absence of references to the extensive literature on aptitude-treatment interaction (Suppes, 1979). The conclusions based upon extensive data analysis, summarized by Cronbach and Snow (1977) show how difficult it is in any area to produce such effects. It is fair to conclude that at the present time, we do not know how to do it, and from a theoretical standpoint, it is not clear how we should proceed. Suppes goes on to say that intelligent CAI, or computerized learning environments that make use of artificial intelligence technologies, may provide a way out of this dilemma. This field is further discussed in the next section.

Learner Modeling in Artificial Intelligence

AI-based instructional systems were designed to achieve individualized instruction for a combination of situations and learning needs (Kaplan and Rock, 1995). A key feature of intelligent systems is the ability to diagnose learners (via the learner model) and tailor remediation based on the diagnosis (via the tutor). The intelligent system must accurately assess learners' knowledge and skills and or aptitudes, using principles to decide what to do next, then adapt instruction accordingly (locally or globally) (Shute, 1993).

After decades of work, development of effective learner models of the sort just described remains a very difficult challenge (Lesgold, 1994). There is a continuum rather than a sharp border between preprogrammed CAI and the autonomous capabilities of artificial intelligence-based CAI programs, or intelligent tutoring systems (ITS) (Bork, 1986). At one end of the continuum are CAI systems that range from very basic electronic page turning to systems that can generate exercises (Uhr, 1969), adapt the level of difficulty based on some measure of student performance (Park and Tennyson, 1983) or provide adaptive testing (Desmarais *et al.*, 1988) by selecting questions based on previous student performance.

At the other end of the continuum, AI-based systems may offer generative ability. These systems need not anticipate all possible learner actions and responses to each action by *a priori* programming. Rather, they are able to trigger appropriate rules based on learner actions. ITS programs adapt their actions based on explicit models of the student, of the domain, of pedagogical strategies and of the communication process (Clancey, 1986). Because CAI programs typically have only implicit models, interaction is predictable and non-motivating. They have little ability to adapt and tend to be based on the "one size fits all" approach that caters to the mythical average student. As a result, CAI systems have never realized the potential initially envisaged.

Any significant headway in machine-guided learning will require the development and effective implementation of new models of thinking. Leading this evolution is the movement from early programmed teaching systems which were based on behaviorist views of learning (Skinner, 1968), to ITS which is a more cognitively oriented form (Hartley and Sleeman, 1973; Carr and Goldstein, 1977; Goodyear, 1991). Bierman (1991) states that the amount of intelligence that goes into traditional systems probably has been underestimated. The whole history of artificial learning environments can be seen as an interplay between available technology and popular learning and teaching theory. When the computer first arrived on the scene, behaviorism with its stimulus-response paradigm was still the leading psychological theory. Early educational software was characterized as simple drill and practice. In the 60's and 70's, this situation changed gradually. Behavior was no longer seen as a series of stimulus-response pairs but as a consequence of complex cognitive processes. Several psychologists formulated global cognitive mechanisms (Ausubel *et al.*, 1978; Bruner, 1961, 1966;) which are processes between the stimulus and the response, using knowledge as a basic intermediary concept. Based on these, principles of instructional design were formulated (Gagne *et al.*, 1988; Merrill, 1983; Scandura, 1983) that were applied in the design of courseware. Most of the CAI today is still designed with these instructional principles.

The development of detailed cognitive theories was accelerated and stimulated by developments in the field of artificial intelligence (and vice versa). Introspection, which was totally discounted by the behaviorists, returned in the form of think aloud protocols. The analysis of these protocols became a method to collect data on human knowledge representation and cognitive reasoning. Part of the AI community claimed that humans were nothing more than complex computers and that most intellectual tasks could be done by computers in the near future. Intelligent systems, like expert systems, required that a great deal of effort be put into the analysis of human experts, knowledge representations and problem solving behaviour. It soon became possible to think that machines could teach as well as humans. Knowledge appeared to be something that could be represented separately therefore education appeared to be the transfer of this from the expert to the novice. The early cognitive and behaviorist CAI programs were criticized because they based their interventions on the last response only. In ITS a dynamic model of the student was maintained and system behaviour generated using conditions from this cognitive model.

Parallel to the AI triggered development of ITS, a separate reaction to the early cognitive CAI became apparent. Bruner (1961) had already pointed to the possible cognitive relevance of the act of discovery. Papert (1980) took up this lead and explicitly proposed that students would profit most from discovering the basic principles of the domain for themselves. The implicit pedagogical assumption was that, given the right discovery environment (e.g., LOGO), students would find their way on their own initiative and without any support. A cognitive diagnosis would therefore be unnecessary and no student model would have to be maintained by the system.

By the end of the eighties, the cognitivist paradigm was challenged by connectivist approaches. Connectionists focused on massive parallel distributed computations and simulations of neural networks. This paradigm succeeded in building systems that could learn to discriminate between patterns. The representation of knowledge in a connectionist system is accomplished by a set of weights while in an AI model of cognition by a set of logical statements. Learning is reflected in changes of the weights and the sole learning mechanism is learning from examples of patterns.

Recently, for a number of reasons, many AI-based cognitive scientists have turned away from their roots, most notably, the 'father' of educational expert systems, W. Clancey (1984). They currentl support the approach advocated by the situated cognition school of thought. Situated cognition doe not adhere to the basic AI premise that knowledge can be seen as something that is context independent and that can be poured into a student. These AI approaches to learner modeling are further detailed below. It is useful to make a distinction between the two major approaches to learner modeling in AI-based educational software: domain-based and learner-based. Domain-based models represent the learner as a subset of correct or expert knowledge for a particular subject matter commonly based on bug catalogs or overlay models (i.e., the learner model is what proportion of expert knowledge the learner possesses). Learner-based models, on the other hand, commonly represent the learner as a set of attributes or traits possessed by the individual (e.g., personality traits, preferred mode of information processing) usually by genetic graphs or default stereotypes.

Domain-based Learner Models in Artificial Intelligence

Domain-based modeling approaches in learner modeling consisted primarily of two types of models: the buggy model and the overlay model. Domain-based models evolved from expert system approaches to intelligent tutoring systems. Intelligent Learning Environments (ILEs) and Performance Support Systems (PSS) represent alternative approaches to learning modeling. Each is discussed in further detail below.

The majority of ITS can be characterized as having a straightforward, authoritarian or behavioris tutoring style guided by simple teaching rules, and a learner model based on *domain* knowledge (Wenger, 1987). A number of authors have developed systems based on this approach for a wide variety of domains (see, for example, Hartley and Sleeman, 1975). Most ITS user modeling efforts have concentrated on modeling learner misconceptions in discrepancy or bug terms. Bug catalogs are common incorrect or buggy versions of plans used to solve problems. They typically consist of an exhaustive list of all the likely ways one can make mistakes on a given task, which are often empirically derived. Given a specific problem and a fairly complete knowledge base of plans, instructional system based on bug catalogs will flexibly diagnose student behaviour and misconceptions. This approach fails, however, whenever the system encounters unusual or infrequently observed bugs, novel plans and ambiguous solutions.

The classic bug or mal-rule approach (Brown and Burton, 1978) tried to distinguish performance slips from genuine misconceptions in an arithmetic task (subtraction) but most errors we described at a large grain size. Misconceptions can, however, be described for any of a number of different bases, even for seemingly simple arithmetic skills (Ridgeway, 1988). Payne and Squibb (1987) also found the malrule approach inadequate for a large collection of children's algebra errors. This lack of robustness in reasoning about unknown plans could prove fatal in ITS. They are also very labor-intensive to develop and to maintain, requiring extensive cognitive task analyses of both expert and novices (Ohlsson, 1992). The buggy model assumes that the student's knowledge is fundamentally different from the expert's knowledge and contains numerous wrong concepts. The implication is that in order to detect student errors, we need to define a library of all the bugs a student can make when solving problems (Bertel, 1994).

In contrast, overlay models (Carr and Goldstein, 1977) are models that map learner knowledge onto expert knowledge. Learner knowledge is viewed as an incomplete version of experts' knowledge. This stems from a behaviorist view of expert-novice differences which postulates that experts possess a quantitatively larger amount of knowledge. This is in contrast to the cognitive approach where expert and novice knowledge are viewed as being qualitatively different. Some of th limitations of this approach are that there may be more than one correct way of deriving a correct solution and all possible deviations need to be pre-programmed as in the bug catalog approach. The overlay model assumes that the student's knowledge can be defined as a subset of the expert's knowledge. The implication is that the implementation of the expert's knowledge is sufficient to detec the majority of the student's misconceptions (Bertel, 1994). A record is kept about the skills the student has mastered and this record is used to decide whether or not the student has sufficiently mastered the domain. Implicit evidence is derived from a comparison of the student's and the expert's behaviour, structural evidence is retrieved from a network of dependencies among the skills and explicit evidence is derived from interactions with the student, by asking questions or using test cases. There are two important limitations of the overlay model: it will fail if the student's correct solution differs greatly from that of the expert's and it can only identify missing concepts, not misconceptions. The student is assumed to differ quantitatively from the expert. The best approach to student modeling may be to combine a number of different modeling approaches (Bertel, 1994).

We know already the effectiveness of one-to-one tutoring (Bloom, 1984) and the importance of having a model of the target competence that can perform the actual task with students or track their performance. Bloom (1984) found that the average student under tutoring was about two standard deviations above the average of the control group. The average student under mastery learning was one standard deviation above the average of the control class. ITS environments attempt to individualize instruction by automating the personal tutor. In ITS environments, the domain or subject matter model represents not only the knowledge to be communicated or acquired but also the standard against which student performance is evaluated. In this respect, the domain model serves as a criterion-referenced measure of student learning (Hambleton, 1984).

The process of learner analysis in Artificial Intelligence is aimed at collecting and inferring information about the student or their actions which does not always result in a student model. Such information about the learner can be used in three ways: to infer internal states assumed to produce behavior (assuming behavior is deterministic in nature) in order to reconstruct learner reasoning chains and account for observed behaviour; to interpret or make sense of observations and inferences in order to explain behaviour; and to classify or make relevant distinctions in order to characterize learner behaviour.

Researchers soon began trying to use knowledge bases not only as decision support systems but also as teaching aids. Early AI learner models were found to be too shallow (very little knowledge of the learner beyond detection of preprogrammed errors) and too static to take into account any student state changes (due to learning and contextual changes) (Hartley and Sleeman, 1973). Eventually, the learner modeling problem began being described as an intractable one as it seemed too difficult to obtain enough meaningful data on the learner and to update this data continually (Orey and Nelson, 1992).

One of the reasons for the disappointment may have been the bottom-up approach to user modeling. The first component to be built in a traditional expert system application was the domain knowledge base. This was the most labor-intensive part of the development work as knowledge engineers typically had to conduct extensive interviews with domain experts in order to extract the knowledge. Once this knowledge was conceptually organized into some type of knowledge representation (e.g., a semantic network or set of production rules), then it was a fairly straightforward task to input this representation into software called expert system shells. More often than not, the interface to the application developed was simply the interface provided along with the shell.

The next stage consisted of adding another, separate knowledge base that contained rules o how to best teach the content found in the domain knowledge base. Following this, it was then necessary to add yet another knowledge base, one that contained knowledge about the student users of the system. Throughout these additions to the original domain knowledge base, the interfac remained the same one that was provided along with the expert system development tool. Recent extensions include adaptive testing which adapts the next question to be asked in a test (pretest, or posttest) based on answers given to previous clusters of questions. Questions are clustered around concept prerequisite hierarchies in order to quickly pinpoint missing or incomplete student knowledge.

Intelligent Learning Environments and Performance Support Systems

Unlike ITS which are still relatively behaviorist and in some respects simply more sophisticated CAI, ILEs are outgrowths of the discovery environment approach as they attempt to accommodate individual learners by allowing them the choice of how to go through the learning environment and providing help but usually upon learner request. The recent development of Performance Support Systems (Gery, 1991) represents an extension of this model, where an intelligent environment is designed to support both task performance and on the job learning of users and often contain a simulation mode for the target task (e.g., air traffic control).

The fundamental unit that was built upon in developing more and more sophisticated ILEs consisted of a microworld, a discovery environment with an underlying simulation or game (or a combination of the two) that allowed users to interact with a high-fidelity model of some real-world system or process. The next generation of ILE had more advanced guidance features in order to ensure that learners met the objectives of the microworld (e.g., to discover and test hypotheses of physical laws such as gravity). These advanced help features addressed primarily navigational issues. Components such as coaches, tour guides, intelligent agents were added to the microworld to ensure students did not get lost and that they maximized their learning opportunities by covering enough of the topic nodes.

The best-known example of navigation in such environments today can be found in hypermedia applications (although not all hypermedia environments are ILEs). Hypertext provides an emergent illustration of innovations in computer use that depart from the traditional use of computers. The essence of hypertext is the dynamic linking of concepts, allowing the reader to follow preferences instantaneously and to be in control. The development of a topic is no longer linearly defined by editor or author and is limited only by the initiative of the reader. To some extent, hypertext allows the student to model himself and make selections based on that model (Carr, 1988). Hypermedia environments may introduce yet more complexity as hypermedia provides a new situation where students spend a lot of time studying on their own and following their own sometimes quite idiosyncratic processes (i.e., navigational routes through the learning materials). Traditional ITS student modeling approaches no longer apply as it becomes difficult, if not impossible, to coordinate several sources of student model information (Brusilovsky, 1994a).

Hypertext can be valuable not only as an delivery tool but also as a means for representing and organizing information according to different learning styles. In such an environment, tutorial material can be customized and tailored to the individual needs of students and instructors, landmarks being supplied by the rule-based tutor components. The major shortcomings of the ILE (including hypermedia) are that some structure is almost always required, students can get "lost" as they investigate the environment and, it is quite difficult to ensure whether or not learning objectives are met. The next section describes research in the area of learner-based student models.

Learner-based Learner Models in Artificial Intelligence

Few learner-based models exist in the AI literature. Vassileva (1990, p. 210) finds it "strange that no attempt has been made to incorporate student characteristics such as learning rate, level of concentration and preferred style of material presentation in ICAI student models."

The two major approaches to date have been genetic graphs and stereotypes. Goldstein (1979) used genetic graphs for learner-oriented modeling. A genetic graph is a semantic network tha attempts to capture the evolutionary nature of knowledge. Individual procedural rules are represente as nodes and links are used to represent their evolutionary relationships (such as generalization or analogy). The learner's individual learning history can be modeled as an overlay on the genetic graph. The degree of confidence in tagging a skill as acquired by a given learner can then be inferred from th number and types of links made to other nodes. The degree of difficulty of a given concept may be measured through a similar topological analysis based on the density of links connecting the node to the rest of the graph to assess how well it has or hasn't been integrated. While promising, this approach is still fairly domain-dependent.

A second approach has been the use of stereotypes which capture default information about groups of people. This simple but powerful idea was introduced by Rich (1979, 1983, 1989) who used people's descriptions of themselves to deduce the characteristics of books they would probably enjoy. The GRUNDY system is the only modeling system which reflects individual characteristics of the user not directly connected with a particular domain. GRUNDY uses the vocabulary of the user to infer a model from a set of existing stereotypes such as "sporting businessman" or "intellectual feminist". It does this by, for example, inferring the degree of concentration from typing errors, misuse of commands or from preference for examples and demos (inductive) or preference for problem solving (deductive).

A stereotype represents a collection of attributes that often co-occur in people. They enable the system to make a large number of plausible inferences based on a substantially smaller number o observations. These inferences, must, however, be treated as defaults, which can be overridden by specific observations. Stereotypes permit us to make predictions which need not be completely accurate because the role of the predictions is not to take the place of specific knowledge about the individual but to provide a basis for action until such specific knowledge becomes available. Sollohu (1989) suggests that the best use of stereotypes may be as initiations of student models with default values (or best guesses). As more knowledge is gained about the student (such as answers given to tests or use of help functions) the student model can gradually evolve to become based more on the student's actual behaviour.

In other words, if one is given the task of constructing a model of another person, one has tw choices: construct the model one piece at a time as information is gleaned from experience with the person, thereby viewing individual facts about the person as independent events; or observe that, empirically, facts about people are not statistically independent events but rather that they can be clustered into groups that frequently co-occur. A new way to build a user model is by adding to it a whole cluster of facts at a time, as soon as some evidence that is known to be a predictor of the cluste is observed. This is the approach used in stereotypes.

Stereotypes are important to people because they permit us to make predictions about othe people on the basis of an amount of evidence that is sufficiently small that it can be acquired before action is required. For these predictions to be useful, they do not need to be completely accurate. The role of these predictions is not to take the place of specific knowledge about the individual; instead, their role is to provide a basis for action until such specific knowledge becomes available. As that happens, the model of the individual must be updated, and stereotype-based predictions that conflict with specific observations must be abandoned in favor of the specific facts.

Stereotypes are long term models of individual users. Adaptation is an important part of an effective stereotype-based system because of the difficulty of defining accurate stereotypes *a priori*. Adaptation consists of modifying the certainty measures on the basis of each experience with a user.

Other approaches to adaptation are also possible. For example, stereotypes can be viewed as concepts to be then identified with statistical concept-learning methods or with a neural network trained to recognize instances of stereotypes (Rich, 89).

Kaunanithi and Alspector (96) discuss the use of clique-based user models in movie recommendation tasks. Cliques are quite similar to the notion of stereotypes (Rich, 83), communities (Orwant, 95) and social filtering as used by Maes (94). Information filtering is based on profiles that describe either individual or group preferences. Such profiles represent long-term interests of the user. On the other hand, information retrieval from a database requires well defined user queries which reflect very short term or instantaneous needs. Depending on the degree of interaction, we can characterize the information filtering task as passive or batch while information retrieval may be characterized as active or on-line. The clique-based approach is built on the hypothesis that the average rating of a clique of users is the best indicator of an individual's future rating. A set of users form a clique if their movie ratings are similar. Each user for whom we wish to predict ratings has a unique clique composed of other users whose ratings are similar. The members of the clique who have rated a movie that the target user has not seen predict the rating of the target user for that movie. The Pearson correlation coefficient (which is a normalized dot product of the vectors of the ratings of the two users) was used as a similarity measure. The features of a movie are extracted and used in the recommendations (e.g., expert critic ratings, movie category, director).

Maguire *et al* (1995) used a combination of user-based and domain-based stereotypes. The first level consisted of an occupation stereotype. The second level represented an individual's level of expertise, given their occupation. This latter level is 'hard-coded' by a knowledge engineer who makes judgments such as " a clerk has high expertise in word processing; a manager has moderate expertise in word processing."

Beaumont (1995) used an adaptive hypermedia system to build a model of the goals, preferences and knowledge of individual users and use it throughout the interaction for adaptation to the needs of that user. The Anatom-Tutor contained a rule-based user modeling component and used stereotypes for making assumptions based on the general information and used deduction mechanisms for inferring new declarative information from that which is already in the model. The user model contains general information on the user's prior knowledge in anatomy, and specific knowledge of the material covered in the Anatom-Tutor lessons. The tutor first decides on the general link structure by looking at the model to find out the user's goals. At the text level, the user's level of experience, the lessons and lectures he has already worked on are taken into account for choosing a default expository style. Then the content and actual local expository style is chosen by comparing the user's fine-grained knowledge with the material covered in that part of the lesson. Adaptive hypermedia systems such as the Anatom-Tutor can help bridge the gap between tutors that provide guided or adaptive machine-driven education (CAI, ITS) and the class of environments that provide 'free' student-driven learning (ILE, hypermedia) by allowing student initiative coupled with an ability to adapt to the student and guide them implicitly but significantly by changing the content and hypermedia links.

While stereotypes appear to be the most promising learner-based student modeling approach they are operating at a fairly large level of granularity. Adaptivity can only be made to generic types of people or to one of only a few cliques. Other researchers have addressed the problem of how to model a single given user performing a well-defined cognitive task at a much deeper or more sophisticated level. These efforts have led to empirically-derived models of learning as described further in the section below.

Empirical Models of Learning

Clancey (1986) was one of the first AI researchers to look to cognitive psychology when modeling the learner. He proposed the idea of qualitative models which characterize spatial, temporal and causal relations. This shift described mental processes rather than quantifying performance with response to stimulus variables. Individuals are described in detail and not in stated generalities. Psychological interpretation is given to qualitative data rather than statistical treatment to numerical measurements.

One of the most impressive achievements of cognitive scientists has been the production of elaborate computer simulations of intelligent behaviour that, at least at face value, incorporate many aspects of cognition. Some examples are; programs that simulate expert and novice behaviour in particular domains such as physics and systems to simulate language learning (Gentner, 1979; Gentner and Stevens,1983).

Anderson (1984) stated that the major contribution of cognitive psychology would be to provide a well-specified model of the target behavior to be tutored - a goal to which the instruction is directed. In the areas of mathematics and science, this amounts to developing a problem-solving model of the ideal student, which specifies the problem-solving goals, the representation of the relevant knowledge, and the operators that control the transition among goals. Such student models can be used to represent both the current state and the state desired for the student at the end of instruction.

Computerized tutors can then guide students through the problems, trying to make their steps correspond to those of the ideal student model. If they don't, immediate explanation is generated which tells the student what the correct step is and why it is correct. This mode of tutorial interaction is referred to as model tracing. Anderson and his colleagues later applied the model tracing paradigm to build one of the few commercially successful intelligent tutoring systems: the LISP Tutor (Anderson *et al*, 1990). This is a program that provides assistance to students as they work on LISP coding exercises. The program presents problem descriptions and as the students type answers, the tutor monitors and stands ready to provide assistance at each step. The tutor has been in use in an introductory Lisp course at Carnegie-Mellon University since the fall of 1984. The lesson material consists of approximately 240 exercises covering the first 12 chapters of an introductory Lisp text. The Lisp tutor represents a relatively large and stable intelligent tutoring system.

The Lisp tutor was developed to serve as a real-life application of the ACT* model of skill acquisition (Anderson, 1984). One goal was to teach Lisp more effectively but a second goal was to collect detailed data with the tutor on the course of skill acquisition in a natural setting. The tutor proved successful on both counts. Evaluation studies have shown that working with the tutor is more effective than doing the same exercises on your own (Anderson *et al*, 1990). In one study, students using the tutor completed the exercises in a little over half the time required by the students working on their own and scored equally well on a posttest. In the other study, students completed the exercises 30% faster and scored 43% higher on a posttest. It should be noted, however, that the tutor is not as effective as a human tutor. If the student repeatedly makes errors that the tutor cannot recognize or repeatedly makes the same type of error, the tutor will tell the student what code would work in that step, explain why and fill in the code for the student.

More recently, Anderson and his colleagues reflected back on the 10-year history of tutor development based on ACT* theory (Anderson *et al*, 1995). Much of the ACT* theory was concerne with the acquisition of cognitive skills and was tested in the domains of proof generation in geometry and initial programming skills in LISP. The theory holds that a cognitive skill consists in large part of units of goal-related knowledge. Cognitive skill acquisition involves the formulation of thousands of rules relating task goals and task states to actions and consequences. The theory employs a production-rule formalism to represent this goal-oriented knowledge.

The authors reflect that they have come a long way from their original goal of putting the ACT* to a tough test. The empirical data harvested has played a major role in leading to a new ACT-R theory They have totally abandoned the original concept of tutoring as human emulation. "We now conceive of a tutor as a learning environment in which helpful information can be provided and useful problems can be selected. We are able to take actions that facilitate learning because we possess a cognitive model of where the student is in that task (p. 202)." Ten years later, it appears that the cognitive modeling approach still seems viable and important in new applications of cognitive tutors.

Summary

Current user models are too simplistic and too static to reason effectively about human learning. Before a student begins to work with a tutor, for example, control issues need to be addressed concerning the acquisition of and reasoning about pedagogy, domain topics, and machine responses. A negotiation process whereby student commitment and motivation are addressed precedes any instructional or learning interaction. Once a student begins to interact with the system, different issues need to be addressed about the dynamic analysis of student behaviour, automatic diagnosis and remediation, identification of appropriate pedagogical strategies and generation of effective responses.

On the other hand, Collins (1996) maintains that there may not be a need for such deep, sophisticated student models. He argues that effective teaching depends on more than the diagnos of student errors. One needs to know not so much why students got into trouble in the first place but rather how to best get them out. To this end, he feels there should be far less emphasis placed on augmenting the intelligence of an instructional system. Instead, the focus should be on the interactivity students can have within the system (both with the system and with other participants). The best way to model the student is to pick up enough clues on where he has been, where he wants to go and then prescribe the best supportive structure to help them reach their goals. Thus the collection of pertinent on-line learner data during real-time (i.e. as they are interacting with the computerized learning environment) may yield more useful data for pedagogical prescriptions than th more traditional approaches to learner modeling.

From the traditional approaches to learner modeling in the fields of educational psychology, cognitive science and artificial intelligence, the subset of learner variables used in this study have bee established. The method of learner modeling, however, is based on a novel approach that has begun to appear in some of the literature addressing alternate approaches to learner modeling in the field of artificial intelligence. These are described further below.

Alternate Approaches to Learner Modeling in Artificial Intelligence

Several new approaches to learner modeling have been developed in recent years. Four nev methods based on advances in computational reasoning appear to be quite promising: constraint-based models, fuzzy diagnostic models, Bayesian probability models and pattern-recognition based models.

Constraint-based models represent learner knowledge as constraints upon expert or correct knowledge. This is an extension to the overlay model with more sophisticated reasoning about domain concepts. This approach can also recognize creative or novel solutions and allow freer exploration of the subject matter by learners. Disadvantages include viewing the learner as a 'buggy' expert and applicability to procedural domains only (Ohlsson, 1992).

Fuzzy diagnostic models use statistical procedures to represent different levels of knowledge by a probabilistic distribution. This allows for a much finer-grained tutorial intervention (Hawkes *et al*, 1990; Katz *et al*, 1992; Self, 1990). Bayesian probability models represent learner characteristics by using conditional probabilities involving either linear or multiple regressions. Data is collected and then the probability that a learner knows a particular set of problem-solving rules is calculated. Disadvantages of both these statistical approaches include the high computational requirement and the labor-intensive definition of probabilities (Katz *et al.*, 1992).

An exploratory effort is underway at the University of New Mexico (Kaplan and Rock, 1995) in the diagnosis of student subskill errors. The system's goal is to find whether a genetic algorithm could automatically produce a good guess as to which subset of subtraction subskills a student lacks. Van Lehn and his colleagues (1994) are using a new approach to student modeling called OLAE (On-line Assessment of Expertise) which uses Bayesian networks. These newer approaches are using new techniques but treating the same type of learner data. As a result, they are all domain-based learner models. A different perspective can be found in the human-computer interaction literature, as described in the next section.

In his most influential paper, Self (1992, p. 281) advocated the need for a new discipline he called Computational Mathetics (CM) defined as "the study of learning, and how it may be promoted, using the techniques, concepts and methodologies of computer science and artificial intelligence." Pattern-recognition techniques have been used in a limited fashion in learner modeling applications. Most have consisted of enhancing learner models developed using more traditional methods such as ITS. Some researchers have used pattern recognition capabilities in the form of induction from a series of examples (rather than deductions from *a priori* coded rules) in order to update student models and to overcome the "brittleness" of rule-based systems (rejection of a rule when any deviation is detected in inputs). Machine learning techniques refer to systems that are able to generalize from a set of examples. They appear to represent the most promising new development in the field of learn modeling and are described below.

Machine Learning- based Learner Models

Holt (1990) conjectured on the possibility of using neural network models in ITS by applying them to the pattern recognition task of classifying students to provide some temporary resolution to the intractable student modeling problem but felt that this was mere speculation at this stage. Machi learning techniques, notably neural networks, have since been used to observe on-line learner behaviors in order to deduce types of learners (Woolf and Murray, 1992). Neural networks are artificia intelligence software that can learn based on the use of historical data with weighted criteria. They represent a more effective means of doing basic pattern recognition or predictive types of methodologies that are alternatives or extensions to statistical modeling.

Neural networks provide significant advantages in problem processing problems that require real-time encoding and interpretation of relationships among high-dimensional variables. A neural network's ability to change its connections in response to experience makes it an ideal tool for modeling cognitive processes in the brain and also gives insights on how the brain might store information as memories (Allman, 1989).

An artificial neural network (ANN) is an information processing paradigm that was inspired by the way biological nervous systems such as the brain appear to process information. (Michalski, *et al* 1983) The key element of this paradigm is the novel structure of the information processing system. I is composed of a large number of highly interconnected processing elements (neurons) working in union to solve specific problems. ANNs learn by example. An ANN is configured for a specific application, such as pattern recognition or data classification, through a learning process. Learning in biological systems involves adjustments to the synaptic connections that exist between neurons. Thi is true of ANNs too.

Their benefits include:

- generalization: neural networks can produce good results from exposure to known data even while working with incomplete or inaccurate input;

• ability to learn: neural nets are trained rather than programmed so their performance improv with experience;

• parallelism: they can leverage performance and fault tolerance;

• speed: using neural technology is quicker than scanning data when trying to locate trends;

• performance: neural nets can improve performance over time;

• data importance: they can evaluate many factors and ignore those not providing value;

• timeliness (perhaps the most important benefit): they enable analysis of conditions and diagnosis in real time.

On the negative side, neural networks require extensive data pre-processing. Since they us number, not text, the ranges of variables must be carefully scaled. There is a steep (human) learning curve. Although they are good at pattern recognition, they are poor at computational tasks. There ar cases where neuronal models are difficult to interpret.

ANNs have been applied to an increasing number of problems of considerable complexity. Their most important use is in solving problems that are too complex for conventional technologies: problems that do not have an algorithmic solution or for which an algorithmic solution is too complex t be found. ANNs are thus well suited to problems that people are good at solving but computers are not, such as pattern recognition and forecasting trends in data.

Machine learning techniques were initially formulated to enable computer-based systems to acquire correct concepts and procedures automatically, as a way to overcome the knowledge acquisition bottle-neck in expert system development (Gilmore and Self, 1988; Mengel and Lively, 1991; Michalski *et al.* 1986; Posey and Hawkes, 1988). Systems were developed that could learn fro examples. Van Lehn (87) was one of the first to use this approach to model human learning. His SIERRA system represented an alternative data-driven way of modeling users instead of *a priori* programming of models of correct and incorrect behaviours. The underlying assumption was that student learning could be represented as a set of procedural rules. However most attempts were still fairly domain-based. They used student data to update the domain-specific knowledge base in order to produce a student model that better accounts for student behavior (Wilkins *et al*, 1986).

More recently, machine learning models have been combined with knowledge-based techniques in a promising new direction for learner modeling (Nichols *et al*, 1995). Researchers have used a neural network to diagnose student errors in an ITS. Since there can be more than one explanation for a given student error, competing interpretations are tested out and the most likely explanation is selected by the machine learning system. Woolf and Murray (1992) made use of machine learning mechanisms to inform and update a learner model. They enhanced the capability of the model to acquire new information and new representations through cases. Nichols (1995) also proposed the use of neural networks to perform cognitive assessment. In this approach, the neural network is treated as a black box that can be trained to classify response patterns as to underlying cognitive skills. Holt *et al* (1990) proposed neural network-based ITS that use pattern recognition to identify students and to classify them based on their responses. Ur and VanLehn (1995) developed an ANN-simulated student to train human tutors.

A student model must reflect changing knowledge and spontaneous reorganization of that knowledge and continually reassess student knowledge. Traditional inferencing mechanisms alone cannot easily do this type of dynamic learner modeling therefore some type of student behavioral trace is needed.

On-line Learner Analysis Using ANNs

A study at the Armstrong Lab (Sorensen, 1993) used artificial neural networks to determine novice-expert categories. Beale and Finlay (1989) used a neural network to model a learner for an ITS. This is example-based rather than knowledge-based and has greater generalizability as well as the ability to handle noisy data. The input vectors are different classes of interactions (e.g., types of errors, commands, selected menu options) and the output is the best fit pattern (novice or expert). Novice users tended to have a high number of help requests, conceptual errors and definition requests when usage was logged over a three-month period. In this way, Beale and Finlay's system was able to distinguish between novices and experts with statistically significant differences based on an analysis of the trace or cognitive audit trail left by students as they interacted with the system. Woolf and Murray's (1992) minimalist approach to student modeling can also be seen as a mapping from observed student behaviour to hypotheses about the student's mental state. This is a difficult process that requires complex technology to be able to observe and make inferences about on-line learner behaviour. A trace facility that can automatically track student actions and feed this as input data to a neural network should be able to render the learner model dynamic. Such up-to-date contextual information can then be added to existing information on the learner and be used to make instructional decisions about that particular learner.

Other authors have begun to mention the possibility of using neural networks in sophisticated learner modeling applications (Holt *et al*, 1994) There have been some recent studies done on the use of AI-based pattern recognition to analyze learning activity in hypertext environments (Sun *et al*, 1995) in order to then adapt instructional interventions. These systems integrate a large amount of quantitative data (navigational sequence and elapsed time) into a dynamically updated student model.

Another group of researchers (Stacey *et al*, 1991) have been collaborating with the IBM Canada Usability Lab to conduct neural network analyses of human-computer interaction. Their goal was to test the usability of software systems by gathering objective data on time spent and options used within the software to detect any usage problems. The results obtained were quite useful; however, they had a very small data sample of only six users.

At the Praxis Technical Group (Millbank, 1996), neural networks have been used to monitor trainees' performance in simulated industrial systems. The data collected provides information on a given trainee's knowledge and ability. Based on this information, trainee suitability for the job is determined. Praxis makes use of this system to market a better way to screen job candidates than traditional resumes and interviews which tend to focus on educational background and more subject information about the applicant.

Harp *et al* (1995) described a novel application of neural networks to model the behavior of students in the context of an intelligent tutoring system. Self-organizing feature maps were used to capture the possible states of student knowledge from an existing test database. The trained network implements a universal student model that can be applied to rapidly assess the knowledge of any student and chart a path from lower to higher states of expertise. A Kohonen feature map was used in order to deal with incomplete inputs. Unlike multilayer perceptron neural networks, Kohonen feature maps are not virtually uninterpretable black boxes. A map unit represents a knowledge state and the weights associated with the unit indicate whether or not a student in that state is capable of correctly answering the problems.

Mukhodpadhyay *et al* (96) describe an adaptive information filtering system that makes use of learning agents for user profiling. The user profiling learning module consists of a learning agent that interacts directly with the user. This subsystem sorts incoming documents according to its belief of the preferences of a particular user. To accomplish this task, the learning agent maintains and updates a simplified model of the user. The user is modeled by means of an estimated relevance vector, which is used during the learning phase to sort the incoming information. On the basis of limited user feedback, the user's concept of relevance, the learning agent updates its estimates so as to improve its performance even while interacting with a less than certain user. The agent is assumed to have no *a priori* knowledge of the user preferences. This model has been applied to filtering internet messages and has shown very satisfactory results. The system is able to learn and adapt its filtering action to achieve the goal of delivering relevant information to the user with a high level of performance.

At the University of York, Finlay and Beale (1991) have explored the use of neural networks in the area of dynamic user modeling. User modeling is required in order to provide systems which can customize their response according to their knowledge of the user. Traditionally such modeling is performed using knowledge-based techniques, which although powerful, have associated problems of knowledge elicitation and domain dependence. Based on the observation that user modeling can be viewed as a classification problem in which users are placed in one of a number of predetermined categories depending on their behaviour, a neural network can be applied to the problem. An associative memory was trained on examples of user behaviour (trace data) and then used to classify users according to their interaction with the system. This approach can be applied to the static analysis of trace information (e.g., interface evaluation).

Beale and Finlay used a neural network to model a learner for an intelligent tutoring system. They interpreted user actions in order to provide appropriate responses. User patterns were extracte from traces of users' interactions and a neural network was used to classify them. This allowed for greater generalizability as well as the ability to handle noisy data. They used an ADAM network architecture (Advanced Distributed Associative Memory). The input vectors represented different classes of interactions and the output vector represented the best fit pattern. Features within an interaction, such as a command, type of input (example, review etc.) or type of error were used to classify the entire pattern. Statistical analysis showed the two groups classified by the network to be significantly different. These two groups could then be identified as novice or expert users. of the target system. Novices tended to have a higher number of help requests, of conceptual errors and definition requests. Usage was logged over three months for four experts and four novice users. Tw different tasks were used: learning how to use a UNIX system and learning how to use an e-mail system. A given pattern represented over 100 interactions per user. The ADAM network was trained using these as examples. 176 traces were abstracted and presented to the network for classification. Four runs were performed for each pattern, using a different randomly chosen example set. A cardinality of eight was used to classify the traces. The network correctly recognized 56% to 87% of users and tasks.

Trace data refers to the physical evidence that is left behind as an unintended artifact of human-computer interaction. The advantage is that it is unobtrusive and therefore does not interfere with ongoing behaviour and flow of execution. It is not likely to be affected by participants' awareness of the research. On the other hand, trace data is not very versatile and it may not be available for concepts to be studied. It is often quite loosely associated with the concepts it is alleged to represent It is time consuming to gather and process trace data, especially if there is a large amount of data that may be discarded. This technique has not been extensively used in the behavioral and social sciences.

Trace data yields a keystroke-level model. Individual keystrokes, mouse clicks and other moto activities are analyzed in human-computer interaction dialogues. This is an empirical way to build a cognitive model of the task at hand and to assess the time required for experts and novices to perforn the task.

Summary

Sandberg (94) discusses two major theoretical stands concerning the nature of cognition. At one end of the spectrum we find Anderson, Ohlsson, Elshout and VanLehn and others who conside the individual agent as the unit of analysis, irrespective of the environment. At the other end we find those that argue that it is impossible to perceive the individual in isolation, the individual is an inseparable part of the larger environment: for example, Collins, Schank and Pontecorvo. The individualists are mainly interested in fundamental questions concerning the nature of cognition as it is manifest in individuals. The contextualists share an interest in cognition in a much broader sense: hov is (individual) cognition embedded in a larger context?

With regards to student modeling, a trace facility may just record exam results but it could also produce a detailed learner model in real-time during the learning process. The learner model represents aspects of the learner relevant for the learning process e.g., learners' misconceptions. Th cognitive state of a learner may also be modeled as well as the transitions between states. Most researchers, he states, no longer believe in real-time student modeling, although for differing reasons The individualists explicitly state that it is not a very profitable endeavor. Anderson claims that detailed knowledge of students' errors and misconceptions forms no basis for better instructional decisions. The contextualists simply consider real-time student modeling to be too difficult.

Sandberg maintains that it is not necessary for learner modeling to take place during the running of an educational program. VanLehn emphasizes the need for good computer models of learning. Others favor the construction of artificial companion learners. Such machine-learning programs can be elaborated to make simulated students who can then be trained and used to try out different types of instruction. Anderson expressed an interest in having a simulation program of a particular learning process which could interact with one of his tutoring programs to see whether it would learn from the tutor. Student modeling is thus no longer seen as a support function for learners but as a method facilitating the design of better learning materials and environments.

Trace analysis is very important within the contextualist perspective. To be able to teach at critical events means observing closely what is happening. Modeling the learner in a context in order to study what is happening will require new methodologies. Assessment of learners is a new area of interest for contextualists.

Although the field of education and technology seems more divided than ever, theoretically and practically, it is Sandberg's opinion that we should avoid the emergence of two strands of researc that do not talk to each other. We should aim for a clarification of the implications of the stated differences (Sandberg, 94). The following section outlines how the systems approach to learner modeling undertaken in this study represents an attempt to unify learner modeling research from thre disparate fields of study: educational psychology, cognitive science and artificial intelligence.

Learner Modeling Approach

A truly adaptive learning environment would be one where all its components could adapt dynamically to the student by taking into account personal factors, cognitive styles, strategies and prior knowledge (van der Veer, 1990) as well as the learner's mental model of the adaptive system (Gentner and Stevens, 1983). The relative usefulness of each source of information about a learner will vary. Thus their evaluation will necessarily always be relative or contextual. However, the objective of this study is not to compare one with the other. Rather, the objective is to undertake a holistic approach to learner modeling, one in which all possible sources of information about a learner are integrated. The development of such a general framework for learner modeling will make possible synergy between the variety of literature addressing this problem.

This research represents preliminary work to be done in addressing the question of *adaptive* user modeling. No intelligent communication can take place without some understanding of the recipient of the information (Clemson, 1984). The adaptability of any system is largely determined by the coverage and accuracy of the information contained in the user model. Ideally, this model should include all aspects of the user's behaviour and knowledge. This is not an easy task for humans *let alone* machines which have a much more constrained communication channel. Craik's (1943) fundamental contribution to cybernetics was to point out that a machine could best interact with its surroundings if it could use an internalized abstraction of relevant aspects of its environment.

Learner Modeling in Ergonomics

The importance of behavioral analysis and empirical evaluation in the design of computer systems has been recognized widely in the field of human-computer interaction or ergonomics (Baecker *et al.*, 1995). Ergonomic analysis explores usage patterns and uncovers where interface design was successful and where it was flawed. Trace code is inserted into the prototype interface in order to unobtrusively and anonymously collect data about users' actions. This field aims to assess trace data as the system is being used in order to provide users with responsive intelligent defaults tha adjust themselves as user populations and needs change.

This type of user model is referred to as a keystroke-level model. Mouse movements and other motor activities are monitored in a human-computer dialogue. This represents an empirical way of building a cognitive model of the task at hand or for assessing the time required for experts and novices to perform given tasks. For example, Egan (1988) reviewed a variety of ways in which individuals differ in their use of computers. He found that people clearly differed in their rate of learning, speed of retrieval and reasoning strategy used.

From the ergonomic perspective, user behaviour is seen as a function of the task requirement, the procedural knowledge , the content and the dynamic control elements of a computer system. Common patterns of mental activity are assumed to give rise to similar patterns of user behaviour. Much of interface design is built around the premise of the 'typical user.' The problem is that there is no such thing - both tasks and users vary with respect to knowledge, and they both change over time. People want systems that conform to their preferences at the individual, group and organizational levels. One response to this has been in the design and development of intelligent interfaces. These are more active because they take the initiative to adapt the interface or its interaction model to fit the perceived needs of each user. Early systems represented anticipated user behaviours on the basis of statistical averages. Later systems constructed dynamic or adaptive models of the individual users an task domains (Benyon and Murray, 1993).

Statistics-based interfaces dynamically reconfigure menu hierarchies (e.g., place paste after cut). They define probability distributions based on access frequency and recency of selection. Eac act of selection alters the distribution therefore the interface is dynamic and evolves with time (Greenberg *et al*, 1994).

Adaptive interfaces incorporate user modeling instead of statistics and are to be found in intelligent tutoring systems and coaching systems. The goal(s) of the interaction are known (concept to be learned) by the means to reach them are not. We can therefore expect to model students, to customize the curriculum and to make suggestions. Intelligent tutoring systems are a variation of acti help systems: they detect and flag user inefficiencies, compare these to buggy or erroneous models interrupt and advise without being annoying.

The problem is trying to do too much with too little: system inferences made by monitoring user interactions are typically information-poor and often contain errors. The system and the user are both trying to model each other at the same time which means both are in a state of flux and neither ca reach a stable state (a phenomenon known as hunting). Users may feel a lack of control when the system takes over. Incorrect assumptions or modeling may have serious consequences. The field of learner modeling has evolved, on the one hand, from strictly domain-based models which view the learner as a faulty and incomplete subset of the domain expert and strictly learner-based models whic endow the learner with fixed domain-independent learning styles to, on the other hand, a more dynamic and more holistic learner modeling framework, one which views the learner in terms of learne characteristics and learner behaviours. Among the trends predicted for the next few years is greater customization of learning based on user profiling of user needs and how well students progress through learning materials (Wohl, 1996).

Self (1987) was one of the first to point out the potential of applying machine learning to student modeling. He felt that the availability of large databases of information that a student could browse through at will (today's hypertext environments) would provide a wealth of information about the on-line learning behaviour of a student. This coupled with new technologies to monitor a student's exploration of learning will enable us to design learning environments that can intelligently intervene to enhance learning.

Livergood (1991) points out the value of incorporating learning theory, learning outcomes and effectiveness of different kinds of materials in leading to the development of skills in the learner. He attributes this to the fact that most ITS were developed by persons with little interest in instructional efficiency. He recommends that user models take into account the specific learning style of the learner from his input and his past record of achievement within the systems.

Shute (1996) states that individual differences in learning are due to a diverse number of cognitive aptitudes such as reasoning, creativity, personality style, motivation, working memory, and learning styles. In order to individualize instruction by modeling the learner. We need to know not only about emerging learner skills and knowledge but also the incoming aptitudes possessed by learners.

In early experiments, Shute (1989) found that individuals differed in their propensity to ask for help, hints and advice. She used ITS applications as controlled instructional environments in order to learn more about learning criteria measures including behavioral measures. Learning efficiency is thought to be mediated by memory, processing speed, prior knowledge and skill. The main research question she addressed was how do learner characteristics relate to learning behavior, efficiency and outcome. She kept statistics on the frequency of use of the help function. This type of system provides an excellent ATI research testbed as different instructional treatments can be used with the same learners.

More recently, Shute (1996) described two different types of approaches to student modeling: a microadaptive approach and a macroadaptive approach. The microadaptive approach is the standard approach that represents emerging knowledge and skills. The computer responds to updated observations with a modified curriculum that is minutely adjusted, dependent on individual response histories during tutoring sessions. This is the more prevalent form that focuses primarily on domain-specific knowledge. The macroadaptive approach is an alternative approach that involves assessing students prior to their use of the tutor, and focuses mainly on general, long-term aptitudes such as working memory capacity, inductive reasoning skill and impulsivity etc. Combining these two approaches enables the system to adapt to both momentary and persistent performance information - to domain-specific knowledge and general aptitudes.

The proposed systems approach to user modeling therefore adopts a macroadaptive approach to test the predictive value of a subset of key variables from educational psychology, cognitive science and artificial intelligence learner modeling perspectives. There is a continuum to the degree of intelligence that can be integrated into performance evaluation aids. At the low end, manual data trapping and analysis can be done through preprogrammed routines. This can be augmented by regression analysis tools and adaptive testing capabilities to apply more sophisticated statistical modeling and question selection techniques in the measurement of student performance. At the most intelligent end, there can be automated pattern recognition, with embedded pattern recognition capabilities to continually note tendencies in student performance. The latter can form the basis for extremely responsive instructional strategies (Allen and Szabo, 1990).

Most models of learning in the field of computer learning are considered to be psychologically invalid (Bierman*et al.*, 1991). It is astonishing that hardly any relevant empirical research exists that tries to evaluate the extra value of a dynamic student model. Most experiments do not compare the dynamic model with a static model, but rather with no model at all. Furthermore, they do not control for task. There is ample evidence that computer assisted education is more effective than some classroom teaching. However, when comparing computer-assisted teaching with individual human coaching, it appears that the human is more effective than the computer. (Bloom, 1984) In the first place, a human coach is not limited by a restricted bandwidth. Secondly, the coach may display more affective states such as enthusiasm and they are armed with a much wider repertoire of possible remedies for possible learning problems(Laurillard, 1988). There appears to be a general feeling of impasse in the field of ITS. Some claim that radically different architectures are needed (Self, 1990). Others try to reconcile the field with more open-ended learning environments (Elsom-Cook, 1990 and others). Bierman (1993) suspects that one of the underlying reasons for this feeling is the implicit recognition that research on complete and truthful cognitive student models, one of the key components in traditional ITS architectures, has shown hardly any progress and that serious doubt exists whether this progress can be expected in the short term or at all.

Research indicates that human tutors do not invest much in building a model of their student, unless it is for students with evident learning problems. This is likely due to the fact that they already have a set of micro stereotypes available for quick matches. These micro stereotypes were likely accumulated through experience and relate to known learning challenges present in particular learning tasks. In fact, deep student modeling and error/misconception diagnosis may be required only for this and other special subgroups of learners (Bierman, 1993). The answer seems to be to forget about detailed cognitive student models, at least for the time being, and to focus on a global classification of the student in a number of subgroups. This is not only similar to the behaviour of human tutors but it also appears to be the most successful way to model users of non-educational systems. These models do not pretend cognitive validity but they appear to be useful for the task at hand. Bierman refers to these as less intelligent tutoring systems (LITS) which use rough classification, and proper localized diagnosis and adaptive feedback .

One of the ultimate aims of intelligent computerized learning environments is to constantly adapt to the students needs and to keep them at the edge of their learning frontier - that is, beyond what they already know but not too far beyond that the knowledge becomes unintelligible. (Duchastel, 1992). This is quite akin to Vygotsky's zone of proximal development. In conceptual terms, the material must be new but nevertheless fit within a familiar cognitive structure and thus remain meaningful. One of the best uses of an interpreted student trace, or student model, may be to provoke the student continually into this zone of optimal learning. Some studies have begun in this area, for example, Gegg-Harrison (1992) who has developed an ITS for Prolog tutoring that measures the student's knowledge zone in order to provide instruction that is truly adapted to the capability level of the student.

CHAPTER 3. RESEARCH DESIGN

Scope of Study

This work represented a preliminary look at and analysis of characteristics of learners that appear to be useful parameters when included in a learner model. It is hoped that the subset of learner variables which emerged from this analysis will guide the development of adaptive learning systems which are better able to match learner characteristics with flexible pedagogical modeling techniques. Although the present analysis necessarily confined itself to a select few manifestations of variables assumed to have an impact on learning, the long term goal of this line of research would be to create systems which can accommodate different course content, different learner populations, and different instructional designs. Finally, a new methodology was developed, tested and implemented whereby neural networks were used to assess and identify categories of prototype learner behavior and juxtapose these results with the categorization process of several learning style inventories and actual learner performance.

Two distinct studies were conducted: the first study (Study I) represented a pilot study whose purpose was to evaluate formatively the materials used (software, questionnaires, tests, instructional design and course content). This extensive pilot study served to establish a sound foundation for the collection of data to be used in the second study. The second study (Study II) generated the data used in the analysis of results.

Study I

Target Population

The accessible population of candidates included students and professionals (such as computer scientists, project managers and software engineers) interested in learning about the field of neural networks The neural network course content was designed such that, in a slightly modified form, it could be used by the general public wishing to learn more about the field, or for people considering a career in knowledge engineering (e.g., as a career counseling tool, an aptitude-testing tool, or to determine whether neural networks would be an appropriate technology to use for a given problem). Thus, no prior knowledge of the field was assumed.

Sample

The first study consisted of two groups: one made up of UQAM students registered for Expert Systems and Decision Support Systems courses in the Bachelor's program in MIS, and a second group was made up of volunteers from a research centre, CWARC, which has since been renamed CITI (Centre for Information Technology Innovation). The total sample size for Study I was 91 students (refer to Table 2 below).

Table 2. Study I Sample Groups

Session	Sample size	Date of experimentation
UQAM students	69	October 1992
CITI group	22	September 1993
Total	91	

Materials

Learning environment software (HIT)

A computer-based neural networks training course was developed for use in this study. The training material was designed and developed using HyperCard. Programmers were hired to develop the necessary software environment. Formative evaluation of course materials was carried out with the collaboration of subject matter experts as well as through field tests with trial user groups. The learning lab was run on a Macintosh microcomputer platform.

The computerized format is particularly appropriate as this is a commonly used format in most self-paced learning contexts. In addition, a computerized delivery system provided an automatic trace of student interactions with the learning materials. The latter is needed for use as data to be analyzed (see Procedures section for details). The HIT hypertext tutorial system and the user modeling components were run concurrently (i.e., user model acquisition was largely performed while the user is reading the current text object in order not to interfere with student interaction with the learning materials).

The software environment underwent a number of changes based on the results of beta testing with the participants of the pilot study (Study I). The first version (HIT 1.0) was operational in October 1991. This was a rule-based approach to trace analysis and was developed using the software shell Nexpert. Much greater flexibility was required so a neural network was incorporated into the rule-based system. MacBrain was initially used as the development tool but was found to be too limited. There was no access to the source code, it was not possible to make an unlimited number of run-time copies without incurring significant costs and the network algorithms provided with the tool were not entirely appropriate to the analysis required. Unsupervised learning is required and the tool provided only backward propagation algorithms. As a result, it was decided to program the entire environment in a neural network developed from scratch.

Version 2.0 was operational and 85% debugged in April 1992. The software was redesigned using object oriented methodologies. However, a problem was encountered with local minima occurring. The network would group students when sufficient data were gathered to optimize locally; this is a common problem encountered in optimization techniques and the major shortcoming is that a global min/max may never be detected when this occurs. This also prevented the network from storing all the data that were generated during the course of students interacting with the software environment.

In August 1992, an induction table was added to the neural network. This simulated rule-based reasoning and allowed the resolution of the local minimum problem. The systems operated at 99% robustness as system errors were quite rare. Hypertext was added to the general text function (words in parentheses were hot buttons that could be clicked on in order to obtain a definition). A hypertext glossary was also made available on-line (words within the definition could be clicked on to lead to other cross references). The HIT learning environment was segmented into six distinct modules: a job aid or browser (allows users to go through course materials at a high level, without any testing of knowledge acquired), a computer-based tutorial on neural networks, a course authoring module (to modify the existing neural network course or to input new courses), a neural network editor (to allow users to design and develop their own neural networks), the trace analyzer (to collect and analyze temporal and keystroke data generated during interactions) and a statistical analysis module (to conduct factor analysis, principal component analysis, discriminant analysis and canonical correlation on data generated). Version 4.0 was used for the data collection in Study II. This version was operational in January 1993 and included color version, PC version, Macintosh PowerBook, Quadra and PowerPC versions.

Student Trace Facility

An extensive record keeping function was implemented in order to monitor how the student interacts with the training materials. A sample of such data items included: temporal data (time to complete training, to complete a particular lesson, a test), sequence data (initial topic chosen, sequence of menu option selected for a given topic: example, test then review) and evaluation of progress (self-evaluation, comprehension tests, number of lessons remediated).

It is expected that each on-line monitoring of student learning will yield a large number of data items for each learner. The Beale and Finlay (1989) study found over 700 items for each student. A neural network, together with other statistical analysis tools (Statistica for the Macintosh, SPSS for the PC), was implemented. The data items (or some subset of them) served as inputs to the neural network which then identified distinct usage patterns of students. This provided an alternative means of assessing student cognitive styles, or of verifying style assessments based on self-report learning style instruments. The neural network can thereby help to check whether the self-reported categories represented actual clusters of behaviours.

The neural network placed individuals into categories based on common tactics used in interacting with the software (trace variables). The Kohonen neural network architecture was used (see Appendix E). In order to address the ordering bias problem with this architecture, input order was varied until groups obtained were stable at a level over 90%. A self-monitoring function that does the group stabilization procedure automatically was programmed into the network itself.

Pedagogical Model

The training courseware allowed the learner the choice of selecting their own path through the learning materials (maximum learner control) or of allowing the computer to guide them (system-guided learning). This in turn enabled the investigation of learner control as a potential critical factor in learning pattern differences. For example, the initial screen displayed approximately five possible objectives or lessons the learner could choose from, or they could elect to let the computer "suggest" a topic for them. Within a given lesson, the student could choose to do a number of things: see an example, try an exercise, look up a term in the hypertext glossary, ask for help, ask for additional information, ask to review previously studied materials or change topics.

The instructional design was modified, based on comments from professors, other subject matter experts and 'novices' involved in the beta testing of the environment. Quiz questions were added at the end of each of the five sections. This replaced the auto-evaluation in the previous version (where participants were asked to rate their own understanding of the material on a scale of one to five). There were a maximum of three quiz questions for each section and it was optional (students could answer as many as three or none at all). More illustrative examples were added to each of the five sections, in the form of sample designs, applications, diagrams and facts and figures. An "elab_info" button was added for those students seeking supplementary information on topics of interest as well as references to additional readings.

Students were also given the choice to be guided in their choice of lessons or to choose from a lesson menu. The guidance option was available throughout the interaction (e.g., some students chose to be guided and remained in guided mode throughout, others were guided for a few lessons, then chose their own lessons).

A 'defer posttest' option was added, to allow students the choice of deferring the posttest and going back to any or all of the five lessons before taking the posttest. More sophisticated help features were added based on questions that were asked of technical assistants during Study I testing. In addition, if time permitted, students were also given the chance to apply what they had learned. Students could choose to develop a neural network using a neural network editor that is provided with the learning lab.

Subject Matter

The course content consisted of both the theoretical basis for artificial intelligence systems as well as a more practical orientation. The IAKE Handbook of Theory and Practice (IAKE, 90), as well as the IAKE Knowledge Engineering Certification Exam provided some of this material. IAKE (International Association of Knowledge Engineers) is an organization that offers professional certification in knowledge engineering. In addition, a number of widely used reference books in the field were used to complete the theoretical content. The practice-oriented content was based on a task analysis of professional knowledge engineers that was done as part of this study as well as an analysis of learning difficulties experienced by novices. UQAM professors validated and contributed to the course materials to be used.

Tests

The rather low scores of the pilot group on both the pretest and posttest indicated that learners did not have the prerequisite knowledge it was assumed they had. The course content was thus modified, particularly the first lesson (Introduction), to introduce better concepts and definitions. A short pretest of 10 multiple-choice items and five short answer items was developed and used to assess students' prior knowledge of the subject matter. This contained some items found on the IAKE certification exam. A posttest of similar structure was developed and used to assess whether or not learning objectives had been met upon completion of the training. Items for the pretest and posttest were selected from a pool of 20 multiple choice and 10 short answer questions using a stratified random sampling approach in order to ensure equitable distribution of questions across lesson modules.

Item analyses were conducted in order to improve upon these tests. The first version of the pretest was 12 multiple choice questions that proved to be too general (about the field of artificial intelligence in general). The initial version of the posttest consisted of six short answer questions which proved to be too difficult for the majority of students. As a result, questions which were not directly relevant to the course content, correlated questions and questions that were too difficult were eliminated. The second version included a good range of item difficulty. All items were matched to specific lesson objectives in each of the five lesson modules and no floor or ceiling effects were detected.

Differences in pretest and posttest scores were used as a measure of learning. Test scoring was: five points for each multiple choice question and five points for each short answer question. An answer grid was developed for the short answer questions in order to minimize subjectivity in scoring student answers. In addition, two different individuals were used to score the short answer questions. Major discrepancies in scores were discussed amongst the raters in order to determine if differences were due to a bias or different interpretation of the answer grid. An inter-rater correlation of 0.94 was obtained for the pretest, and 0.91 for the posttest.

Tests of homogeneity (Chi-Square and Kolmogorov-Smirnoff) proved reasonable for the distribution of both pretest and posttest scores (for both the multiple choice, short answer and combined answer sets). Content validity (face validity) is expected to exist for these tests as questions are derived directly from course content and both professors and subject matter experts have corroborated on the items. Finally, the multiple choice questions were used to test for recognition and the short answer items for recall.

Questionnaire

A biographical questionnaire was developed and administered in order to obtain data on students' educational background (e.g., undergraduate major), work experience (e.g., number of expert systems developed), and demographic variables (such as age, sex, first official language). Answers were then grouped together to form coded categories (refer to Appendix C).

Learning Style Inventories

Two learning style instruments were used to categorize students' learning styles: the Kolb Learning Style Inventory (1976, 12-item version) and the Entwistle Approaches to Studying Instrument (1981). The Kolb instrument was chosen as one that is representative of the educational psychology approach in that it purports to measure stable individual predispositions to learning. The Entwistle instrument was chosen as one that is representative of the cognitive science perspective in that it purports to measure contextual approaches students adopt to learning. Computerized versions were developed and administered so as to reduce the amount of time required to complete them. Esichailkul, 1994, found that a computerized version of the Kolb LSI required four minutes to complete.

These two instruments yielded a variety of categories and other data for all students. A number of expected on-line learning behaviours were derived from the literature on both the Kolb LSI and the Entwistle ASI. Essentially, this involved an inferential mapping process, extrapolating from the characteristics identified by Kolb and Entwistle as being typical of the different learner types, as identified by the instruments, and the types of learning choices and other patterns of learning behaviours that would then be expected of each of these learner types, within this particular learning context. These expected feature usage learner profiles, together with specific research hypotheses to be tested, are summarized below.

Kolb expected learner profiles. Individuals scoring high on the RO dimension are expected to make greater and more frequent use of the following options: note-taking, self testing and condense information options. The notes option allows them to reflect upon their experience while the self-test option allows them to observe their learning progress. The condense information option includes the outline and summary choices which help provide a higher level structure to the learning process. These students are also expected to spend more time in general, and in each of the options they select, which should be reflected in a greater time on task variable value.

Individuals high on the CE dimension are expected to make greater and more frequent use of the examples and self-test options. The examples option allows them to actively test their own knowledge throughout the learning process. They would be expected to spend less time on each of the options they select and in general.

Individuals scoring high on the AE dimension are expected to make greater and more frequent use of the elaborate information and hypertext options. The elaborate information option represents examples and definitions, which allow students to be very active during the learning process. The hypertext option similarly allows students to play a more active role in the learning environment. The learning environment features did not accomodate actual experimentation learning behaviours, other than in the form of an optional ANN editor.

Individuals scoring high on the AC dimension are expected to make greater and more frequent use of the condense information and general text options. These options contain mostly theoretical course content. Students scoring high on the AC-CE dimension are expected to make greater and more frequent use of the examples option, while those individuals with a high AE-RO score should make lesser and less frequent use of the examples option. Finally, based on some of the results found in the literature (refer to literature review chapter, page 34), convergers are expected to perform best in computer-based learning environments. These students would therefore be expected to have the highest posttest scores. They would also be expected to prefer system control of the lesson sequence.

<u>Entwistle expected learner profiles</u>. As the four Entwistle quadrants were created for the purposes of this study, no information exists in the literature to enable specific expectations about learner behavior to be derived. However, as these groupings are based

on only two of the eight Entwistle dimensions (reproducing and comprehension orientations), expected profiles were generated and analyzed for the dimensions.

Students with high reproducing orientation scores are expected to make greater and more frequent use of the self-test and defer-posttest options. This reflects their emphasis on being able to learn in order to do well on test materials. They are also expected to make greater and more frequent use of the dictionary and definition options, as these students tend to concentrate on being able to reproduce (i.e., memorize) information. They would be expected to spend less time in learning, overall and in each of the options selected, as this particular approach to studying tends to represent surface learning. Students with this profile would also be expected to show a breadth-first search strategy as they would attempt to maximize coverage of all topics rapidly, in order to be better able to reproduce most of the content on the posttest. These students would be expected to prefer system control of the lesson sequence.

Students with a comprehension orientation to studying would be expected to make greater and more frequent use of the elaborate information and note-taking options. Learners who score high on this dimension should spend more time interacting with the materials and a depth-first search strategy in lesson selection. These students would be expected to prefer to control the lesson sequence themselves. As this study focused on these two particular Entwistle dimensions, no expected profiles were generated for the remaining six orientations to studying.

Procedures

Two UQAM professors teaching decision support system courses were contacted to select classes for the UQAM study. Participants for the CITI study were selected from all those who replied to an e-mail invitation that was sent out to all employees. All participation was on a voluntary basis. Those who requested their results received a package explaining what their learning style profiles showed. University students benefited from exposure to course content that would be included in their final course exam. Students who did not wish to participate in the experiment itself were still given the opportunity to interact with the learning materials; their trace data were discarded at the end of the session.

The experimental sessions were held in a computer lab equipped with Macintosh computers under the direction of myself, the professor and lab assistants. The lab was reserved for the blocks of 4-hour periods in order to minimize any external distractions to the students. Class sizes ranged from about 20 to about 40 students. Classes had to be divided into two in those cases where the number of students exceeded the number of computers available in the lab (maximum of 27 at any given time).

All students were given the same five-module unit on knowledge engineering: Neural Networks. This course material was studied in open-ended (i.e., no time limit) sessions, with teaching assistants present to help in either technical or course-related problems. Students each had their individual microcomputer and participated in the experimentation at the computer lab room for the UQAM students or in a self-contained demo room for the CITI group.

All students were administered the learning style inventories and questionnaires one week before the experimentation session in order to maximize the time spent interacting with the course materials. Each participant was then asked to sign up for a particular time slot.

At the beginning of the experimental session, each group was given an orientation by the professor (what was required of them, assurance of anonymity of results, introduction of myself and assistants) as well as a brief introduction to the system (this was read from prepared notes so as to ensure uniform information to all participants). A diagram was provided on the board at the front of the computer lab in order to serve as an aide-memoire for system features (see Appendix A). Subjects were told they would be allowed as much time as they required since time spent learning will be one of the variables to be investigated by this study. Any questions posed by participants were answered.

Participants were then asked to complete the pretest. These were collected as each student finished and were later scored and analyzed. Students were then asked to begin the training session. Teaching assistants were on hand to help the students should they have any questions concerning system use. These questions were noted down and later analyzed. In addition, observers in the lab took notes on any behaviours which could affect the trace data (such as computer malfunctions, talking between students, having to leave before completing the session). These notes were later compiled, cross-referenced to the trace data and analyzed.

At the end of the training sessions, which students completed at their own pace, a posttest was administered to assess the results of the training. Some students were randomly selected for exit interviews. These interviews followed a set of structured questions and were tape recorded. Questions were asked concerning their experience of learning in this fashion, whether they felt they could study as they normally would, whether they found the experience useful and any other general comments they had. These interviews were later transcribed and analyzed (Appendix D). A summary of the results of Study I may be found in Appendix B.

Study II

The second study (Study II) consisted of six sessions with UQAM students to gather the actual data used for this study. Subjects were taken from a number of sessions of a UQAM course on Management Information Systems over the course of a three-year period (see Table 2). Participants received the results of their learning style assessments within approximately two weeks (if they requested these) as well as five extra course credits for participation in the study. This may provide a bias toward an achievement orientation to studying; however, the achievement orientation dimension of the Entwistle ASI, while collected as data, was not addressed in the primary research questions of this study. In addition, the content covered by the software was included on their final exam. All students followed the same course with the same professor.

The data collection was conducted at roughly the same phase in the course (after the mid-term examination had been given) in order to give the professor enough time to establish a rapport with the students and cover appropriate preparatory information. Procedures and materials developed and validated in the first study were used for all six experimentation sessions (see Table 3 below). Nine participants were eliminated from the study. Of these, four did not complete the pretests and/or posttest and the remaining five did not show required interaction patterns with the software (they spent less than 10 minutes on the course and/or did not complete all five lesson modules).

Data collected was analyzed in its entirety. In addition, the first lesson visited by each student was omitted and the data re-analyzed, in order to test and possibly omit any learner behaviours related to a 'novelty effect' (i.e., explorations of all options present in the interface due to curiosity and learning curve associated with becoming familiar with the sofware).

Table 3. Study II Sample Groups

Session	Sample size	Date of experimentation
1	26	March 1993
2	18	August 1993
3	19	November 1993
4	20	March 1994
5	45	June 1994
6	43	September 1994
Total	171	

There was evidence of such a novelty effect in the pilot study. Students exhibited higher initial interactivity levels and eventually showed more "stable" patterns. Each student had a different length of time associated with such an effect. As a result, it was decided to omit the first lesson visited by each student in its entirety and to run these data in addition to the entire lesson set data. The novelty effect is likely associated with the novelty of the learning environment (interface) rather than the course content. Students appeared to be familiarizing themselves with all the available options before focusing more on the content of the lessons.

CHAPTER 4. RESULTS

Introduction

This section will first present the results of the pilot study, Study I. This study was used for formative evaluation purposes. Next, the study sessions used to collect the experimental data, Study II, will be discussed. This analysis first addresses descriptive statistics and demographic data obtained on the sample data used. Next, the variables extracted from each of the three disciplines contributing to user modeling will be presented.

The cognitive science variables will be presented to assess their contribution to explaining variance in the two primary measures (posttest score and time on task).

The learning style variables from educational psychology were assessed with respect to posttest and time on task variance. In addition, the expected learner behaviours (as extracted from the learning style constructs) were compared to actual learner behaviours in order to assess the local predictive value of these constructs. The groupings based on learning styles were also compared to on-line learning behaviours without any *a priori* hypotheses in order to ascertain whether individuals in different groups exhibited differences in learning behaviour. All these results are presented in Section 4.3.

In the final section, the groupings generated by both conventional cluster analysis and neural network classification will be presented with respect to their contribution to posttest and time on task variance, and to discuss differences in their learning behaviours. The neural network classifications will be discussed in order to assess the usefulness of neural networks to learner modeling.

Table 58, on page 148, presents a summary of the results obtained, along with their instructional implications, which are more fully addressed in the discussion section.

Study I

The first study was conducted from October 1992 to September 1993 and had a total sample size of 91 students. This sample consisted of a group of 69 university students and 22 researchers. These data were used in order to conduct formative evaluations of the materials to be used in Study II (refer to methods section for more detail).

Two sessions were conducted with two classes of university students. As the number of computers available in the computer laboratory was limited to 24, each class was divided into two separate testing sessions. Students were asked to sign up for a time. Of all the students registered in the two classes, only one student did not show up. No student elected to opt out of the study - that is to say, no one completed the experimentation and asked that their data set not be used. The average session length was 37 minutes. In this group, all students completed the materials and therefore all the data generated were used in the formative evaluation. Randomly selected students were interviewed after the sessions in order to assess how well the environment was able to accommodate their approaches to learning and studying (refer to Appendix D).

The group of 22 researchers were asked to sign up for a session and each completed the materials within a two-week span of time in September 1993. All those who volunteered presented themselves for the experimentation and completed all the materials. The average session length was 68 minutes (refer to Table 4). The longer interaction time of this group was likely due to the fact that researchers were specifically asked to formatively evaluate the system whereas the university group was not asked to do so. This group provided more extensive comments on the actual materials used, how the study was conducted and how they felt both could be improved. All participants were interviewed after the experimentation session in order to further explore their reactions to the learning environment.

Teaching assistants were on hand to answer any questions raised by the students not related to the subject matter. These questions were noted and associated with the desk number of the student (and hence with their data sets). The majority of questions raised appeared to be related to how to use the interface (in one extreme case, a student had to be instructed on how to use a mouse). This led to the formulation of a diagram explaining how to navigate through the system which was placed at the front of the class on a white board and explained during the initial orientation talk for all Study II test sessions (refer to Appendix A).

There was a great deal of anxiety expressed over failure to answer questions on the pretest which led to the realization that students were completely unfamiliar with the concepts. An explanation was later incorporated into the orientation talk where students were reassured that they were not expected to be able to answer all of the pretest questions but that after exposure to the learning materials, they should be able to answer questions on the posttest.

Table 4. Average Time on Task for Study I Participants

Group	Mean	SD	Min	Max	N
students	36.7	24.5	0	102.192	68
researchers	67.6	13.4	19.4	366.3	22

Study II

As described in the Methods section, both data and observation of procedures gathered from Study I were used as a pilot study and formative evaluation to prepare Study II. Data obtained in Study II were then used in the testing of research hypotheses. The principal changes included: modifications in the software interface, in the instructional design, the course content and the pretest and posttests used for the actual study. The software used to deliver course materials and to gather data on learner behaviors was iteratively improved through pilot study testing. The neural network algorithms used were tested and refined in order to eliminate any bias due to data input order. The software interface was simplified and improved in order to minimize problems learning to navigate through the course materials as opposed to interacting with and learning the course content. The pedagogical design was modified through the addition of quiz or self-test questions at the end of each lesson module, an option to obtain supplementary information on any given topic, an option to defer taking the posttest and review the content, and the inclusion of more illustrative examples for each of the lesson topics. Finally, the pretest and posttest instruments were modified based on results of item analyses in order to provide a more accurate assessment of knowledge of the course content prior to and following interaction with the course materials. Appendix B contains the detailed results of Study I that were used to make these modifications.

The second study subjects consisted only of university students. Students were given five marks for participating in the exercise and were told that the content would appear on their final exams. Six separate sessions were conducted from March 1993 to September 1994. A total of 171 data sets were obtained. Of registered students, four did not show up for the sessions. Of those who came, six participants did not fill out the questionnaire, pretest and learning style instruments and two participants did not interact with the system long enough to produce a valid trace. Five students from each test session were randomly selected for an exit interview. Five students went through the session a second time, stating that there were topics they wished to revisit. Students spent an average of 35.44 minutes interacting with the computerized course modules (see Table 5). Students spent an average of 11.59 minutes in the first of the five lesson modules.

Table 5. Average Time on Task for Study II Participants

Group	Mean	SD	Min	Max	N
all lessons	35.4	17.3	0	81.3	171
lesson one	11.6	7.8	0	38.9	171

There were fewer questions asked in this study (no questions in three of the six sessions). Of those questions that were raised, two were due to a system failure (the machine spontaneously crashed or booted them out of the program) and they had to be restarted on another machine. Others were comments on the content or procedural questions (such as why after three wrong answers they couldn't continue trying to answer the same question).

Scoring Procedures

Questionnaire Data

A questionnaire consisting of multiple choice and open-ended answers was completed by each of the participants prior to experimentation. These questions elicited biographical and background information on subjects. Data gathered included age, sex, educational background, degree program enrolled in, career plan, previous experience with the course content matter, reason for participating in the study and whether or not they wished to receive the results of the learning style assessments.

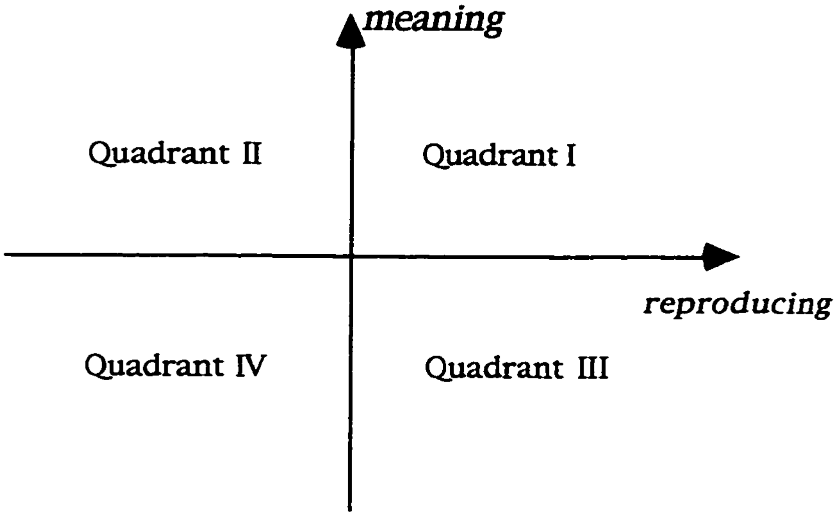
For those questions that were open-ended, category codes were derived in order to associate a numerical value to each category for subsequent statistical analyses. Answers were grouped according to their frequency with less frequent or unique answers being assigned to an "Other" category. These data were then used to generate descriptive statistics on the sample used (Appendix C presents the category codes used for the questionnaire).

<u>Learning Style Data</u>

The Kolb Learning Style inventory produces three types of data: groupings (converger, diverger, accomodator or assimilator), scores on the two dimensions (AE-RO and AC-CE) as well as scores on the individual constructs (CE, AE, RO, and AC). The scores on the constructs and dimensions were retained in their original form since they are numerical data. Kolb groups were obtained using the procedure outlined in the accompanying guide to using the LSI. The groupings were converted into category codes for statistical analysis purposes.

The Entwistle Approaches to Studying Instrument produces scores on eight dimensions (achieving, reproducing, meaning, comprehension, operation, versatility, pathology of learning and prediction of success). These scores were obtained using the procedure outlined by Entwistle in his guide to using the ASI. These data were retained as is. In addition, groupings similar to those generated by Kolb were created by using scores on two of the dimensions: reproducing and meaning (refer to Figure 6). Students were placed on the high or the low half of the dimension using an arbitrary threshold value of 12 (e.g., students in quadrant II had a meaning orientation dimension score that exceeded 12 and a reproducing orientation dimension score that was less than 12. These groups were created in order to be able to treat the Entwistle data to cluster analysis and thus directly compare them to the groups produced by the Kolb LSI instrument.

Figure 6. Entwistle Quadrants



Trace Data

Trace data were automatically collected and analyzed by tracking student behaviors while they were interacting with the learning environment. Keystroke data and temporal data were collected. Every choice made by the student in terms of which option they clicked on as well as how long they spent on each option was recorded. Students had a pause button available at all times which they made use of when they took a break or asked a question. This prevented the temporal trace data from misrepresenting the time on task variable.

The original twelve trace variables were grouped together into a new short list of eight variables according to a conceptual definition of the type of learning activity they represented and based on usage data obtained in the pilot study. This grouping was done before any data was collected for Study II. These new trace variable groups (refer to Table 6 below) were then used in the statistical analyses.

Table 6. New Trace Variable Groups

Name	Variables combined	Comment
pref		Guided vs. unguided mode
elab_info	dictionary, definitions + examples	Elaborate information
cond_info	outline + summary	Condense information
self-test	question + skip posttest	
gen_text		Lesson text content
hyp_text		Hypertext navigation
notes		Using notepad for notes
help		Requesting system help

The guidance preference (pref) variable refers to the preference shown by a student for guided vs. unguided interaction with the learning materials and was retained in the short list. The elaborate information category (elab_info) groups together those activities that lead to the elaboration of information for a given topic in a given lesson. This category includes the original trace variables of dictionary, definitions and examples. The condense information category (cond_info) groups together those variables which condense the information and includes the outline and summary options. The self-test group includes both the question and defer posttest options as they reflect activities geared towards improving achievement results in the learning environment. Other variables that were retained unchanged in the short list were: general text (gen_text), hypertext (hyp_text), notetaking (notes) and help options.

Demographics

The sample population represented a multi-ethnic mix of students, who were predominately francophone or native French speakers. Biographical information obtained from the questionnaire showed that the majority (69%) were male. The average student age was 26.92 with a standard deviation of 7.89. The youngest student was 18, the oldest 53 years old (see Table 7).

Educational background questions showed that most students had completed a Cegep degree (45%) and were currently enrolled in an MIS major program in the department of Administrative Sciences (75%). Their degree specialization was in four areas: computer science, MIS, human resources, and economics. The vast majority had some knowledge of artificial intelligence (93%) but very little, if any, familiarity with neural networks (13%).

Most student career plans included one of the following four options: programmer - analyst, continuing their education, becoming a manager, or starting their own business. Most students (92%) were interested in receiving the results of their learning style inventories.

Table 7. Demographic Variable Statistics

Variable					
Sex	69% male	29% female	2% no answer		
Degrees	45% CEGEP	32% CEGEP +certificates	13% CEGEP+ cert.+ Bach.	2% other	
Major	13% MIS	15% Science	22% comp. sci.	26%% admin	24% other
Program	7% Comp. Science	75% MIS	1% human resources	4% economics	
Career plans	26% programmer	14% continue studies	23% manager	9% start own business	9% Other
AI knowledge	42% a lot	51% some	5% little or none		
ANN knowledge	1% a lot	12% some	83% little or none		
requested results	92% yes	8% no			
Participation motivation	29% personal interest	34% required	12% personal interest & req'd	1% other	

Descriptive Statistics

Validity of the Learning Environment

The first research question addressed was: are the learning materials and learning environment used to collect the experimental data instructionally valid? Two variables, derived from the cognitive science literature, were measured: time on task (total amount of time spent interacting with the learning materials) and learning (measured as the difference in scores between the pretest and the posttest) were used to establish the validity of the learning environment. In particular, they were used to test whether or not significant learning had taken place, whether or not the different session groups were equivalent with respect to prior knowledge of content and whether there was a positive correlation between pretest score and time on task score with subsequent posttest scores.

A pretest and a posttest were given to all participants (see Appendix F). Each consisted of 10 multiple choice questions (Q1-Q10) and five short answer questions (Q11-Q15). Items were selected using stratified random sampling from a pool of available items. There were two items per lesson for each of the five lessons. Each multiple choice test item was worth two points, and each short answer was worth a maximum of five points, making the entire test scored out of a maximum possible score of 75. Raw data were used for all analyses.

Both pretests and posttests were scored by two different individuals. The short answer questions were scored according to an answer key that identified the number of points to be assigned to each element of a correct answer. Any significant differences in scoring were discussed in order to ensure that the disparity was not due to mistakes in using the answer key. An inter-rater correlation was calculated and found to be 0.94 for the pretest and 0.91 for the posttest.

Tests of homogeneity (Chi Square and Kolmogorov-Smirnoff tests) confirmed that the pretest and posttest scores had normal distributions. Content validity is established due to the fact that the items were created by subject matter experts and all questions are related to specific learning content.

An item analysis was carried out to calculate item discrimination, item difficulty and item reliability for both the pretest and posttests. Item difficulty was calculated as the total number of incorrect responses to a given multiple choice question divided by the number of students. Item difficulty for short answer questions was calculated as the one minus the average score on a given item. Tables 8 and 9 show the pretest and posttest item difficulty analyses below.

Table 8. Pretest Item Difficulty

Gp	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	Q11	Q12	Q13	Q14
1	0.2	0.6	0.7	0.9	0.4	0.6	0.5	0.6	0.5	0.5	0.8	0.6	0.9	0.9
2	0.3	0.6	0.6	0.7	0.1	0.6	0.6	0.5	0.2	0.6	0.9	0.9	0.9	0.8
3	0.5	0.6	0.5	0.8	0.4	0.5	0.6	0.7	0.5	0.6	0.9	0.9	0.9	0.9
4	0.2	0.8	0.7	0.8	0.5	0.4	0.3	0.6	0.6	0.6	0.9	0.9	0.9	0.9
5	0.3	0.6	0.6	0.7	0.4	0.7	0.4	0.7	0.5	0.3	1	0.9	1	0.9
6	0.1	0.7	0.7	0.8	0.6	0.6	0.5	0.6	0.5	0.6	1	1	1	0.9
TOT	0.3	0.6	0.6	0.8	0.4	0.6	0.5	0.6	0.5	0.5	0.9	0.9	0.9	0.9

Table 9. Posttest Item Difficulty

Gp	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	Q11	Q12	Q13	Q14
1	0.3	0.6	0.1	0.5	0.4	0.3	0.5	0.7	0.5	0.1	0.7	0.5	0.7	0.9
2	0.2	0.4	0	0.3	0.3	0.2	0.5	0.4	0.4	0	0.7	0.6	0.7	0.8
3	0.2	0.6	0.2	0.5	0.5	0.1	0.4	0.6	0.6	0.1	0.8	0.5	0.7	0.9
4	0.4	0.5	0.2	0.3	0.4	0.2	0.6	0.4	0.3	0.1	0.9	0.5	1	0.9
5	0.3	0.3	0.2	0.4	0.4	0.2	0.5	0.4	0.3	0	0.8	0.7	0.9	0.9
6	0.3	0.4	0.2	0.6	0.5	0.1	0.3	0.5	0.4	0.1	0.9	0.7	0.9	0.9
TOT	0.3	0.4	0.2	0.5	0.4	0.2	0.4	0.5	0.4	0.1	0.8	0.6	0.8	0.9

Item discrimination was calculated only for the multiple choice items. G 1- 6 represent the six sessions and TOT represents all participants. The top and the bottom 10% scorers were identified and the proportion of correct-answering low scorers was subtracted from the proportion of correct-answering high scorers. A discrimination index of 1.0 represents the best discrimination (see Tables 10 and 11 below).

Table 10. Pretest Item Discrimination

Gp	Q 1	Q 2	Q 3	Q 4	Q 5	Q 6	Q 7	Q 8	Q 9	Q 10
1	0.6	0.1	0.5	0.2	0.6	0.4	0.9	0.3	0.6	0.3
2	0.3	0.6	0.3	0.3	0.3	0.1	0.7	0.8	0.6	0.6
3	0.4	0.2	0.5	0	1	0.5	0.5	0.8	0.5	0.2
4	0.2	0.5	0.8	0.3	0.8	0.2	0.5	0.6	0.7	0.5
5	0.7	0.2	0.8	0.6	0.7	0.5	0.6	0.5	0.6	0.8
6	0	0.4	0.6	0.5	0.8	0.6	0.2	0.6	0.5	0.7
TOT	0.7	0.6	0.9	0.6	0.8	0.8	0.7	0.7	0.8	0.7

Results showed that there was an acceptable level of discrimination for all items and an acceptable range of difficulty for both the pretest and posttest questions.

Table 11. Posttest Item Discrimination

Gp	Q 1	Q 2	Q 3	Q 4	Q 5	Q 6	Q 7	Q 8	Q 9	Q 10
1	0.5	0.5	0.3	0.4	0.5	0.5	0.4	0.5	1	0
2	0.3	0.5	0	0.3	0.5	0.4	0	0.6	1	0
3	0.4	0.7	0.6	0.2	0.9	0	0.3	0	0.1	0
4	0.6	0.8	0.2	0.6	0.2	-0.2	1	0.6	1	0
5	0.9	0.6	0.5	0.3	0.5	0.4	0.6	0.9	0.6	0
6	0.2	0.7	0.5	0	0.3	0.3	0.7	0.6	0.7	0
TOT	0.6	0.7	0.4	0.5	0.8	0.4	0.8	0.6	0.9	0.2

Next, the different experimental groups were assessed in order to make sure that there were no significant differences across the different test sessions and that the data obtained could thus be grouped together for analysis. Group equivalence of the different experimental sessions was tested with respect to pretest scores using a one-way ANOVA. The results are shown in Tables 12 and 13 below.

At a significance level of $p > 0.05$, there were no significant differences between different session groups for pretest multiple choice questions (Pre_MC) and for overall scores (Pre_Tot). The first group had a higher score on the short answer questions (Pre_SA). There were no significant differences for posttest multiple choice scores (Post_MC) and for overall posttest scores (Post_Tot). However, groups 1 and 2 scored higher on the short answer questions (Post_SA).

Table 12. ANOVA for Test Session Equivalencies

Dep. variable	MS Effect	MS Error	F(df1,2) 5, 158	p-level
Pre_MC	104.82	132.25	0.79	0.56
Pre_SA	36.58	8.51	4.29	0
Pre_Tot	176.45	163.15	1.08	0.37
Post_MC	150.34	90.58	1.66	0.15
Post_SA	102.73	23.92	4.29	0
Post-Tot	217.71	152.87	1.42	0.22

Table 13. Average Test Scores for the Six Sessions

Grp	Pre_MC	Pre_SA	Pre_Tot	Post_MC	Post_SA	Post_Tot
1	23.64	4.02	27.66	30.91	9.23	40.14
2	26.94	2.97	29.92	36.11	9.92	46.03
3	20.28	2,22	22.5	29.72	7.94	37.67
4	21.32	2.08	23.39	31.32	5.53	36.63
5	23.8	1.25	25.05	35.22	5.25	40.41
6	22.32	0.9	23.22	33.78	5.76	38.24

The average scores on the pretest and posttest when all six groups are combined are shown in Table 14. Thus the data could be combined for the overall test scores from all six experimental groups. These combined pretest and posttest scores are used in subsequent analyses. The average time spent on learning was 35.4 minutes with a standard deviation of 17.34. The least amount of time was 0 (although these did not produce a usable trace) and the maximum length of time was 292.7 minutes (4.9 hours). Average gain (calculated as the pretest-posttest difference divided by the posttest score) was found to be o.4 (40%).

Table 14. Average Pretest and Posttest Scores Combined Across Sessions

Variable	**mean**	**standard dev.**	**minimum**	**maximum**
pretest MC	23 (46%)	11.5 (23%)	0 (0%)	50 (100%)
pretest SA	2 (8%)	3 (12%)	0 (0%)	16 (64%)
pretest TOT	24.75 (33%)	12.75 (17%)	0 (0%)	60.75 (81%)
posttest MC	33.5 (67%)	9.5 (19%)	10 (20%)	50 (100%)
Posttest SA	6.75 (27%)	5.25 (21%)	0 (0%)	21 (84%)
Posttest TOT	39.75 (53%)	12 (16%)	15 (20%)	66 (88%)
Average Gain	0.4 (40%)	0.25 (25%)	-1.0 (-100%)	1.0 (100%)

This interaction time was thus long enough to allow for significant learning to occur. A pretest-posttest Pearson correlation of 0.45 was obtained for the multiple choice items (MC), 0.46 for the short answer items (SA) and 0.56 for the overall tests. A test of significance of learning found that significant learning occurred with MC (multiple choice) items, with SA (short answer) items, and for the combined test items. The post-session interview data showed that most students felt they were able to learn and study the way they normally would have and all those interviewed found the exercise useful (see Appendix D for transcripts of these interviews).

Analyses

Cognitive Science Variables

Three major variables from the cognitive science domain were first examined to assess what proportion of the posttest variance they could account for. The three variables investigated were prior knowledge, time on task and motivation. It was expected that a significant amount of posttest variance will be explained by these variables as they have been extensively documented to correlate with learning achievement in computerized learning environments. It was important to establish the validity of the tools used in the experimentation by assessing these three variables. In addition, it was necessary to assess them in order to then analyze how much of the remaining variance could be explained by learning style and learner trace variables.

Prior knowledge was assessed through use of a pretest. Ten multiple choice items and five short answer items were used to assess prior knowledge of the subject matter (neural networks). Raw scores were obtained (out of a maximum score of 75) as well as percentage scores for the multiple choice section alone, the short answer section alone, and for the two combined.

Motivation was measured through responses to one questionnaire item. Four categories were used to describe student responses: intrinsic motivation (due to personal interest, personal career plans), extrinsic motivation (required course, extra marks provided for participation), both intrinsic and extrinsic motivation (students mentioned both types of motivation in their responses) and a category of 'other' for miscellaneous responses and no response.

Time on task was measured as the total time students spent interacting with the learning materials minus any time taken for pauses. It was expected that the longer students spent interacting with course content, the better they would score on the posttest. Achievement on the posttest was expected to be higher for those students with higher time on task scores, both as an absolute score and in terms of a greater difference between pretest and posttest scores (referred to as "gain"). The time on task variable was assessed both with and without the first lesson visited by all students.

Multiple regression was used to assess the relative contribution of the cognitive science variables pretest, time on task and motivation for participation in the study. The overall multiple regression was significant at $p < 0.05$ with an $r=0.48$ and $r^2=0.23$. Only the pretest and time on task variables were found to be significant. The pretest had a ß coefficient of 0.43 and the time on task a ß coefficient of 0.18 . When only the pretest and time on task variables were included, the overall regression was significant at $p < 0.05$ with and $r=0.48$ and $r^2=0.23$. The pretest variable had a ß coefficient of 0.44 and the time on task variable had a ß coefficient of 0.19.

When the first lesson is omitted, the overall multiple regression with the three variables was significant at $p < 0.05$ with an $r=0.47$ and $r^2=0.22$. Only the pretest and time on task variables were found to be significant. The pretest had a ß coefficient of 0.43 and the time on task had a ß coefficient of 0.14. When the regression was run with the pretest and time on task variables, the overall regression was significant at $p < 0.05$ with $r=0.47$ and $r^2=0.22$. The pretest variable had a ß coefficient of 0.45. The time on task variable had a ß coefficient of 0.14.

It was thus found that the learning environment was instructionally valid and could be used to generate learner data during learner interactions with the computerized course materials. The longer students spent interacting with the course materials and the higher they scored on a pretest of prior knowledge, the higher their achievement score on a posttest.

Educational Psychology Variables

Two learning style instruments were administered in order to evaluate their predictive value within this particular learning context: the Kolb Learning Style Inventory (LSI) and the Entwistle Approaches to Studying instrument (ASi). They were first analyzed using an ANOVA to assess whether there were any significant differences in the pretest, posttest, gain and time on task results across the different groups established by the instruments. Next, a comparison of the expected learner behaviours, as extrapolated from the two learning style theories, to actual learner behaviours detected in the analyses of student traces was carried out. The last type of analysis was to assess the similarities between the learning style groupings and groupings on trace data using both conventional statistical procedures and a neural network. ANOVAs were then carried out to determine whether the four Kolb groups showed any differences in on-line learner behaviours without reference to theoretical constructs.

<u>Kolb Learning Style Inventory</u>

The data obtained from student LSI forms showed a 0.59 correlation with the norms published by Kolb, for the 12-item version. This correlation was obtained through a Pearson correlation between the scores obtained by students on the learning style dimensions (AE, RO, CE, AC) and axes (AE-RO and AC-CE) and those scores obtained by Kolb in his published study (Kolb,1976). In the sample used, it was found that 45% of the students were classified as accomodators, 29% assimilators, 12% convergers and 14% divergers.

The first research question addressed was to assess the relative contribution of the Kolb learning style instrument to posttest, gain and time on task variance. This analysis was done using multiple regression analysis on the four Kolb LSI dimensions (AE, RO, CE, AC), the two Kolb axes (AE-RO and AC-CE) and the four Kolb groupings (converger, diverger, accomodator, assimilator). These measures were, of course, highly redundant. They were assessed independently of one another in order to evaluate the possibility that the Kolb LSI was not able to accurately assess the Kolb constructs postulated. In this way, all possible Kolb scores were evaluated with respect to their predictive value in this context.

None of the Kolb data had overall significance at a *p*-level of 0.05 when introduced into a multiple regression together with the pretest and time on task variables. When the two significant cognitive science variables of pretest, gain and time on task were omitted from the multiple regression, the overall regression was found to be significant. However, the Kolb variables were again found not to be significant.

An ANOVA of the four Kolb groups showed there was no significant difference between pretest and posttest scores across the different groups. In addition, no significant differences were found for the time on task variable (time) and for time on task when the first lesson visited was omitted (time-1) for the different Kolb categories. using a one-way ANOVA. Table 15 shows the mean data for the pretest and posttest. Table 16 shows the data for the time on task, both when all lessons are included in the analysis ("time") and when the first lesson visited is omitted from the analysis ("time -1").

Table 15. Mean Pretest, Posttest and Gain Scores for Kolb groups

Group	Pretest	SD	Posttest	SD	Gain	SD	N
accomodator	24.6	12.3	38.8	14.8	0.4	0.4	77
assimilator	22.9	12.8	37.6	12.9	0.4	0.7	50
converger	29.1	16.6	36.2	15.1	0.3	0.4	21
diverger	23.0	13.1	36.7	24.6	0.4	0.3	23

Table 16. Mean Time on Task Scores for Kolb groups

Group	Time	SD	Time -1	SD
accomodator	35.81	18.84	24.26	15.10
assimilator	34.16	16.77	22.93	14.00
converger	30.40	15.78	19.06	12.09
diverger	41.57	13.27	25.89	12.51

Thus the Kolb data do not appear to contribute any further explanation of posttest variance, beyond what can already be explained by the pretest and time on task variables.

The next analysis to be done was to order to compare the expected and actual profiles of students as characterized by the Kolb LSI. Expected profiles were established for each of the four groups and compared to actual on-line behaviors exhibited by students. A correlation analysis was used to assess the degree of fit for each of the four groups (converger, diverger, accomodator and assimilator), the two dimensions (AC-CE and AE-RO) and the four axes (AC, CE, AE, RO), both for all lessons visited and also when data from the first lesson visited was omitted.

Kolb groups. As discussed in the literature review, it was expected that converger students would show the best posttest achievement results as previous studies have shown convergers to perform best in computer-based learning environments. This hypothesis was not confirmed as there were no significant differences between posttest scores at $p=0.05$ both with and without the first lesson visited data as shown in the previous section.

Kolb axes. When the two Kolb axes are analyzed, it was expected that students scoring high on one should show behaviours that are diametrically opposed to students scoring high on the other dimension. Individuals with a high score on the AC-CE dimension should make greater and/or more frequent use of the examples option when interacting with the learning environment. Those individuals with a high score on the AE-RO dimension should show a negative correlation with example use. These differences were not found.

Thus expected correlations between student scores on the two Kolb axes and their subsequent learning behaviour in this environment were not found. The Kolb axes appear to have little predictive value relative to the research questions.

Kolb dimensions. Specific hypotheses concerning scores on the individual Kolb dimensions were then analyzed. The following behavior patterns were expected:

1. RO: individuals scoring high on the reflective-observation dimension are expected to make greater and more frequent use of the notes and self-test options as well as the condense information option. The notes option allows them to reflect upon their learning experience while the self-test option allows them to observe their learning progress. The condense information option includes the outline and summary which help provide higher level structure to the learning session. These students are expected to spend longer on all options they select.

No significant correlations were found for the time on task, condense information and notes variables, both when the first lesson visited was included and when it was omitted. A significant but low positive correlation was obtained for the frequency of self-test variable (r=0.156) when all lessons were included.

2. CE: individuals scoring high on the concrete- experimentation dimension were expected to make greater and more frequent use of the examples option and the quiz option. The examples option allows these learners to be concrete when learning concepts. The quiz option allows them to actively test their own knowledge throughout the learning sessions. They were expected to spend less time on all the options. No significant correlations were obtained for the examples and quiz options. No significant correlation was obtained for time on task.

3. AE: individuals scoring high on the active-experimentation dimension were expected to spend more time and make more frequent use of the elaborate-information and hypertext options. The elaborate-information options represent examples, definitions and so on which would allow the learner to be very active during the learning session. The hypertext option similarly allows more hands-on experimentation as they illustrate applications of the theoretical concepts. No significant results were obtained for these two options, both when the first lesson visited was included and when it was omitted.

4. AC: students scoring high on the abstract-conceptualization dimension were expected to make greater and more frequent use of condense information and gen_text options. These options contain the theoretical course content. No significant correlations were obtained for this dimension, both when the first lesson visited was included and when it was omitted.

Thus the four Kolb dimensions have very little, if any, predictive value relative to the research questions.

It was also hypothesized that perhaps the Kolb constructs had some validity but the instrument used did not adequately measure them. To this end, the behavioral variables were used to categorize students into four groups. This was done using conventional statistics (cluster analysis) and a new method, neural network-based pattern recognition (refer to the

Appendix E for more detailed explanation). The third research question addressed was to assess the similarity between groupings generated by the Kolb instrument and groupings generated independently, through both a cluster analysis of trace data and neural network classification on the same learner trace data. The neural network classification method was implemented in two different ways: in one, the number of groups was limited to four to allow direct comparison with the Kolb groups ("constrained neural network"). In the second, the number of groups was not specified and the neural network classification yielded a total of six groups ("unconstrained neural network").

No significant correlation was obtained between the four groups generated by the Kolb LSI and groupings generated using cluster analysis on trace data. This was the case both when all lessons were included in the analysis and when the first lesson visited was omitted. When a neural network was used to place all students into four different groups, using the same trace data, no significant correlation was obtained with the four Kolb groups. This was the case both when the first lesson visited was included and when it was omitted from the analysis. ANOVAs were then carried out in order to assess whether the Kolb groups showed any difference in means on the trace variables. This included means for duration and for frequency trace data, both for all lessons and without the first lesson visited.

When all lessons are included and the duration trace data are analyzed, no significant differences are found for any of the means at a *p*-level of 0.05. Similarly, no significant differences are found when the frequency trace data is analyzed.

When the first lesson visited is omitted, no significant differences are found, both for and duration trace data and for frequency trace data . When frequency trace data are analyzed, no significant differences are found for any of the trace variable means.

There do not appear to be any differences in learner behaviour across the different Kolb groups as manifested in the frequency and duration of use of the various options in the learning environment used for this study. Thus the expected learner behaviours as extrapolated from the Kolb LSI data do not appear to correlate significantly with the actual learner behaviours manifested in this learning environment. The Kolb LSI does not contribute to the posttest variance, nor to the time on task variance. The 12-item version of the instrument thus appears to be a very weak candidate to establish default values for a stereotype learner model for this kind of learning environment.

Entwistle Approaches to Study Inventory

In addition to the standard data generated for the eight Entwistle dimensions by the ASI, four quadrants were created to yield four groups. These groups were created using scores on the reproducing and meaning dimensions: Quadrant 1 (less than 12 on both), Quadrant 2 (less than 12 on reproducing but greater than 12 on meaning), Quadrant 3 (greater than 12 on reproducing and less than 12 on meaning) and Quadrant 4 (greater than 12 on both dimensions). In the sample, 9% were in Quadrant 1, 10% in Quadrant 2, 38% in Quadrant 3 and 44% in Quadrant 4. Quadrants were created in an arbitrary fashion in order to be able to compare Entwistle groupings with Kolb and trace data groupings using cluster analysis.

The first research question addressed was whether the Entwistle groupings helped explain any of the posttest variance. This was done both for the Entwistle dimensions as assessed by the ASI and for the four quadrants that were created using the meaning orientation and reproducing orientation dimensions.

Entwistle dimensions. The Entwistle dimension score data are summarized in Table 17 below. A correlation of 0.96 is obtained with the norms published by Entwistle. These data represent the more conventional way of using Entwistle ASI data.

A multiple regression analysis was carried out to determine what proportion of the posttest and gain variance could be explained by the eight Entwistle dimensions, all together, in various combinations, or alone. When all eight Entwistle dimensions were included in the regression equation, and posttest was used as the dependent variable, the overall regression was significant at $p < 0.05$ when the achieving, reproducing and meaning orientation dimensions were included ($r = 0.23$, $r^2 = 0.05$). Of these dimensions, only the reproducing dimension was significant with a ß coefficient of 0.22 . A multiple regression with only the meaning and reproducing dimensions was significant with an r=0.24 and r^2=0.06.

Table 17. Entwistle Approaches to Study Data

Variable	Mean	SD	Minimum	Maximum
achieving	12.2	3.0	0	22
reproducing	14.6	3.2	0	36
meaning	12.9	2.9	0	21
comprehension	14.6	2.9	0	36
operation	15.9	3.01	0	23
versatility	25.3	4.9	0	39
path. of learning	32.8	5.4	0	55
pred. of success	52.7	7.5	32	71

When only the reproducing dimension was included, the regression was significant at $p < 0.05$ with ß = 0.24. The pathologies of learning dimension alone yields a significant regression with an r=0.22 and r^2=0.05 and a ß coefficient of 0.22.

When the gain variable was used as the dependent variable, the overall multiple regression was significant at a *p*-level of 0.05 (r=0.19, r^2=0.03). Again, only the reproducing and pathologies of learning dimensions were found to be significant. The reproducing orientation variable had a ß of -0.11 and the pathologies of learning dimension variable had a ß of 0.18.

Thus the Entwistle reproducing dimension appears to be the most useful of the Entwistle dimensions in predicting learner success in this environment. This is in keeping with previous research findings that have the reproducing dimension to have the highest validity of all the Entwistle dimensions.

Entwistle quadrants. Next, the Entwistle quadrants were analyzed in the same manner to assess their contribution to posttest, gain and time on task variance. Multiple regression analyses were conducted using the four quadrants that were created based on reproducing and meaning scores in order to determine what proportion of the posttest variance was accounted for by these learning style variables. When all lessons are included, a multiple regression with pretest, time on task and the Entwistle quadrant variables was significant overall at a *p*-level of 0.05 with $r=0.52$ and $r^2=0.27$. All three variables were significant (pretest $\beta = 0.41$, time on task $\beta = 0.20$, and Entwistle quadrant $\beta = 0.19$). When the first lesson visited was omitted, the overall regression was again found to be significant with $r=0.5$ and $r^2=0.25$. The overall regression was significant, with $r = 0.51$, $r^2 = 0.27$ and the r^2 change for the Entwistle quadrant variable when entered last into the equation was 0.034. All three variables were again found to be significant: the pretest variable ($\beta=0.41$), the time on task variable ($\beta=0.15$) and the Entwistle quadrant variable ($\beta=0.18$).

Similarly, when learning gain was used as the dependent variable, the overall multiple regressions was found to be significant at a *p*-level of 0.0.5 ($r = 0.23$, $r^2 = 0.05$). The Entwistle quadrant variable had a β of 0.15.

Significant differences were found for pretest, posttest amd gain scores for the four Entwistle quadrants. Table 18 shows the means for pretest, posttest and gain scores for the quadrants. An ANOVA on the pretest scores showed a significant main effect (df effect =3, MS effect = 431.77, df error = 166, MS error = 166.41 and $F=2.59$) at a *p*-level of 0.05. A Tukey HSD post-hoc comparison found the quadrant II differed significantly from quadrant IV with respect to pretest scores. Students in quadrant II had the lowest pretest scores while those in quadrant IV had the highest.

An ANOVA on posttest scores showed a significant main effect (df effect = 3, MS effect = 711.96, df error = 166, MS error =188.12 and $F=3.78$) at a *p*-level of 0.05. A Tukey post-hoc comparison showed that Quadrant II again differed significantly from Quadrant IV with respect to posttest scores.

An ANOVA on learning gain scores showed a significant main effect (df effect = 3. MS effect = 36.06, df error = 166, MS error = 20.01 and F = 2.08. A Tukey post hoc test showed that the second quadrant differed from the others in that these students had a higher gain.

Table 18. Mean Pretest and Posttest Scores for the Entwistle Quadrants

Quadrant	Pretest	SD	Post	SD	Gain	SD	N
I	22.60	16.38	31.70	15.69	0.5	0.3	15
II	16.13	11.55	30.13	8.750	0.6	0.5	16
III	24.66	13.34	38.58	15.61	0.3	0.3	64
IV	26.63	12.00	40.47	12.35	0.5	0.4	75

Students in Quadrant II had the lowest scores while students in Quadrant IV had the highest scores. No significant differences were found for the time on task, both when all lessons were included and when the first lesson visited was omitted.

The Entwistle quadrants thus appear to account for some of the posttest and gain variance found and the use of such four groups to represent the ASI data also appears to be useful in this context.

The second research question addressed was to compare the expected learner behaviours, as extrapolated from their Entwistle ASI responses, to their actual learning behaviour patterns, as detected by their learner trace data. ANOVAs were done to ascertain whether there were any significant differences in their frequency and duration of use of the various learning environment options.

As was done with the Kolb LSI, the expected and actual profiles of students as characterized by the Entwistle ASI were derived. Expected profiles of the four Entwistle quadrants were compared to actual on-line behaviours exhibited by students. A correlation study was done in order to assess the fit between expected and actual behaviors. The Entwistle constructs were then investigated in order to determine whether or not there were any differences in learning behaviour that were correlated with different ASI scores. The following correlations were expected between the Entwistle profiles and actual learning behaviours exhibited by students:

1) High reproducing orientation score: Individuals scoring high on this dimension were expected to make greater and more frequent use of self-test. This reflects their emphasis on being able to learn in order to do well on test materials. They were also expected to make greater use of the condense information option as they tend to concentrate on being able to reproduce (i.e., memorize) material. In addition, it is expected that students possessing high reproducing scores will spend less time overall in learning (lower time on task) and should show a breadth-first lesson order strategy. This is because this particular approach to studying represents an attempt to maximize coverage of all topics and do so rapidly, in order to be able to reproduce the material on subsequent tests.

When all lessons are included in the analysis, a correlation was found between the reproducing dimension and time spent in self-testing ($r = 0.235$). No significant correlation was found for the condense information, time on task and lesson order strategy variables. In addition, there was an unexpected significant correlation between the reproducing orientation score and frequency that the help option was used ($r = -0.23$).

When the first lesson visited was omitted from the analysis, a correlation was found between the reproducing orientation score and time spent in self_testing ($r = 0.228$) and time spent in the help option ($r = -0.235$). No significant correlation was found for the condense information, the time on task and breadth-first lesson order variables.

2) Comprehension orientation: The score for the comprehension dimension was expected to be correlated with greater and more frequent use of elaborate information options and notes options. Learners who scored high on this dimension were expected to spend more time interacting with the materials (greater time on task) and a depth-first search strategy through the lesson topics.

No significant correlation was obtained with the time on task measure, both when all lessons were included and when the first lesson visited was omitted from the analysis. When all lessons were included in the analysis, a significant correlation was obtained for time spent in note-taking ($r = 0.29$). There was no significant correlation with the elaborate information option. When the first lesson visited is omitted, significant correlations were obtained for the time spent in the elaborate_information option ($r = 0.181$) and the time spent in note-taking ($r = 0.309$).

3) No significant correlations were expected for the other Entwistle dimensions. There were no hypotheses generated with respect these dimensions as there were insufficient data found in the literature review.

Thus a number of expected learner behaviour patterns based on Entwistle ASI scores were found to be exhibited by the learners. These correlations were both more numerous and stronger than those found with the Kolb LSI. The Entwistle ASI thus appears to be a more useful instrument in this particular learning context. Students that may be distinguished on the basis of their ASI responses appear to show correspondingly different learning behaviours that are consistent with the predicted behaviours. In particular, scores on the reproducing and meaning orientation dimensions appear to be the most useful items of information about learners. These dimensions explain some additional variance on the posttest and they correlate with actual learner behaviours in this particular context.

Next, the four Entwistle groupings were compared to groupings obtained using both conventional cluster analysis techniques and neural network-based classification. This was done to address the third research question which was whether the learner trace data, independent of any ASI theoretical constructs, showed any variation across the four Entwistle quadrants. This analysis was done by assessing the correlation between the Entwistle quadrants and the groupings formed using the trace variables. ANOVAs were then carried out to identify any differences in the means for these trace variables.

It was expected that there would be a correlation between Entwistle quadrants, based on reproducing and comprehension orientation scores, and trace variables. When all lessons are included in the analysis, significant weak correlations were obtained between the Entwistle groups and the groups based on lesson order strategy (r=-0.16), cluster analysis based on both frequency and duration trace data (r=-0.17) and groups based on guidance preference based on duration data (r=-0.18). No significant relationship was found between Entwistle quadrants and neural network groupings. When the first lesson visited was omitted from the analysis, a significant correlation was found with the cluster analysis grouping based on duration data only (r = -0.217). No significant relation was found with the ANN groups.

The Entwistle quadrants thus showed some correlation with cluster analysis groups based on duration trace data, after the first lesson visited was eliminated from the analysis.

ANOVAs were done both for duration and for frequency trace data, for all lessons and for the case where the first lesson visited was omitted from the analysis.

When all lessons are included and duration trace data are analysed, a significant difference at $p < 0.05$ was found for the help option and for the self-test option. Table 19 below presents the mean time spent in the various options. A main effect was found for the help option (df effect = 3, MS effect = 815.33, df error =166, MS error = 180.66 and F=4.52) at a p-level of 0.05. A Tukey post-hoc comparison showed that students in quadrant II differed from students in the other quadrants in that they spent much more time in the help option. Similarly, a main effect was found for the self-test option (df effect = 3, MS effect = 0.20, df error = 166, MS error = 0.67 and F =2.98) at a p-level of 0.05. A post-hoc comparison of the means showed that students in quadrant II spent more time in the self-test option than students in any of the other three quadrants.

Table 19. Means for Duration Trace Variables When All Lessons are Included

Quad.	Elab_inf	Cond_inf	Gen_txt	Help	Self_test	Notes	Hyp_txt
I	5.7	4.2	9.6	0.2	2.6	1.0	4.6
II	6.7	4.6	11.9	0.5	4.3	0.6	3.8
III	4.36	4.72	10.69	0.09	2.51	0.56	4.63
IV	4.82	4.05	11.61	0.11	2.76	1.05	3.79

When all lessons were included and frequency trace data were analysed, a main effect was found for the mean number of times the help button was used (df effect = 3, MS effect = 426.02, df error = 166, MS error = 80.19 and F = 5.31) at a *p*-level of 0.05.. A Tukey post-hoc analysis showed that students in Quadrant II had the most frequent use of help, averaging ten times during the session, while Quadrant III students had the lowest frequency, averaging once during the session. Students in the other two quadrants had an intermediate level of help usage (see Table 20). Thus the correlation between Entwistle quadrants and cluster analysis groups based on duration trace data appears to be primarily related to use of the help option (number of times used and length of time spent in the help option), when all the lessons are used in the analysis.

Table 20. Means for Frequency Trace data When All Lessons are Included (df effect = 3 and df error = 166, N=171)

Quad.	Elab_inf	Cond_inf	Gen_txt	Help	Self_test	Notes	Hyp_txt
I	8.45	5.1	4.8	1.7	9.8	1.8	6.9
II	11.0	8.3	5.8	9.9	21.0	1.3	6.4
III	8.4	6.9	6.8	1.1	12.8	1.8	7.1
IV	8.8	6.1	5.6	3.2	12.7	1.5	5.5

When the first lesson is omitted from the analysis and duration trace data are analysed, a significant main effect is found for the same two options: help and self-test. An ANOVA on the duration of use of the help button was significant (df effect = 3, MS effect = .78, df error = 166, MS error = 0.26 and F=3.08) at a *p*-level of 0.05. However, a Tukey post-hoc analysis did not show any significant differences between the four Entwistle group means on the use of this option.

A significant main effect was found in an ANOVA on the time spent in the help option (df effect = 3, MS effect = 877.78, df error = 166, MS error = 169.22 and F=5.19) at a *p*-level of 0.05. A Tukey post-hoc test found that students in Quadrant II used the help options for longer periods of time than students in any of the other three quadrants (refer to Table 21 below).

Table 21. Means for Duration Trace data When the First Lesson Visited is Omitted

Quad.	Elab_inf	Cond_inf	Gen_txt	Help	Self_test	Notes	Hyp_txt
I	2.7	2.9	8.23	0	2.3	0.9	2.4
II	5.7	3.5	10.68	0.5	4.1	0.5	2.5
III	2.7	3.7	8.74	0.1	2.4	0.5	3.3
IV	2.8	3.1	10.00	0.1	2.5	0.9	2.7

A significant main effect was also found for the time spent in the self-test option (df effect = 3, MS effect = 0.2, df error = 166, MS error = 0.62 and F=3.26) at a *p*-level of 0.05. A Tukey post-hoc analysis showed that students in Quadrant II spent more time in self-test activities.

When the first lesson visited is omitted and the frequency trace data are analysed, students in the different Entwistle quadrants were found to differ in the frequency with which they made use of the help button. An ANOVA on the frequency of use of the help button showed a significant main effect (df effect = 3, MS effect = 395.8, df effect = 166, MS effect = 89.12 and F=4.44) at a *p*-level of 0.05. A Tukey post-hoc test showed that students in quadrant II used this option an average of 11 times, while those in quadrants I and III used help an average of 2 times and those in quadrant IV an average of 4 times. Table 22 below summarizes the mean data for these options.

Table 22. Means for Frequency Trace Data When the First Lesson Visited is Omitted

Quad.	Elab_inf	Cond_inf	Gen_txt	Help	Self_test	Notes	Hyp_txt
I	4.9	2.9	5.8	2.1	6.0	0.8	3.2
II	8.5	6.7	5.6	10.	17.9	0.9	3.8
III	6.6	4.7	6.6	2.0	9.4	1.3	4.4
IV	6.6	4.4	5.4	3.8	9.6	1.1	3.8

Thus Quadrant II students showed different learning behaviours in this learning environment as exhibited by values for a trace variable. These students also scored the lowest on the pretest which indicates that this group may be comprised of students with the least prior knowledge of the subject matter, and thus may represent more novice-like learning behaviour. The questions on both the prestest and posttest wre analyzed to verify whether the questions were oriented to high reproducing orientation learners. The pretest consisted of 60% recall questions, 40% mearning questions, whereas the posttest had 40% recall items and 60% meaning items. Questions were labeled reproducing-type if students were required to provide answers that were almost verbatim from the content (e.g., definitions). Questions were considered to be of a meaning-type if students were required to carry out some inferencing (e.g., judgments, comparisons).

Artificial Intelligence Variables

Groupings based on learner trace data were analyzed in order to identify learning behaviour differences exhibited by the students. The relative contribution of trace variables to posttest variance and time on task variance was then analysed.

Trace variables yielded scores on the time spent in each option (duration) as well as the number of times each option was visited (frequency). This was automatically recorded for each student. The amount of time spent in Pause mode was subtracted from the total time. A second set of data was generated by omitting the first lesson visited by each student. This was done in order to assess whether there was a novelty effect or learning curve at the beginning of the interaction. The system recommended strategy consisted of following the default lesson order sequence, beginning with lesson module one through to lesson module five.

Classifications were created for learner data on guidance preference. Two types of data were used: selection frequency of a clear guidance preference and the amount of time spent as a percentage of total time in a particular guidance mode. Three groups were thereby formed: one that contained those students with a clear preference for system guidance (in excess of 75% of selections or time spent in this mode), those with a clear preference for student guidance (in excess of 75% of the selections or duration spent in learner control) and a third category of students who did not show a clear preference pattern. These three groups were formed for both frequency and duration guidance data.

Classifications were also created for learner data on lesson search strategy or sequence. Four groups were created in the following manner: the first group contained those students who exhibited a depth-first lesson selection strategy, as manifested by selection of the same two or fewer lessons for the first five lesson choices. Students in the second category chose to look at four or more different lessons within their first five lesson choices, thus exhibiting a breadth-first search strategy. Those students who chose three lessons within their first five lesson choices were placed in a third category as they did not show a clear strategy preference. Finally, those students who remained in system-guided mode throughout their interaction time, and hence necessarily followed the system recommended lesson strategy, were placed in a fourth group.

Classifications were also created based on learner activities during the learning session. These included the frequency with which options were chosen and the length of time students spent in them. Two parallel methods were used to create groups based on learner trace data: the first was statistical clustering using two statistical software packages (SPSS for PC and Statistica for the Macintosh). The second method used a Kohonen neural network to group students. Both methods used the same inputs to generate three groupings: one based on time data, one based on frequency data and one based on both time and frequency data. This was then repeated for the trace data after the first lesson visited was omitted.

In all, eight different types of groups were created and analysed: groups based on guidance preference frequency data (GF) guidance preference duration data (GD), lesson selection strategy (LS), cluster analysis on frequency data on learner options (CAF), on duration data on learner options (CAD), on the combination of both frequency and duration data on learner options (CAFD), and two neural network classifications, an unconstrained classification which yielded six categories (NN6) and a constrained classification which classified students into one of a maximum of four categories (NN4).

The similarities between the different types of groups are analysed first. Next, each grouping is analysed with respect to the amount of posttest and time on task variance that they can account for as well as differences in learner behavioural patterns across the groups. The final section summarizes the parameters identified for use as default values in a learner model, based on the results of this study.

Comparison of the classifications

A correlation matrix (shown in Table 23 below) was derived to investigate the similarities between the different types of groups that can be generated using learner trace data. When all lessons were included, there was a high correlation between the cluster analysis groups based on frequency data and those based on duration data. The groups based on both frequency and duration data were, of course, highly redundant with groups based on frequency data alone or those based on duration data alone. They were assessed separately in order to identify the most useful measure. It appears that both frequency and duration data together yield the most useful groupings.

Similarly, the groups based on guidance preference using duration and frequency data were also redundant and therefore highly correlated with one another. The groups based on guidance preference, both frequency and duration, showed some correlation with the cluster analysis groups based on duration, and based on both duration and frequency data. This is due to the fact that guidance preference choices were among the inputs to the cluster analysis groups.

Similarly, groups based on lesson selection strategy showed some correlation with the groups based on frequency learner data since a lesson choice was a choice like any other and was thus used as input to the cluster analysis.

Finally, there was some correlation between the groupings produced by the constrained neural network classification and the groupings produced by a cluster analysis on the same data. This suggests that the neural network classified the data in a similar but not identical way.

Table 23. Correlation Matrix of Groups Derived from Learner Trace Data When All Lessons are Included (significant correlations at a *p*-level of 0.05 shown only)

Group	CAF	CAD	CAFD	GD	GF	LS	NN6	NN4
CAF		0.34	-0.56			-0.20		
CAD	-0.34		-0.66	-0.21	-0.15			
CAFD	-0.51	-0.66		0.24	0.21			0.21
GD		-0.21	0.24		0.65			
GF		-0.15	0.21	0.65				
LS	-0.20							
NN6			0.21					
NN4								

The correlation analysis was repeated for the case where the first lesson visited was omitted. The results are shown in Table 24 below. Most of the data is similar to the case where all lessons were included in the analysis. However, the correlation between the constrained and unconstrained neural network groups is much higher (0.78) which indicates that the neural network classification is likely to be more useful if samples of learner data are obtained after a certain amount of time has elapsed. This allows the system to exclude any behavior patterns that may be due to novelty effects of the learning environment. Alternatively, it may require a certain amount of time before useful learner differences are manifested in learning environments, as most learners may act in a similar way when using a learning tool for the first time. The next section looks at each of the groups in more detail.

Table 24. Correlation Matrix of Groups Derived From Learner Trace Data When the First Lesson Visited is Omitted (significant correlations at a *p*-level of 0.05 shown only)

Group	CAF	CAD	CAFD	GD	GF	LS	NN6	NN4
CAF		0.29	0.72					
CAD	0.29		0.48	0.22	0.16		-0.16	-0.16
CAFD	0.72	0.48						
GD		0.22			0.53		-0.24	-0.17
GF		0.16		0.53		0.19	-0.24	-0.2
LS								
NN6		-0.15		-0.25	-0.23			0.78
NN4		-0.16		-0.17	-0.2		0.78	

Guidance Preference Groups

Trace data provided information on whether students showed a clear preference for a guided navigation mode (where the system suggests a lesson order) or an unguided one (where they may visit lessons in an order of their choosing). The proportion of time spent in each mode was used to create three groups: 1) those with a clear preference for guided (greater than 75% of time spent in guided mode), 2) those with a preference for unguided (greater than 75% of time spent in unguided mode) and, 3) all others (mixed mode). Similarly, three groups were created based on the frequency with which students selected guided vs. unguided modes. A clear preference for guided mode was associated with greater than 75% of choices to select guided, a clear preference for unguided was associated with those students who selected unguided greater than 75% of the time they made a choice, and the third category represented a mixed mode. This was repeated for data where the first lesson visited was omitted.

The majority of students (over 75%) were in the guided group (1). The groupings generated based on frequency data are presented first, followed by the analyses then repeated for groupings based on duration data.

When all lessons are included in the analysis, students in the three groups were found to differ with respect to the time they spent in the following options: selection of guidance mode (df effect = 2, MS effect = 0.83, df error = 167, MS error = 0.16, F = 5.22), elaborate information (df effect = 2, MS effect = 0.12, df error = 167, MS error = 0.3, F = 3.86) condense information (df effect = 2, MS effect = 0.59, df error = 167, MS error = 0.2, F = 2.95), self-test (df effect = 2, MS effect = 0.19, df error = 167, MS error = 0.68, F = 2.77) and notes (df effect = 2, MS effect = 0.15, df error = 167, MS error = 0.52, F = 2.79). They also differed with respect to the frequency with which they looked at the general text within each lesson (df effect = 2, MS effect = 129.73, df error = 167, MS error = 35.99, F = 3.6). Table 25 below summarizes these means.

The greatest differences are found between groups 1 (guided) and 3 (no clear preference). Students in group 3 spent significantly longer periods of time in changing their guidance mode, in elaborate information, condense information, self-test and notes. They consulted general text more frequently, an average of nine times, as opposed to the first group which averaged 5.5 times. The second group, which preferred an unguided mode of interaction was distinct in that they made the most use of the notes option.

Table 25. Trace Variable Means for the Guidance Preference Groups based on Frequency data when All Lessons are Included

Group	Prefs.	Elab. info	Cond.info	Self-test	Notes	Gen. text	N
1	1.5	4.6	4.1	2.6	0.7	5.6	129
2	1.7	9.2	3.9	2.9	2.6	4.7	4
3	2.3	5.5	6.3	3.9	0.6	9.1	27

When the first lesson visited was omitted from the analysis, the groups were found to differ only in the frequency of use of the help option (df effect = 2, MS effect = 258.11, df error

= 167, MS error = 92.43m, F = 2.79). The guided group (1) had the lowest mean (2.89 times), the unguided group had a mean of 7.25 times and group 3 (no clear preference) had a significantly higher mean frequency of use of the help option (8.96 times).

These guidance preference groups based on frequency data were next analysed with respect to differences on pretest, posttes, gain and time on task scores. When all lessons were included, and the groups were analyzed with respect to pretest, posttest and time on task scores, a significant main effect was found (df effect = 2, MS effect = 0.154, df error = 168, MS error = 0.47 and F = 4.19) at a p-level of 0.05. However, a Tukey post-hoc analysis failed to identify any significant differences in the means for the time on task variable.

When the first lesson visited was omitted, a significant main effect was found for the time on task variable (df effect = 2, MS effect = 0.16, df error = 166, MS error = 0.23 and F=7.21) at a *p*-level of 0.05. However, a Tukey post-hoc test did not show any significant differences in the mean time on task for these three groups.

Similarly, no significant effects were observed when gain was used as the dependent variable, both when all lessons were included and when the first lesson visited was omitted.

Next, the groups devised from guidance preference based on duration data were analysed in a similar manner. When all lessons are included in the analysis, students in the three groups are found to differ with respect to the amount of time they spent in choosing a guidance mode (df effect = 2, MS effect = 0.6, df error = 167, MS error = 0.16, F = 3.65), in elaborate information (df effect = 2, MS effect = 0.10, df error = 167, MS error = 0.3, F = 3.33) and in self-test (df effect = 2, MS effect = 0.29, df error = 167, MS error = 0.66, F = 4.37). They also differed with respect to the frequency with which they selected a guidance preference (df effect = 2, MS effect = 362.42, df error = 167, MS effect = 67.39, F = 5.38). Table 26 below summarizes these means.

Students in the guided group (1) spent less time in choosing guidance preference mode, in elaborate information and in self-test. They also made less frequent use of the guidance mode selection option. This group differed significantly from group 3 (no clear preference) as students in group 3 spent longer in these options and made more frequent use of the guidance mode selection option. Groups 1 and 2 (unguided) differed only in that students in the unguided group spent longer in the elaborate information option.

Table 26. Trace Variable Means for Guidance Preference Groups based on Frequency Data When All Lessons are Included

Group	pref time	pref freq	elab_info time	self-test time	N
1	1.6	10.4	4.7	2.6	134
2	1.7	10.4	8.6	3.0	4
3	2.2	17.0	5.5	4.3	24

When the first lesson is omitted from the analysis, no significant differences were found for the trace variable means for students in different groups.

An analysis of these groups with respect to pretest, posttest and time on task scores showed a significant difference in means only for the time on task variable, both when all lessons are included and when the first lesson visited is omitted from the analysis. When all lessons are included, a significant main effect was found for the time on task variable (df effect =2, MS effect = 0.152, df error = 167, MS error = 0.369 and F = 4.12) at a *p*-level of 0.05. However a Tukey post-hoc analysis showed no significant difference in these means. When the first lesson visited is omitted, a significant main effect is again found for the time on task variable (df effect = 2, MS effect = 0.2, df error = 167, MS error = 0.23 and F = 8.78) at a *p*-level of 0.05. However, a Tukey post-hoc analysis showed no significant difference in these duration means.

The groups based on guidance preference do not appear to explain any additional variance in the posttest and time on task variables. This is probably due to the fact that the majority of students chose to follow the system guided mode throughout their interaction with the learning environment. Students did differ in their learning interactions. The greatest difference is found between students who chose to be guided by the system and those who did not show a clear preference for either a guided or unguided mode. Students in the guided group spent less time in elaborate information, condense information, notes and self-test. Students in group 3 with no clear guidance preference spent longer on all these options. Students in the unguided mode made greater use of notes and elaborate information. Students in these groups were not found to differ significantly in any of the demographic variables (e.g., age or sex).

Lesson Sequence Strategy Groups

A lesson sequence grouping was created by looking at the order in which lessons were visited. This was roughly analogous to a breadth-first vs. a depth-first strategy. Those students who selected less than 2 lessons for the first five lesson choices were placed in the depth-first category (group 1). Those who selected four or more different lessons to look at within the first five lesson choices were placed in the breadth-first category (group 2). The remainder were placed in a mixed search strategy category (group 3). A fourth group was used for those students who remained in system-guided mode throughout their learning session and thus had no choice but to follow the system-recommended sequence. The analysis was repeated for data obtained when the first lesson visited was omitted.

These groups were first analyzed with respect to any significant differences in trace variable means. When all lessons are included in the analysis, a signficant difference is found for the time spent in notes (df effect = 3, MS effect = 0.18, df error = 166, MS error = 0.51, F = 3.48). Group 4, which consists of eight individuals who chose to remain in guided mode throughout their learning were found to spend longer in notes than students in groups 1 and 3 (see Table 27). No significant differences are found for any of the trace variables.

Table 27. Trace Variable Means for Lesson Strategy Groups When All Lessons are Included

Group	time spent in notes
1 (depth-first)	0.5
2 (breadth-first)	1.1
3 (mixed strategy)	0.4
4 (100% guided throughout learning)	2.8

These groups were then analyzed with respect to pretest, posttest, gain and time on task scores. No significant main effects were found for the pretest and time on task variables. A significant main effect was found for the posttest scores of these groups (df effect = 3, MS effect = 974.75, df error = 166, MS error = 181.79 and F=5.36) at a *p*-level of 0.05 when all lessons are included in the analysis. No significant differences were found for gain. The means are shown in Table 28 below. A Tukey post-hoc analysis showed that students in group 1 had posttest scores that were significantly lower than students in the group 4.

Table 28. Pretest, Posttest, Gain and Time on Task Means for Lesson Sequence Strategy Groups When All Lessons are Included

Group	Pretest	SD	Post	SD	Gain	SD	Time	SD	N
1	23.6	12.7	35.0	13.9	0.3	0.3	34.1	18.0	70
2	25.2	12.8	42,0	13.8	0.4	0.3	37.4	17.1	66
3	24.8	14.9	39.3	13.1	0.4	0.4	34.0	16.4	24
4	29.9	13.0	43.1	7.6	0.3	0.3	43.3	3.9	8

When the first lesson visited was omitted from the analysis, a significant main effect is again found for posttest scores (df effect = 3, MS effect = 595.24, df error = 167, MS error = 193.36 and F=3.08) at a *p*-level of 0.05. Table 29 shows the means for these groups. A Tukey post-hoc analysis showed that there was a significant difference in between the first and last group. Students in the first group had lower scores than those in the fourth group.

It thus appears that the choice of lesson topic strategy had an effect on posttest achievement. Students who adopted a depth-first approach had lower posttest scores than students who adopted the system-guided approach. Students in different groups did not show any significant differences with respect to demographic variables such as age, sex and background. A distinct group of students was found, consisting of eight individuals who chose to remain in guided mode throughout their learning session and who made more extensive use of the notes option.

Table 29. Pretest, Posttest, Gain and Time on Task Means for Lesson Sequence Strategy Groups When the First Lesson Visited is Omitted

Group	Pretest	SD	Post	SD	Gain	SD	Time-1	SD	N
1	15.5	15.0	30.7	26.9	0.4	0.4	4.9	8.5	3
2	23.2	13.5	37.8	15.2	0.4	0.3	24.0	15.4	77
3	25.7	13.1	37.6	12.9	0.4	0.3	23.1	13.7	62
4	26.2	12.1	39.3	12.9	0.3	0.3	24.5	10.8	29

Learner Trace Variables

Trace variables were analysed to determine which, if any, could contribute to explaining posttest variance independent of any groupings of these variables. A multiple regression of all trace variables showed overall significance with an r=0.3 and r^2=0.09. Three variables were significant: the general text duration variable was significant with a ß coefficient of 0.25, the help frequency variable with a ß coefficient of -0.2, and the condense information duration variable with a ß coefficient of -0.23. A forward stepwise regression of trace variables was found to be significant overall when all lessons were included with an r=0.29 and r^2=0.08. Six variables entered into the equation at a significance level of p=0.05 in the following order: general text duration, help frequency, condense information duration, condense information frequency, general text frequency and hypertext duration (refer to Table 30 below).

Table 30. Forward Stepwise Multiple Regression of All Trace Variables when All Lessons are Included (N = 171)

Variable	**Beta**	**p-level**	**r^2 change**
Gen text time	0.26	0.002	0.05
help frequency	-0.20	0.009	0.02
cond time	-0.23	0.046	0.02
cond frequency	0.19	0.048	0.01
gen text frequency	-0.11	0.049	0.01
hypertext time	-0.13	0.050	0.01

When the first lesson visited is omitted, a multiple regression with all trace variables was found to have overall significance with an $r=0.31$ and $r^2=0.06$. The only variables that were found to be significant were: general-text-duration with a ß coefficient of 0.27, and help-frequency with a ß coefficient of -0.22. A forward stepwise regression of all trace variables was found to be significant when the first lesson visited was omitted with $r = 0.32$, $r^2 = 0.10$. Three variables were entered into the equation: general text duration (r^2 change = 0.05), help frequency (r^2 change = 0.02) and elaborate information duration (r^2 change = 0.02). Only the first, general text duration, was significant with a ß coefficient of 0.27. Tolerance levels were found to be well within acceptable range for tests of multicollinearity.

Similarly, when gain was used as the dependent variable, the overall multiple regression was found to be significant at a *p*-level of 0.05 ($r = 0$-.2, $r^2 = 0.03$). Only two trace variables were entered into the forward stepwise regression: time spent in hypertext and time spent in the elaborate information options.

When the first lesson was omitted from the analysis, the overall multiple regression was significant at a *p*-level of 0.05 ($r = 0.3$, $r^2 = 0.04$). Three variables were entered into the forward stepwise regression equation: frequency of hyptertext use, time spend in elaborate information and time spent in condense information.

Therefore trace variables appear to contribute to a small percentage of the posttest variance. The highest amount of variance is explained when the first lesson is omitted and with the variables general text duration, help use frequency and time spent in elaboration of information. Groups based on frequency trace data did not show any significant main effects with any of the demographic variables. The next section looks at the analysis of groups formed by using a conventional statistical procedure, cluster analysis, on trace variables.

Cluster analysis groups of trace variables. Clusters obtained using trace variables were examined using duration data (time spent in each option), frequency data (number of times each option selected) and a combination of the two. Cluster analysis was used to group students based on their learning behavior. These groups were analyzed with respect to significant differences in the mean use of the various learning options. They were then analyzed with respect to differences in pretest, posttest and time on task means. As in the previous analyses, results are presented both for the case when all lessons are included and for the case when the first lesson visited is omitted.

When all lessons are included and groups are created based on frequency data, students were found to differ significantly with respect to all eight options: guidance preference selection (df effect = 3, MS effect = 0.1, df error = 166, MS error = 0.15, F = 6.81), elaborate information (df effect = 3, MS effect = 339.11, df error = 166, MS error = 5.12, F = 1.23), condense information (df effect = 3, MS effect = 258.12, df error = 166, MS error = 22.17, F = 11.64), general text (df effect = 3, MS effect = 264.42, df error = 166, MS error = 32.19, F = 8.21), help (df effect = 3, MS effect = 2752.14, df error = 166, MS error = 24.14, F = 114.03), self-test (df effect = 3, MS effect = 1093.12, df error = 166, MS error = 184.37, F = 5. 93), notes (df effect = 3, MS effect = 45.76, df error = 166, MS error = 6.79, F = 6.74) and hypertext (df effect = 3, MS effect = 1155.95, df error = 166, MS error = 61.9, F = 18.67). Table 31 summarizes the means for these variables.

Students in group 1 used the self-test, notes and hypertext options quite frequently. Students in group 2 showed much greater use of the help option and made no use at all of the notes and hypertext options. Students in group 3 are distinguished by the most frequent use of the guidance mode selection, condense information, general text and hypertext options. Group 4 students had generally less frequent use of these options.

Table 31. Trace Variable Means for Cluster Analysis Groups based on Frequency Data When All Lessons are Included

Group	pref	elab_inf	cond_inf	gen_txt	help	self-test	notes	hyp-text
1	13.4	11.0	8.5	6.1	1.0	19.0	2.6	6.8
2	13.4	7.6	7.2	0.7	30.0	5.9	0	0
3	16.1	14.6	11.1	12.3	0.3	17.8	3.1	21.8
4	8.6	6.6	4.3	5.8	0.6	9.8	0.8	4.3

When the first lesson visited is omitted, differences were once again found for all eight trace variables across the four cluster analysis groups: guidance preference selection (df effect = 3, MS effect = 635.77, df error = 165, MS effect = 63.96, F = 9.94), elaborate information (df effect = 3, MS effect = 174.96, df error = 165, MS error = 60.44, F = 2.89), condense information (df effect = 3, MS effect = 185.18, df error = 165, MS error = 21.34, F = 8.68), general text (df effect = 3, MS effect = 493.64, df error = 165, MS error = 34.91, F = 14.34), help (df effect = 3, MS effect = 972.01, df error = 165, MS error = 81.41, F = 11.94), self-test (df effect = 3, MS effect = 1511.29, df error = 165, MS error = 182.68, F = 8.27), notes (df effect = 3, MS effect = 12.87, df error = 165, MS error = 4.71, F = 2.73) and hypertext (df effect = 3, MS effect = 540. 87, df error = 165, MS error = 44.97, F = 12.03) (seeTable 32).

Students in group 1 were characterized by low use of help, moderate use of self-test and hypertext together with the most frequent use of notes. Students in group 2 showed the least frequent use of the general text option and the most frequent use of help. Students in group 3 spent less time changing their guidance mode, made the least use of condense information, self-test, notes and hypertext options and the most frequent use of general text. Students in group 4 made the most frequent use of guidance mode selection, elaborate information, condense information, self-test and hypertext and the least frequent use of help.

Table 32. Trace Variable Means for Cluster Analysis Groups based on Frequency Data When the First Lesson Visited is Omitted

Group	pref	elab_inf	cond_inf	gen_text	help	self-test	notes	hyp-text
1	8.6	7.3	5.3	6.0	0.9	11.3	1.6	5.0
2	8.2	5.2	4.1	1.8	10.6	8.5	0.8	1.5
3	6.2	5.2	2.0	9.9	2.4	3.0	0.4	1.1
4	18.2	10.8	8.4	8.7	0.9	21.8	1.3	11.2

ANOVAs were then done for the groups based on frequency trace data for the pretest, posttest, gain and time on task results (Table 33). A signficant main effect was found for the time on task variable (df effect = 3, MS effect = 0.284, df error = 166, MS error = 0.331 and F=8.6) at a *p*-level of 0.05.

Table 33. Means for Pretest, Posttest, Gain and Time on Task Variables for Cluster Analysis Groups based on frequency When All Lessons are Included

Group	pretest	SD	post	SD	Gain	SD	time	SD	N
1	22.7	12.9	36.8	13.8	0.4	0.5	41.5	15.8	59
2	28.4	10.2	39.1	13.0	0.3	0.4	35.9	12.7	14
3	19.4	10.3	39.1	13.2	0.3	0.3	45.8	18.0	15
4	26.6	13.4	40.3	13.5	0.4	0.4	29.9	16.3	81

A Tukey post-hoc analysis showed that students in group 4 differed from students in all the other groups in having spent much less time on interacting with the learning materials.

When the first lesson visited was omitted from the analysis, no significant difference was obtained for test score and gain means across the cluster analysis groupings based on frequency trace data. The difference in time on task means remained, however (df effect = 3, MS effect = 1518.6, df error = 165, MS error = 161.34 and F = 9.41) at a *p*-level of 0.05. Table 34 below summarizes these results. A Tukey post-hoc analysis showed that students in group 4 spent more time than students in groups 2 and 3. Students in groups 1 and 2 were also found to have significantly different time on task means. Students in group 2 spent the least amount of time on the learning task.

Table 34. Means for Pretest, Posttest, Gain and Time on Task Variables for Cluster Analysis Groups based on Frequency Data When the First lesson Visited is Omitted

Group	pretest	SD	post	SD	Gain	SD	time-1	SD	N
1	24,9	13,8	40.2	13.4	0.4	0.5	26.6	13.1	71
2	25.2	13.5	35.6	15.1	0.4	0.5	18.1	13.5	45
3	25.8	10.5	40.8	10.7	0.3	0.4	20.5	11.6	34
4	21.3	12.6	33.3	13.9	0.3	0.4	32.2	15.9	19

Thus cluster analysis groups based on frequency data showed that considerable differences existed across the four groups with respect to eight trace variables. Each group appears to have different interaction patterns with the learning materials. These patterns were found to be different when the first lesson was included and when it was omitted from the analysis. In addition, a difference was found for the mean time on task scores using this grouping method.

Cluster analysis groups were also generated using duration trace data. Four groups were obtained and differences with respect to trace variable means, pretest, posttest, gain and time on task means were analyzed as in the same manner (both with and without the first lesson visited).

When all lessons were included, the four groups were found to differ with respect to their means on the following trace variables: guidance mode preference (df effect = 3, MS effect = 0.19, df error = 166, MS error = 0.13, F = 14.22), condense information (df effect = 3, MS effect = 0.27, df error = 166, MS error = 0.15, F = 17.9), general text (df effect = 3, MS effect = 0.74, df error = 166, MS error = 0.81, F = 9.1), self-test (df effect = 3, MS effect = 0.83, df error = 166, MS error = 0.52, F = 16.14) and hypertext (df effect = 3, MS effect = 0.19, df error = 166, MS error = 0.47, F = 4.14). Table 35 shows the means for these variables.

Group 1 spent the most time in self-test and the least amount of time spent in hypertext. Group 2 had the longest time in general text and hypertext. Group 3 spent the most time selecting a guidance preference and in condense information. Group 4 students spent the least amount of time selecting a guidance preference, in condense information, and general text.

Table 35. Trace Variable Means for Cluster Groups based on Duration data When All Lessons are Included

Group	pref	cond. info	gen. text	self-test	hypertext
1	2.45	3.9	10.4	5.1	2.2
2	1.5	4.1	15.9	2.6	6.6
3	3.1	13.0	8.4	4.4	5.4
4	1.3	3.7	8.2	1.9	3.0

Cluster analysis groupings based on trace duration data were next analyzed to identify any significant differences in means for pretest, posttest, gain and time scores. No significant differences were found for pretest means, both when all lessons were included in the analysis and when the first lesson visited was omitted. When all lessons are included in the analysis, a significant difference was found, both for posttest scores and for the time on task variable (See Table 36).

Table 36. Pretest, Posttest, Gain and Time on Task Means for Cluster Analysis Groups based on Duration Data When All Lessons are Included

Group	pretest	SD	post	SD	time	SD	N
1	23.3	12.8	33.2	14.6	40.9	16.0	30
2	25.0	11.4	44.6	10.9	41.4	14.3	57
3	22.9	14.4	21.9	12.0	46.9	16.5	10
4	25.8	14.0	37.7	12.6	27.9	16.4	72

A significant main effect was found for posttest scores (df effect = 3, MS effect = 2174.35, df error = 166, MS error = 152.88 and F=14.22) at a *p*-level of 0.05. A Tukey post-hoc analysis determined that students in group 2 differed from students in the other three groups. Their scores were significantly higher than scores of students in the other groups. A significant main effect was also found for the time on task scores (df effect = 3, MS effect = 0.352, df error = 166, MS error = 0.314 and F=11.21) at a *p*-level of 0.05. A Tukey post-hoc comparison of the means showed that students in group 4 differed from students in the other three groups in having spent the least time interacting with the materials.

When the first lesson visited was omitted, the four cluster groups based on trace duration data show significant differences in the following trace variable means: guidance mode preference (df effect = 3, MS effect = 0.17, df error = 165, MS error = 0.11, F = 16.54), condense information (df effect = 3, MS effect = 0.36, df error = 165, MS error = 0.13, F = 2.77), general text (df effect = 3, MS effect = 0.11, df error = 165, MS error = 0.67, F = 16.59), self-test (df effect = 3, MS effect = 0.72, df error = 165, MS error = 0.53, F = 13.73), notes (df effect = 3, MS effect = 0.12, df error = 165, MS error = 0.44, F = 2.73) and hypertext (df effect = 3, MS effect = 0.45, df error = 165, MS error = 0.26, F = 17.48). Table 37 summarizes this data.

Table 37. Trace Variable Means for Cluster Analysis Groups based on Duration Data When the First Lesson Visited is Omitted

Group	pref	cond	gen text	self-test	notes	hyper-text
1	1.0	3.0	6.6	2.0	0.5	2.3
2	1.2	2.9	16.0	2.4	0.4	1.5
3	1.1	3.2	12.9	2.6	1.2	9.4
4	2.3	4.8	7.5	4.6	1.4	1.7

Group 1 spent the least amount of time choosing a guidance preference, in general text and in self-test. Students in group 2 spent the most time in general text and the least amount of time in condense information. Group 3 showed the most use of hypertext. Group 4 spent the most time choosing a guidance mode, in condense information, self-test and notes.

These groups were then analyzed with respect to pretest, posttest, gain and time on task scores. Significant differences were again obtained for posttest and time on task variable means. Table 38 below shows the results of this analysis. A significant main effect was observed for the posttest means (df effect = 3, MS effect = 1518.6, df error = 165, MS effect = 161.34 and F=9.41) at a *p*-level of 0.05. A Tukey post-hoc analysis showed that students in group 4 had lower posttest scores than students in the other three groups. In addition, students in group 2 differed from students in group 1. Students in group 2 had the highest posttest scores.

A significant main effect was also obtained for the time on task means when the first lesson visited was omitted from the analysis (df effect = 3, MS effect = 0.237, df error = 165, MS error = 0.214 and F=11.06) at a *p*-level of 0.05. A Tukey post-hoc analysis determined that students in group 1 differed from students in the other groups in that they spent the least amount of time in interacting with the learning environment.

Table 38. Pretest, Posttest and Time on Task Scores for Cluster Analysis Groups based on Duration Data When the First Lesson Visited is Omitted

Group	Pre	SD	Post	SD	Gain	SD	Time-1	SD	N
1	25.8	13.4	37.8	13.7	0.3	0.5	18.0	13.7	79
2	24.3	10.9	45.1	12.3	0.5	0.4	27.3	11.6	36
3	24.7	14.1	42.3	7.9	0.4	0.3	33.4	13.5	21
4	22.9	13.4	29.5	13.2	0.2	0.3	27.4	11.7	33

Cluster analysis groups were next formed using both frequency and duration data. They were again analyzed for significant differences in trace variables, pretest, posttest, gain and time on task scores, for all lessons and for the case where the first lesson visited is omitted from the analysis. When all lessons were included, the cluster analysis groups were found to differ with respect to trace variables. These results are presented in Table 39 below. Trace variable means are shown in Table 40.

Students in group 1 spent less time and (or) made less frequent use of the guidance preference, elaborate information, condense information, general text, self-test and notes options. Students in group 2 spent significantly more time on elaborate information, condense information and general text. They made more frequent use of the condense information option. Group 3 students were characterized by significantly more frequent use of elaborate information, condense information, general text, notes and hypertext. They spent more time in hypertext. These students used the help option less frequently. Students in group 4 spent more time in guidance mode selection and self-test. They made more frequent use of the guidance mode selection, help and self-test options. They used general text the least frequently of all the groups.

Table 39. ANOVA Results on Trace Variable Means for Cluster Analysis Groups based on Both Time and Frequency, When the First Lesson is Omitted (df effect = 3, df error = 166, *p*-level = 0.05; time in minutes, frequency in number of times used)

Variable	MS effect	MS error	F
pref time	0.2	0.1	11.3
pref freq	573.7	60.5	9.5
elab time	0.2	0.28	5.8
elab freq	365.9	65.6	5.6
cond tme	0.1	0.18	6.9
cond freq	200.7	23.6	8.5
gen text time	0.3	0.9	3.2
gen text freq	165.9	34.6	4.8
help freq	931.8	68	13.7
self-test time	0.64	0.6	11.3
self-test freq	697.9	193.9	3.6
notes freq	21.8	7.4	3.0
hypertext time	0.5	0.4	12.9
hypertext freq	11.0	63.4	17.3

Cluster analysis groups based on both frequency and duration trace data also showed significant differences in both posttest means and time on task variable means, when all lessons were included in the analysis. No significant effects were obtained for the pretest means and gain means(see Table 41).

Table 40. Trace Variable Means for Cluster Groups based on Frequency and Duration when all Lessons are Included (time in min, frequency in number of uses)

Variable	Group 1	Group 2	Group 3	Group 4
pref time	1.3	1.6	1.5	2.8
pref freq	8.5	10.6	12.2	18.5
elab time	3.1	6.7	4.7	6.5
elab freq	5.6	11.3	12.5	9.5
cond_info time	2.6	5.7	5.0	5.6
cond_info freq	4.2	8.3	9.4	6.9
gen_text time	9.4	13.9	12.0	9.8
gen_text freq	5.1	6.4	10.5	4.2
help freq	1.1	1.2	0.3	13.4
self-test time	2.0	2.8	2.7	5.0
self-test freq	9.8	12.9	16.1	20.4
notes freq	1.0	1.8	3.2	1.2
hyp_text time	3.0	3.8	11.8	1.8
hyp_text freq	3.7	6.4	18.9	2.8

A significant main effect is obtained for the posttest scores (df effect = 3, MS effect = 663.54, df error = 166, MS error = 160.83 and F=4.125) at a *p*-level of 0.05. A Tukey post-hoc analysis showed that students in group 2 and 4 differed from one another. Students in group 2 had the highest posttest score while students in group 4 had the lowest score. A significant main effect was also obtained for the time on task scores (df effect = 3, MS effect = 0.355, df error = 166, MS error = 0.313 and F=11.34) at a *p*-level of 0.05. A Tukey post-hoc analysis revealed that students in group 1 differed from the other groups in having spent the least time on the learning task.

Table 41. Pretest, Posttest, Gain and Time on Task Means for Cluster Groups based on Both Time and Frequency When All Lessons are Included

Group	Pre	SD	Post	SD	Gain	SD	Time	SD	N
1	27.9	13.67	38.7	14.1	0.3	0.3	27.1	16.0	65
2	23.7	12.1	41.8	13.0	0.4	0.4	40.9	15.7	53
3	19.0	10.6	39.2	11.3	0.5	0.5	44.3	16.1	22
4	24.0	13.1	30.2	12.7	0.2	0.3	40.0	13.9	29

When the first lesson was omitted, the groups were found to differ significantly with respect to trace variables, as summarized in Table 42 below.

Group 1 students spent the most time in general text and used the notes option frequently. Students in group 2 spent the least amount of time in, and made the least frequent use of, guidance mode selection, elaborate information, condense information, self-test and notes.

Group 3 students spent the most time in guidance mode selection, self-test, help and notes and made the least use of general text and hypertext. Students in group 4 spent the most time in elaborate information, condense information, general text, self-test and hypertext with very little use of help (see Table 43 for the variable means).

Table 42. ANOVAs of Trace Variables for Cluster Groups based on Both Time and Frequency data When the First Lesson Visited is Omitted (df effect = 3, df error = 165, p-level = 0.05)

Variable	MS effect	MS error	F
pref time	0.1	0.1	11.9
pref freq	887.6	59.4	15.0
elab time	0.1	0.3	5.3
elab freq	295.3	58.3	5.1
cond time	0.7	0.1	5.9
cond freq	240.9	20.3	11.9
gen text time	0.2	0.8	2.9
gen text freq	207.9	40.1	5.2
help freq	2405.7	55.3	43.5
self-test time	0.7	0.5	13.8
self-test freq	1911.2	175.4	10.9
notes time	0.4	0.4	8.8
notes freq	18.0	4.6	3.9
hyp text time	0.2	0.3	5.1
hyp text freq	400.7	47.5	8.4

Table 43. Trace Variable Means for Cluster Groups based on Time and Frequency Data, When the First Lesson Visited is Omitted

Variable	Group 1	Group 2	Group 3	Group 4
pref time	1.2	1.0	2.1	1.9
pref freq	8.7	5.8	11.4	18.0
elab time	3.6	1.6	4.6	5.2
elab freq	8.1	4.1	7.5	10.4
cond time	3.6	2.4	3.7	5.5
cond freq	5.7	2.3	6.7	7.7
gen text time	11.9	8.2	7.2	9.6
gen text freq	6.1	6.1	1.3	8.9
help freq	1.4	1.8	22.2	1.0
self-test time	2.6	1.8	4.6	4.0
self-test freq	11.5	4.0	9.1	22.8
notes time	0.8	0.2	2.5	0.4
notes freq	1.7	0.4	1.3	1.6
hyp text time	3.6	2.0	0.5	5.7
hyp text freq	5.1	2.	1.2	9.6

These groups were then analyzed with respect to pretest, posttest, gain and time on task scores. No significant differences are found for the pretest. However, the posttest and the time on task variable means showed significant differences. Table 44 below shows the results of this analysis.

Table 44. Pretest, Posttest, Gain and Tme on Task means for Cluster Analysis Groups based on Both Time and Frequency Data When the First Lesson Visited is Omitted

Group	Pretest	SD	Post	SD	Gain	SD	Time-1	SD	N
1	24.2	13.2	41.4	13.9	0.4	0.4	27.4	12.8	58
2	26.0	13.6	39.5	13.1	0.3	0.4	17.3	12.8	69
3	25.9	11.8	30.9	11.1	0.2	0.4	25.2	9.1	19
4	21.6	11.5	32.9	13.3	0.3	0.3	32.5	15.6	23

A significant main effect was found for the posttest means (df error = 3, MS error = 783.51, df error = 165, MS error = 174.71 and F = 4.4.8) at a *p*-level of 0.05 when the first lesson visited was omitted from the analysis. A Tukey post-hoc comparison showed that students in group 1 differed from students in groups 3 and 4. Students in group 2 also differed significantly from students in group 3. The highest scores were obtained by group 1 and the lowest by group 3 students.

A significant main effect was also found for the time on task means when the first lesson was omitted (df error = 3, MS error = 0.237, df effect = 165, MS effect = 0.214 and F = 11.04) at a *p*-level of 0.05. A Tukey post-hoc analysis showed that students in group 2 differed from students in all the other groups in that they spent significantly less time interacting with the learning environment.

The cluster analysis groups thus appear to contribute to the explanation of variance in the posttest and time on task variables. The critical variables distinguishing the groups appear to be the amount of time spent in general text, condensing information and hypertext, together with the frequency of user of help, condense information and general text options. In the final section, the groupings found using a neural network classification on the same trace data were analyzed.

Neural network classifications. Two different types of groups were obtained when the neural network was used to classify students based on their trace variable data. In one the neural network was constrained, i.e., the number of categories was specified, to be four in total. In the second, the network used was an unconstrained one: that is, there was no *a priori* determination of the number of categories students were to be assigned to and as a result, the network placed students into a total of six groups. Results are first presented for the constrained network, followed by the results for the unconstrained network.

The four groups obtained using the constrained neural network were analyzed with respect to trace variable means. When all lessons are included in the analysis, they were found to differ in their use of learning options. The results of these analyses are presented in Table 45 below.

Table 45. ANOVA Results on Trace Variables for Constrained Neural Network Groups (df effect = 3, df error = 166, *p*-level = 0.05)

Variable	MS effect	MS error	F
elaborate info time	0.1	0.3	4.0
condense info freq	93.3	26.1	3.6
general text time	0.39	0.9	4.5
general text frequency	105.3	36.0	2.9
help frequency	260.4	84.2	3.1
notes time	0.2	0.5	4.7
hypertext time	0.3	0.4	7.4
hypertext frequency	696.6	73.1	9.5

The trace variable means are shown in Table 46. Group 1 students spent the most time in general text and they made the least frequent use of notes. Students in group 2 spend the most time in hypertext and made the most frequent use of hypertext, condense information, and help. Group 3 students spent the least amount of time in elaborate information, general text and hypertext. They made the least frequent use of condense information, general text and hypertext. The showed significantly more frequent use of help and spent more time in notes. Students in group 4 spent the most time in elaborate information and used general text the most frequently.

Table 46. Trace Variable Means for Constrained Neural Network Groups When All Lessons are Included

Variable	Group 1	Group 2	Group 3	Group 4
elaborate info time	4.8	3.8	2.1	7.4
condense info freq	6.5	8.8	2.9	5.7
general text time	13.2	10.1	5.8	8.0
general text freq	5.7	8.0	0.6	6.3
help frequency	3.1	0.3	13.4	3.8
notes time	0.6	0.9	3.9	0.7
hypertext time	3.1	8.9	0.0	3.6
hypertext freq.	4.2	14.3	0.4	6.3

These groups were then analyzed for pretest, posttest , gain and time on task differences. No significant main interaction effect was found for pretest scores across the four neural network groups. A main interaction effect was obtained for the four groups created by the neural network for both posttest scores and time on task scores, when all lessons were included in the analysis. The posttest interaction effect (df effect = 3, MS effect = 1459.84, df error = 166, MS error = 170.1 and F = 8.58) is significant at a *p*-level of 0.05. Table 47 below shows these results. A Tukey post-hoc analysis found that students in group 1 differed significantly from students in group 4. Students in group 1 had the highest posttest score while students in group 4 had the lowest posttest score.

Table 47. Pretest, Posttest, Gain and Time on Task Means for Constrained Neural Network Groups When All Lessons are Included

Group	Pretest	SD	Post	SD	Gain	SD	Time	SD	N
1	26.9	12.5	41.8	12.9	0.4	0.4	35.9	16.0	95
2	20.8	11.5	35.4	13.0	0.4	0.4	39.1	20.5	32
3	22.8	13.6	36.9	10.9	0.4	0.4	26.2	18.3	9
4	22.9	14.6	31.8	14.1	0.3	0.4	35.8	15.1	35

A main effect was also found for the time on task variable (df effect = 3, MS effect = 0.142, df error = 166, MS error = 0.365, and F= 3.89) at a *p*-level of 0.05 when all lessons were included in the analysis. A Tukey post-hoc analysis revealed that students in group 2 were significantly different from students in group 3. Students in group 2 spent the most amount of time interacting with the learning environment while students in group 3 spent the least time.

When the first lesson visited was omitted from the analysis, the groups were found to differ in the time spent in four trace variables: elaborate information (df effect = 3, MS effect = 0.12, df error = 165, MS error = 0.25, F = 4.79), general text (df effect = 3, MS effect = 0.6, df error = 165, MS error = 0.76, F = 7.93), self-test (df effect = 3, MS effect = 0.19, df error = 0.62, F = 3.1) and hypertext (df effect = 3, MS effect = 0.35, df error = 165, MS error = 0.27, F = 12.76) and the frequency of use of hypertext (df effect = 3, MS effect = 501.39, df error = 165, MS error = 45.68, F = 10.98) (see Table 48).

Table 48. Trace Variable Means for Constrained Neural Network Groups When the First Lesson Visited is Omitted

Group	elab_info time	gen_text time	self-test time	hyp_text time	hyp._text freq
1	6.1	6.9	3.5	2.0	4.8
2	3.0	11.0	2.9	2.1	2.5
3	2.9	11.7	2.1	8.0	10.7
4	1.2	3.4	1.8	1.3	2.0

Group 1 students spent the most time in elaborate information and self-test. Group 2 students spent the most time in general text and made very infrequent use of hypertext. Group 3 students also spent a lot of time in general text but they spent the most amount of time in hypertext and made use of hypertext the most frequently of all the groups. Group 4 is characterized by the least amount of time spent in all the options and the least frequent use of the hypertext option.

When the groups were analyzed for pretest, posttest, gain and time on task measures, no main interaction effect was found for the pretest means. A main interaction effect was found for the posttest scores (df effect = 3, MS effect = 631.86, df error = 165, MS effect = 177.46 and F = 3.56) at a *p*-level of 0.05. Table 49 shows these results. Students in group 2 had the highest posttest score while students in group 1 had the lowest.

Table 49. Pretest, Posttest and Time on Task Means for Constrained Neural Network Groups When the First Lesson is Omitted

Group	Pre	SD	Post	SD	Gain	SD	Time-1	SD	N
1	22.1	12.4	30.7	13.6	0.3	0.4	24.2	13.5	21
2	26.4	12.5	40.5	13.4	0.4	0.5	24.5	13.4	98
3	24.4	14.0	38.6	11.3	0.4	0.4	29.0	12.5	26
4	24.2	13.0	35.4	14.8	0.3	0.3	14.6	14.6	24

A significant main effect was also found for the time on task variables (df effect = 3, MS effect = 0.121, df error = 165, MS error = 0.234 and F = 5.15) at a *p*-level of 0.05. A Tukey post-hoc analysis showed that students in group 4 were different from students in the other three groups. Students in group 4 spent the least amount of time interacting with the learning environment. Thus the four groups formed by a constrained neural network classification appear to explain some of the posttest variance and the time on task variance. The final section looks at the results of the unconstrained neural network which found six groups of students.

The unconstrained neural network groups were first analyzed with respect to differences on trace variable means. When all lessons are included in the analysis, students were found to differ in the use of learning environment options (see Table 50). Group 1 students spent the most time in elaborate information and less time in notes. Group 2 students spent the most time in general text and very little time in notes. Group 3 students spent the longest in the general text option and made greater use of hypertext. Group 4 students spent the most time in notes together with a fairly long time in general text. They made very infrequent use of hypertext. Group 5 students spent the least amount of time in hypertext and elaborate information. They had greater time in general text and notes. Group 6 spent the least amount of time on condense information, general text and hypertext options. Table 51 shows the trace variable means for these data.

Table 50. ANOVAs of Trace Variable Means for Unconstrained Neural Network Groups When All Lessons are Included (df effect = 5, df error = 163, p-level = 0.05)

Variable	MS effect	MS error	F
elab_info time	0.8	0.3	2.5
cond_info time	0.8	0.2	4.4
cond_info freq	62.1	26.2	2.4
gen_text time	0.3	0.9	3.2
notes time	0.1	0.5	2.5
hypertext time	0.3	0.4	7.4
hypertext freq	509.8	69.5	7.3

Table 51. Trace Variable Means for Unconstrained Neural Network Groups When All Lessons are Included

Variable	Group 1	Group 2	Group 3	Group 4	Group 5	Group 6
elab_info time	7.5	4.7	4.1	4.5	3.1	3.3
cond_info time	4.6	4.0	3.3	4.1	9.7	4.7
cond_info freq	5.7	6.7	6.5	4.7	11.3	4.0
gen_text time	8.1	12.9	13.3	13.0	6.8	4.6
notes time	0.8	0.6	1.1	3.9	0.3	0.6
hyp_text time	3.7	2.4	10.3	1.0	7.6	0.3
hyp_text freq	7.1	3.2	15.2	1.0	10.9	1.0

These unconstrained neural network groups were then analyzed with respect to pretest, posttest, gain and time on task scores. No significant main effect was observed for the pretest means, gain and time on task means. A main effect is found for the posttest means (df effect = 5, MS effect = 843.71, df error = 163, MS error = 172.82 and F = 4.88) at a p-level of 0.05. The results are summarized in Table 51 below. A Tukey post-hoc analysis showed that group 6 was significantly different from group 2. Group 6 had the lowest posttest score while group 2 had the highest. Table 52 below shows these results.

When the first lesson was omitted from the analysis, the six neural network groups were found to differ with respect to learning options (see Table 53 below). Group 1 students spent the most time in elaborate information. They also spent a lot of time in general text and self-test. They spent little time in notes. Group 2 students had high general text time and frequency together with very low amount of time spent in notes. The trace variable means are presented in Table 54

Table 52. Pretest, Posttest, Gain and Time on Task Variables for the Unconstrained Neural Network Groups When All Lessons are Included

Group	Pre	SD	Post	SD	Gain	SD	Time	SD	N
1	21.1	15.3	32.7	14.2	0.4	0.4	35.6	16.1	37
2	27.1	12.0	41.4	13.2	0.3	0.3	35.0	16.1	82
3	24.1	12.6	39.5	9.9	0.4	0.5	40.7	16.0	27
4	20.23	14.6	37.9	15.6	0.5	0.6	38.3	26.9	6
5	22.5	10.4	34.5	17.4	0.3	0.4	40.4	20.6	11
6	27.1	11.1	25.9	9.5	0	0.5	31.4	10.79	4

Group 3 students spent the longest time in general text and hypertext. They were the most frequent users of hypertext. They used general text frequently and help very infrequently.

Table 53. ANOVA Results for Unconstrained Neural Network Groups and Trace Variables When the First Lesson Visited is Omitted (df effect = 5, df error = 162, p-level = 0.05)

Variable	MS effect	MS error	F
pref time	0.4	0.1	3.2
elab_info time	0.7	0.3	2.8
cond_info time	0.5	0.1	4.2
gen_text time	0.5	0.7	6.3
gen_text freq	106.6	40.8	2.6
help freq	364.4	87.4	4.2
self-test time	0.2	0.6	3.0
notes time	0.2	0.4	4.2
hyp_text time	0.2	0.3	6.7
hyp_text freq	263.7	46.1	5.7

Table 54. Trace Variable Means for Unconstrained Neural Network Groups When the First Lesson Visited is Omitted

Variable	Group 1	Group 2	Group 3	Group 4	Group 5	Group 6
pref time	1.5	1.4	1.0	1.3	1.9	0.1
elab_info time	6.3	3.1	2.9	2.1	1.4	0
cond_info time	3.4	3.5	2.2	6.6	3.3	0.7
gen_txt time	7.2	11.6	11.7	5.2	1.8	0.5
gen_txt freq	6.7	6.7	7.0	4.6	0.4	0.3
help freq	5.5	3.6	0.7	1.6	17.6	0
self-test time	3.5	2.9	2.1	2.5	2.8	0
notes time	0.4	0.5	1.1	0.3	3.5	0.5
hyp_text time	2.1	2.2	8.0	2.3	0.4	0
hyp_text freq	5.1	2.5	10.7	3.5	1.2	0.4

Group 4 students spent the longest time in condense information and the least amount of time in notes. Group 5 students spent little time in general text and used this option infrequently. They showed the greatest frequency of use of the help and notes options. Group 6 was characterized with very low scores on both time and frequency of use of all the options.

These groups were then analyzed with respect to pretest, posttest, gain and time on task scores. No significant main effects are observed for the pretest. A main effect was obtained for the posttest means (df effect = 5, MS effect = 400.36, df error = 162, MS error = 177.62 and F = 2.25) at a p-level of 0.05. The results are summarized in table 55 below.

Table 55. Pretest, Posttest and Time on Task Means for Unconstrained Neural Network Groups When the First Lesson is Omitted

Group	Pre	SD	Post	SD	Gain	SD	Time-1	SD	N
1	18.0	12.7	30.8	14.0	0.4	0.3	24.6	13.7	20
2	26.2	12.4	41.0	13.2	0.4	0.5	25.3	13.3	93
3	24.5	14.0	38.6	11.3	0.4	0.3	29.0	12.5	26
4	20.3	13.8	35.4	15.8	0.4	0.3	20.5	14.0	13
5	28.3	12.5	31.2	11.8	0.1	0.3	14.9	10.2	9
6	28.7	11.3	36.4	16.5	0.2	0.3	11.8	4.8	7

A Tukey post-hoc analysis revealed that groups 1 and 5 had significantly lower posttest means than group 2, which had the highest. A significant main effect was also obtained for the time on task means (df effect = 5, MS effect = 0.124, df error = 162, MS error = 0.216 and F = 5.71) for a p-level of 0.05. A Tukey post-hoc analysis showed that group 6 students spent the least amount of time in learning. Students in group 3 spent the most time on the learning task. When all lessons were included, the groupings were found to be significantly correlated with three trace variables: condense information (0.206), elaborate information (-0.229) and career plans (0.156). When the first lesson was omitted, the neural network clusters showed significant correlations with a larger number of trace variables, as shown in Table 56 below.

Table 56. Significant Correlations of Neural Network Clusters and Trace Variables When the First Lesson Visited is Omitted

Variable name	Correlation coefficient
Kolb AE-RO dimension	-0.2
Kolb AE score	-0.2
Entwistle reproducing orientation	0.2
Time on task	-0.3
Guidance preferences duration	-0.2
Elaborate information duration	-0.3
General text duration	-0.3
Self-test duration	-0.3
Guidance preferences frequency	-0.3
Elaborate information frequency	-0.2
Condense information frequency	-0.2
General text frequency	-0.3
Cluster grouping based on time	-0.2
Cluster grouping based on frequency	-0.2
Cluster grouping based on time and frequency	-0.2

Multiple Regression Combining Cognitive Science, Learning Style and Trace Variables

An attempt was then made to fit a multiple regression equation using the best possible combination of cognitive science, learning style and trace variables in order to maximize the amount of posttest variance that could be explained in this learning environment. A forward stepwise multiple regression of all variables was found to be significant, with 17 variables entered into the equation in the following order: pretest, gen-text-duration, Entwistle quadrant, help-frequency, hypertext-duration, cond-info-duration, time on task, elab-info-duration, preference, Kolb LSI AE score, gen-text-frequency, cluster group based on frequency trace data, condense-info-frequency, cluster group based on both duration and frequency trace data, motivation, general knowledge of AI, and the ASI reproducing orientation score.

Six variables are found to be significant: pretest, help-frequency, condense-info-duration, time on task, elaborate-info-duration, and general-text-frequency. Table 57 summarizes these results.

When all lessons are included, the best multiple regression equation was found to be: posttest = 21.93 + 0.39 (pretest) -0.21 help frequency -0.21 condense information duration + 0.39 time on task - 0.24 elaborate information duration -0.15 general text frequency.

When gain was used as the dependent variable, and all lessons were included in the analysis, the forward stepwise regression was found to be significant. The following variables were entered into the equation: pretest, hypertext frequency, Kolb LSI AC score, cluster group based on duration data, time spent in elaborate information and lesson sequence order group. Of these, the pretest, hypertext duration, cluster analysis group based on duration data and time spent in elaborate information were found to be significant. The best regression equation thus appears to be: gain = 18.44 - 0.15 pretest -0.15 hypertext frequency + 0.13 cluster analysis group based on duration data + 0.13 time spent in elaborate information.

Thus the unconstrained neural network groups also appear to contribute to the explanation of posttest, gain and time on task variance.

When the first lesson visited was omitted, 15 variables were entered into the equation but only four are found to be significant: pretest, general-text-duration, cluster group based on both duration and frequency trace data, and general AI knowledge (see Table 58).

Table 57. Forward Stepwise Multiple Regression of All Variables.

Variable: Posttest Dep Var	Beta	Tolerance level	r^2 change
pretest	0.39	0.89	0.17
help-frequency	-0.21	0.76	0.05
cond-info-duration	-0.21	0.68	0.03
time on task	0.38	0.27	0.02
elab-info-duration	-0.24	0.61	0.02
gen_text_frequency	-0.15	0.79	0.01
Variable: Gain Dep Var			
pretest	-0.15	0.90	0.2
hypertext_frequency	-0.15	0.59	0.07
cluster group duration data	0.13	0.80	0.06
elaborate_info_time	0.13	0.52	0.05

The overall regression was significant with $r = 0.59$ and $r^2 = 0.35$. The r^2 change when the Entwistle quadrant variable was entered last into the equation was 0.04. The best multiple regression was found to be: posttest = 37.41 + 0.35 pretest + 0.16 general text duration - 0.27 cluster analysis group based on time and frequency + 0.13 Entwistle quadrant. Tolerance levels were again acceptable for these values, as shown in Table 58 below.

Table 58. Forward Stepwise Multiple Regression of All Variables When the First lesson Visited is Omitted (N=171)

Variable: Posttest Dep Var	Beta	Tolerance level	r^2 change
pretest	0.34	0.88	0.17
general text duration	0.19	0.77	0.04
cluster duration and frequency	-0.26	0.40	0.06
Entwistle quadrant	0.16	0.64	0.04
Variable: Gain Dep Var			
pretest	-0.13	0.91	0.13
elaborate_info_time	0.26	0.52	0.05
condense_info_time	0.16	0.47	0.05

Table 59 below summarizes the major findings of the study. The possible implications are included in this summary table. While this study did not implement and assess the effectiveness of any instructional interventions, the implications stated here are possibilities that may form the subject of future studies. These possible implications are discussed further in the subsequent Discussion chapter.

Table 59. Summary of Results and Implications

Phenomenon	Variables	Observations	Poss. Implications
Data collection over six test sessions	Group equivalency	No significant differences	It is possible to pool the data collected from the different test sessions
Student learning and instructional validity of materials	Pretest	Content validity exists good item difficulty good item discrimin. normal dist. of scores good inter-rater correl.	The pretest is a valid, easy to use, reliable *a priori* assessment tool of existing domain knowledge that can distinguish experts from novices
	Posttest	Content validity exists good item difficulty good item discrimin. normal dist. of scores good inter-rater correl.	Both the pretest and the posttest are valid instruments
	Gain	Average 40%	The learning environment is instructionally valid and students did in fact learn as they interacted with the system
	Time on task	Average 35 minutes	
	Student learning	Sig. improvement	
	Exit interviews	Corroborates that learners were able to study and learn	
Learning styles	Kolb LSI	No relationship to learning achievement, learning efficiency, learning interaction patterns; expected profiles not observed	Not a useful *a priori* assessment tool to use in this context
	Entwistle ASI	<u>Summary</u> reproducing and meaning orientation scores related to learning achievement and interaction patterns; both reproducing and meaning orientation profiles observed....... (details presented on next page.....)	useful *a priori* assessment tool to help predict posttest achievement and specific learner behaviors with respect to learning environment options used..... (details presented on next page....)

Table 59.....continued

Detailed Entwistle ASI variables	Observations	Possible Implications
Students with high meaning orientation and high reproducing orientation scores	High posttest scores	Because these students were successful learners, no further intervention is necessary
Low reproducing orientation and high meaning orientation scores	low posttest scores	These students require early intervention as they are not successful at learning with this system
High meaning score should be linked to: (1) higher use of notes (2) higher use of elaborate information (3) greater time on task (4) a depth-first lesson order sequence	 observed observed not observed not observed	These students could be encouraged to try to self-test or quiz themselves more, at least after each lesson module, in order to improve their posttest scores
High reproducing score should be linked to: (1) higher use of self-test (2) higher use of defer posttest (3) higher use of definitions and dictionary (elaborate information option) (4) low time on task (5) breadth-first lesson order sequence	 observed not observed not observed not observed not observed	Feedback to encourage these students to spend less time on self-testing and more time interacting with the content (as found in the general text option) in order to increase learning achievement

Table 59....continued

Phenomenon	Variables	Observations	Implications
Summary Interaction variables yield distinct groups of learners	Trace data from learner-system interactions	Cluster analysis and ANNs find very similar groups that differ with respect to learning achievement and learning efficiency - except for the fourth group (see next page for details)	ANNs are useful real-time learner diagnosis tools to predict posttest achievement, time on task; these data can be used to update an initial stereotype model (see next page for details)
Guidance preference groups	General Guidance preference was used to create the following groups:	General No link to posttest performance	General (see next page for details) Guidance mode preference appears to be just that - a preference
	(1) prefer guidance	changed guidance mode less; spent less time in all options, used help an average of three times	Students a bit passive - can be encouraged to make more use of the options offered
	(2) prefer self-guidance	spent more time in elaborate information; used help an average of seven times	Would be helpful to initially guide these students, in order to decrease use of help
	(3) no clear preference	frequent change of guidance mode; spent more time in all options; used help an average of nine times	These students may require human intervention to help them understand the difference between the two modes
Lesson order sequence groups	General Lesson selections used to create the following groups:	General Related to posttest achievement	
	(1) depth-first strategy	low posttest score	These students may have to be reminded that they need to allow enough time to visit all five lesson modules
	(2) breadth-first strategy	high posttest score	Successful learners - no intervention needed
	(3) no clear preference	high posttest score	Successful learners - no intervention needed
	(4) guided 100% of the time	high posttest score high use of notes	successful learners - no intervention needed

Table 59...Continued (Detailed Outcomes of Cluster Analysis)

Cluster analysis of trace data	Cluster analysis on trace data based on time and frequency created the following groups:	Related to posttest, time on task and interaction variable mean scores	Conventional statistical analysis of trace data yields groups that differ in learning achievement, efficiency and use of learning options
	(1) high posttest score with low time on task	very interactive students who made frequent use of all the options without spending too much time in any one option	Successful learners
	(2) high posttest score with moderate time	longer time spent in all the options used, especially general text, elaborate information, condense information	Successful learners who require intervention only if there are time constraints on learning
	(3) high posttest score with moderate time	frequent use of all the options together with greater use of hypertext and much less use of help	Successful learners who are proficient users of the system features; may require intervention only if there are time constraints
	(4) low posttest score with high time on task	spent less time in general text and made greater use of self-test	Early feedback may be required to re-direct these students to content options such as general text with less focus on self-testing

Table 59....Continued (Detailed Outcomes of ANN Classification)

ANN classification of trace data	ANN pattern classification yielded the following groups:	related to posttest, time on task and interaction variable mean scores	ANN analysis of trace data yields groups that differ in learning achievement, efficiency and use of learning options
	(1) high achievement with high efficiency	spent less time in options and made greater use of notes and help	Successful learners; a study of their help usage can be used to improve the system design
	(2) high achievement with moderate efficiency	less use of notes and greater time spent in general text	Successful learners; no feedback required unless there are time constraints
	(3) high achievement with moderate efficiency	greater use of hypertext and less use of help	Successful learners and proficient at using the system; no feedback required unless there are time constraints
	(4) poor achievement with moderate efficiency	frequent use of general text	Human intervention may be required to identify the causes of poor performance

CHAPTER 6. DISCUSSION

Introduction

The central research question addressed by this study was how to endow computerized learning systems with greater adaptivity. In order to be intelligently adaptive, a system needs to have a valid and useful model of users (Kearsley, 1987). To date, three disciplines have contributed to learner modeling, Educational Psychology, Cognitive Science and Artificial Intelligence, but there has been little cross-fertilization of concepts and methods across the three fields (Ohlsson, 1991). Intelligent learning applications have yet to demonstrate a level of adaptivity that approximates the effectiveness of one-to-one human tutoring interactions, leading some to conclude that the learner modeling problem is an intractable one (Lesgold, 1994; Orey and Nelson, 1992). Others circumvent the problem by maintaining that a learner model is not needed at all if one has a sufficiently rich environment for discovery learning (e.g., Sleeman, 1989). More recently, researchers have begun to question the need for deep, domain-based models of the learner that are capable of explaining the origin of all possible learning errors and suboptimal problem solving strategies (Collins, 1996). This is more compatible with advances in learning theory that advocate constructivist learning environments with an emphasis on the interaction between learners and learning materials (Cooper, 1993; Fox, 1994). As a result, this study addressed not only learning outcomes, in the form of posttest achievement, but also learning processes, in the form of learning environment options and paths selected. Finally, recent technological advances have made it possible to study complex and data-rich learning interactions more easily.

This study represents an attempt at improving learner modeling by providing a means of quickly assessing local usefulness of learner variables derived from the three disciplines in using data acquired during usage (i.e., during the learning interaction). The subset of variables extracted from the three disciplines was integrated into a systems approach to learner modeling to identify which, if any, proved to be useful to inlcude in a learner model and whether they showed any relationship to posttest score (achievement) and time on task (efficiency of learning). These variables, identified from *a priori* assessment tools, can be used as default values in a stereotype user model (Rich, 1983). The basic premise of this research was that the model can be updated using additional information from the learner's on-line learning behaviors, and as the learner's knowledge and skills evolve over time (Sollohub, 1989).

From Cognitive Science, tests of prior knowledge, time on task and motivational variables were selected and used to evaluate the instructional validity of the learning environment. From Educational Psychology, variables drawn from learning style instruments by Entwistle and Kolb were used to predict on-line learner behavior by using the theoretical constructs of the two respective learning theories. Those expected behaviors were then compared to actual learner behaviors manifested in the learner trace data in order to assess the local usefulness of these instruments. The field of Artificial Intelligence contributed the Artificial Neural Network technique (ANNs) to collect and analyze learner keystroke-level data was investigated with respect to its potential usefulness as an enabling technology for dynamic, real-time learner modeling using the integrated approach. In addition to achievement variance, the variables were assessed with respect to how well they distinguished between different patterns of learner behavior.

Summary of Major Findings

The study results, as summarized in the Results chapter (p.148), raise a number of issues with important instructional implications. The so-called cognitive science variables, prior knowledge and time on task, have traditionally been good predictors of learning achievement. The former should always be incorporated into a learner model; the latter a probable indication that the student is attending to critical, new content, this assumed to be at the heart of the instructional goals. As expected, these variables did indeed account for the majority of variance on a posttest of learning achievement. This result both confirmed the central role of these indices in any instructional design, and by their effect, helped to establish the validity of this study's instructional environment and materials. However, in the context of dynamic, computer-based instructional systems, added variables from educational psychology (learning styles) and artificial intelligence (pattern recognition trace data) may also prove useful for learner models.

Of the learning style variables, the investigation of the Kolb LSI did not provide any useful information in this particular learning context. However, the Entwistle ASI did prove useful: The ASI contributed to an explanation of posttest and gain variance and some expected learner profiles were observed. The behavioral analysis showed that students with a high reproducing orientation score made greater use of the self-test option (58% more time and twice as frequently), whereas high meaning orientation students made more use of notes (53% more time in notes, 2.3 times more frequently), as expected from their profiles. The reproducing and meaning orientation scores thus appear to be good candidates to use in a learner model, at least as default values, or starting points. The basic premise of this study has been, however, that computer-based systems can benefit from dynamic, individualized learner models which require additional learner data beyond such *a priori* categorization, i.e., trace variables. Analyses were thus extended into the domain of behavioral, pattern recognition strategies for learner propensities (i.e., AI-based trace data analysis).

The AI learner trace variables used duration and frequency data (extracted at the keystroke/location level) on each learner as they progressed through the five modules in the computerized learning environment. Of central importance to this thesis was the finding that the way in which students chose to cover the five modules (lesson order strategy) showed a clear relation with learning achievement. Students with a depth-first strategy had lower posttest scores. These data support learner-control studies which promote the continued use of sound, instructional design principles in designing and directing the learning process.

Further ANN-based pattern recognition analysis yielded a series of additional, interesting outcomes. The frequency with which certain options were used discriminated well between different learning patterns early on in the learning and the time students spent in each option became a useful parameter after they had interacted with the system. The best discrimination between learning patterns was achieved by using a combination of frequency and duration trace data. It appears that an analysis of trace data can pick out students who are "in trouble" fairly early on in the learning process and, after a certain amount of time has elapsed, it became easier to identify students who were most likely to show both good and poor learning achievement and learning efficiency. For example, the ANN identified a group of students who performed poorly (average score of 64% on the posttest; see Table 47) based on their early learning interactions. This learning pattern was characterized by students who made no use of the help and self-test options, who spent very little time in all options and who spent twice as much time in condense information as in general text. Another group of students learned well (80% on the posttest) in a short space of time (28 min.) and were characterized by having spent twice as much time in general text as in condense information. The key difference between these data and those generated by *a priori* measures is that they are responsive to the context (content, task demands, delivery medium, instructional design, etc.). Trace data generated as learners interact with a computerized learning environment can thus be used to adapt the learning environment to maximize learning achievement and learning efficiency based on the learning interaction pattern of each student.

Major Findings

Study I - Formative Evaluation

As described in the Methods section, both data and observation of procedures gathered from Study I were used as a pilot study and formative evaluation to prepare Study II. Data obtained in Study II were then used in the testing of research hypotheses. The principal changes included modifications to: the software interface, the instructional design, the course content, and the pretest and posttests used for the actual study. The software interface was simplified and improved in order to minimize the problem of learning to navigate through the course materials. The pedagogical design was modified through: the addition of self-test questions, an option to obtain supplementary information on any given topic, an option to defer taking the posttest, and the inclusion of more illustrative examples for each of the lesson topics. Finally, the pretest and posttest instruments were modified based on results of item analyses in order to provide a more accurate assessment of knowledge of the course content prior to and following interaction with the course materials (see detailed discussion of the pilot study in the Methods section and in Appendix B).

Study II - Experimental Sessions

The second study consisted of a total of 171 data sets collected over a two-year period. The sample population represented a multi-ethnic mix of students, who were predominately francophone or native French speakers. Biographical information obtained from the questionnaire showed that the majority (69%) were male. The vast majority had some knowledge of artificial intelligence (93%) but very little, if any, familiarity with neural networks (13%). Reasons given for participating in the study included personal interest, being required to, or a combination of the two.

Instructional Validity of the Learning Environment

One of the first questions addressed was whether the learning materials and learning environment used to collect the experimental data were instructionally valid. Time on task (total amount of time spent interacting with the learning materials) and learning (measured as the difference in scores between the pretest and the posttest) were used to establish the validity of the learning environment for the study. In particular, they were used to test whether the pretests and posttests were acceptable instruments, whether or not the different experimental groups were equivalent with respect to prior knowledge of content, and whether or not significant learning had taken place.

The pretest and posttest instruments were found to be valid and reliable. Tests of homogeneity confirmed that the pretest and posttest scores had normal distributions. Results showed that there was an acceptable level of discrimination for all items and an acceptable range of difficulty for both the pretest and posttest questions. Content validity was established due to the fact that the items were created by subject matter experts and all questions were related to specific learning content.

Group equivalencies were demonstrated for the six test groups used for data collection. There were no statistically significant differences between session groups for the overall pretest scores of the six sessions. Similarly, there were no significant differences for overall posttest scores for all sessions. Therefore the data from all six session groups could be combined.

The average time spent in on-task learning was 35.44 minutes with a standard deviation of 17.34. A test of significance of learning found that significant learning occurred with multiple choice items, with short answer items, and for the combined test items. The interaction time was therefore long enough to allow for significant learning to occur. The minimum number of interactions was calculated to be 35 (five modules with seven different options each) when all lessons are included and 28 without the first lesson visited. The average number of interactions exceeded these in both cases (57 and 45, respectively). The post-session interview data showed that most students felt they were able to learn and study the way they normally would have and all those interviewed found the exercise was a useful one (see Appendix D). Given that students were able to learn in this context and that the tests devised for this study were able to detect that learning had occurred, the instruments were deemed to be acceptable in testing the research hypotheses.

Contribution of Cognitive Science Variables

Three major variables from the cognitive science domain were first examined to assess what proportion of the posttest variance could be accounted for to assess their usefulness in providing default values for a stereotype user model. The three variables investigated were time on task, motivation and prior knowledge. A multiple regression with pretest, time on task and motivation variables was significant. However, only the pretest and time on task variables were found to be significant. The pretest accounted for 43% of the total posttest variance and the time on task accounted for 18%. The longer students spent interacting with the course materials and the higher they scored on a pretest of prior knowledge, the higher their achievement score on a posttest. These are traditional variables that are well tested and well researched. The learning environment and learning materials were thus instructionally valid; students could be said to have undergone a realistic learning task with sufficiently challenging material.

Student data on prior knowledge, which can be easily assessed through an *a priori* pretest, appears to be a useful element in a default stereotype learner model. This information can be used to select, for example, an appropriate starting point for the student. The system may recommend a lesson module but the student may or may not elect to follows the recommendation. This type of mixed initiative interaction can allow for adaptivity both on the part of the system and on the part of the student.

The time on task variable, which can only be assessed *a posteriori*, is also a useful learner model parameter. This parameter can be assessed on-line and added to the learner model during learning interaction (for example, the time spent on each lesson module) in order to update the learner model continually. Better methods of assessing student motivation appear to be required, as the single questionnaire item did not appear to be a valid measure of the student motivation and task perception.

Contribution of Educational Psychology Variables

Two learning style instruments were administered prior to learning in order to evaluate their predictive value within this particular learning context: the Kolb Learning Style Inventory (LSI) and the Entwistle Approaches to Studying instrument (ASI).

Kolb LSI. The first research question addressed was to assess the relative contribution of the Kolb learning style instrument to posttest variance, beyond what can be explained by the pretest and time on task variables. None of the Kolb data (dimensions, axes, groups) proved significant when introduced into a multiple regression together with the pretest and time on task variables. The *a priori* assessments of learners using this instrument do not appear to be useful in predicting achievement and do not provide any useful initial values of a learner model.

The Kolb LSI is based on the premise that learning styles represent stable long term traits. Kolb viewed learning styles as innate ways of learning and as such, they were not necessarily contextualized but rather more analogous to personality type. Researchers have expressed some contradictory views on this point: while some maintain that learning styles are neutral with respect to learning achievement (e.g., Messick, 1984), others have reported findings that showed convergers performed better in computerized learning environments (e.g., McNeal, 1986), particularly in hypermedia-based environments. (e.g., Esischaikul *et al*, 1994). This study was unable to find a correlation between higher achievement results and the converger group.

The second research question to be addressed was whether students exhibited the behaviors postulated by their responses to the Kolb LSI. Expected profiles were established for each of the four groups and compared to actual student on-line learning behaviors. None of the hypothesized learning behaviours were observed using the Kolb profiles with the exception of low positive correlations between a high RO score and the frequency of self-test as well as with time spent in the help option. The Kolb groups, axes and dimensions thus appear to have no predictive value in this learning context.

Whereas the posttest and gain scores measured learning achievement, the time on task variable could be considered to be an indicator of learning process (efficiency). However, this study showed no differences in time on task across the different Kolb profiles. No useful learner data could be obtained from the Kolb LSI despite the variety of ways used to treat the LSI data and the variety of methods used to analyse these data. This one-shot *a priori* measure, although used extensively in diagnosing learners to prescribe instruction, does not serve to make initial large-grained stereotypical models.

It was next hypothesized that perhaps the Kolb constructs had some validity but the instrument used did not adequately measure them. To this end, the behavioral variables were used to categorize students into four groups. This was done using conventional statistics (cluster analysis) and neural network-based pattern recognition. The third research question addressed was to assess the similarity between groupings generated by the Kolb instrument and groupings generated independently, through both a cluster analysis of trace data and neural network classification on the same learner trace data. No significant correlation was obtained between the four groups generated by the Kolb LSI and classifications generated using cluster analysis on trace data, nor with those classifications produced using a neural network. The neural network was used to place all students into four different groups, using the same trace data, and no significant correlation was obtained with the four Kolb groups. ANOVA analyses did not show any significant differences found for any of the trace variable means.

Therefore, the Kolb LSI data do not help differentiate learners with respect to how well they learned, how quickly they completed the learning task, nor the particular learning options they availed themselves of during the learning process,which lends further support to the variety of critiques on both the reliability and validity of the Kolb LSI that have been published. (e.g . Newstead, 1992; Stumpf and Freedman, 1981). Any further consideration of the Kolb LSI instrument in this context must therefore take one of two approaches. Either the LSI does not validly represent the underlying theoretical constructs, or, alternatively, it may be that the LSI does not in fact represent the theorietical learning constructs well, but that these constructs are not useful in creating an initial learner model for computer-based learning environments. This instrument thus appears to be a very weak candidate to establish default values for a stereotype learner model in this learning environment,

Entwistle ASI. The first research question addressed was whether the Entwistle groupings could prove useful in specifying the default values of a learner model and whether the ASI could help explain any of posttest variance beyond that accounted for by the pretest and time on task variables. This was done both for the Entwistle dimensions as assessed by the ASI and for the four quadrants that were created using the meaning orientation and reproducing orientation dimensions (which were created to be able to compare Entwistle groupings with Kolb and trace data groupings using cluster analysis).

A multiple regression with all eight Entwistle dimensions showed that only the reproducing dimension was significant, accounting for 22% of posttest variance. The reproducing dimension appears to be the most useful predictor of learner success in this environment, which is in keeping with previous research findings (Allinson, 1991).

A multiple regression with pretest, time on task and the Entwistle quadrant variables was also significant and all three variables were significant. The pretest variable explained 41% of the posttest variance, time on task 20% and the Entwistle quadrant 19%. The creation of four quadrants based on the Entwistle dimensions of reproducing orientation and meaning orientation thus appear to be useful predictors of student achievement in this learning context.

The Entwistle instrument yields eight sets of raw scores for each individual and does not group them in any way, as the Kolb LSI does. Previous research (Newstead, 1992), as well as this study, has shown that two of these dimensions, the meaning orientation and the reproducing orientation, are found to be more reliable and valid. It was therefore expedient to take these two dimensions and assign students to one of four groups, using an arbitrary cutoff score of 12 to distinguish between a 'high' and 'low' score on the two dimensions. These four quadrants enabled a more direct comparison of the Kolb and Entwistle instruments using cluster analysis. Stereotype user models necessarily require that learners be assigned to one of a few classes or categories as early as possible in order to adapt the subsequent learning experience. The use of these quadrants enabled an assessment of the Entwistle ASI as a good stereotype learner modeling tool to be made.

Students in the four Entwistle quadrants were found to differ in pretest and posttest scores. Students with low reproducing orientation scores and high meaning orientation scores (Quadrant II) had the lowest pretest scores and posttest scores. Students with both high reproducing and high meaning orientation scores (Quadrant IV) had the highest posttest scores, with concurrently high pretest scores. Students in the second quadrant had the highest gain scores. No significant differences were found for the time on task. Quadrant II students had significantly less prior knowledge of the content and consequently, showed significantly less learning. The difference in achievement for Quadrant II students may therefore have less to do with the Entwistle ASI constructs than with the fact that prior knowledge accounts for a large amount of achievement variance in learning.

The second research question addressed was to compare the expected learner behaviors, as extrapolated from their Entwistle ASI responses, to their actual learning behavior patterns, as detected by their learner trace data. Learner profiles were established for the two dimensions: reproducing orientation and meaning orientation. Students with a high reproducing orientation score are characterized as surface learners who are primarily motivated by acheivement. They tend to try to learn things by rote, especially if they perceive these items or concepts to figure prominently on any subsequent tests to be taken (Marton, 1981). In terms of on-line learning behaviours, these students were expected to make extensive use of the self-test and condense information options. They were not expected to spend too long in the overall learning task and to use a breadth-first lesson selection strategy.

As expected, a significant correlation was found between the reproducing dimension and time spent in self-test. The Entwistle reproducing dimension thus appears to be an interesting *a priori* predictor of learning activities. Students who have a high reproducing orientation, as assessed by the ASI, are much more prone to self-testing, which was expected, and also much less prone to seek help, which was not expected. Given that they predominantly used general text and self-test options, these students would not have required help in terms of explaining what the other options offered. The correlations were about the same both with and without the first lesson visited, which means that these learning profiles did not change substantially during the process of learning.

Some of the expected comprehension dimension learning behaviors were also observed. A significant correlation was found between a high comprehension orientation score and with time spent in notes and for time spent in the elaborate information option. The Entwistle meaning orientation thus also appears to have local predictive value in this learning context. The correlations were slightly greater when the first lesson visited was omitted, suggesting that it may take some time before learners are comfortable enough with the new learning environment to be able to learn in their preferred or habitual manner. Students who score high on the meaning orientation scale probably take some time to study the material using a deeper approach before they exhibit note-taking and elaborate information behaviors.

The Entwistle ASI thus appears to provide some useful information that can be used to initialize a stereotype learner model. In particular, student scores on the reproducing and meaning orientations can serve to classify students into one of four quadrants, each with a characteristic expected pattern of interaction with the learning environment and each with a different expected learning achievement and learning efficiency outcomes. It is possible to predict some specific learning environment options that will be selected by students in the different quadrants. This type of *a priori*, one-shot assessment can be used to populate a stereotype learner model template with starting values and these can serve as the basis for preliminary instructional decisions in adapting the learning environment to each group of students. For example, in selecting the best starting point or level of difficulty in test items for a given student. Once again, a duality exists in that both the learner and the system are capable of, and make use of, adaptivity.

Next, the four Entwistle groupings were compared to groupings obtained using both conventional cluster analysis techniques and neural network-based classification. This was done to address the third research question which was whether the learner trace data, independent of any ASI theoretical constructs, showed any variation across the four Entwistle quadrants. It was expected that there would be a correlation between Entwistle quadrants, based on reproducing and comprehension orientation scores, and trace variables.

The Entwistle quadrants showed some weak correlations with classifications based on lesson order strategy, classifications based on guidance preference duration data and cluster analysis classifications based on both frequency and duration data. The strongest correlation was with the cluster analysis classification based on duration trace data, after the first lesson visited was eliminated from the analysis. This finding is likely explained by the fact that most users faced with a never before encountered system will tend to explore the available options extensively at first, in order to become more familiar and comfortable in interacting with the system (Baecker *et al*, 1995). This type of 'novelty effect' may vary considerably from individual to individual - some may spend a few seconds exploring, others may spend considerably longer. In order to eliminate the effects of this interface exploration, the first lesson visited by students was omitted in its entirety and compared to analyses with all the lessons. This was a better way of dealing with the noisy data then eliminating a certain preliminary segment of time given the variability across individuals. The correlation was strengthened when the first lesson visited was omitted as the time devoted to learning how to use the interface was not taken into account. Student interactions with the system in subsequent lessons are much more likely to be focused on the course content than on the system interface and as such will yield more valid learner data.

Students with low reproducing and high meaning orientation scores (Quadrant II) differed from students in the other quadrants in that they spent much more time in the self-test option, as well as using the help option longer and more frequently than students in any of the other three quadrants. Students, with both high reproducing and high meaning orientation scores, had the lowest help use frequency (averaging once during the session). Students in the other two quadrants had about the same frequency of use and this was at an intermediate level between the other quadrants. No differences were found for learning time.

The pretest is used to classify students as either expert or novice with respect to the content of the learning materials (in this case, ANNs). The use of the help option during learning helps to classify students as novices or experts with respect to using the system and its features (the interface or navigational expertise). The Entwistle ASI adds to these two items of information by predicting whether students will be more likely to make use of notes or the self-test option, based on their responses to the ASI questions.

It appears that the Entwistle ASI is useful not only in predicting posttest achievement, along with pretest score and time on task, but it can also predict some specific learning behaviours, in particular, the use of the self-test and help options. The ASI can thus be used prior to learning and make use of reproducing orientation and meaning orientation scores to populate the stereotype learner model with some default data on the learner.

In summary, the educational psychology and cognitive science variables produce useful information but one has to be very careful in selecting the learning style instruments. The Entwistle ASI meaning and reproducing dimensions appear to be useful in this learning context and merit further experimentation.

Contribution of Artificial Intelligence Variables

The next research question addressed was to look at the classification of learners based on trace data. The values obtained for the trace variables were analyzed in order to identify learning behavior exhibited by the students. The relative contribution of trace variables to user model design was then analyzed. Three types of classifications were generated: groups based on guidance preference, groups based on lesson order strategy, and groups based on cluster analysis and neural network pattern classification of the learner trace data.

Classifications based on user preference for system guidance vs. learner control made use of two types of data: frequency of selection of a clear guidance preference, and the amount of time spent as a percentage of total time in a particular guidance mode. Each type of data was used to classify learners into one of three groups: (1) students with a clear preference for system guidance , where 'clear preference' was defined to be in excess of 75% of selections made or or of total time spent in this mode, (2) those with a clear preference for student guidance (in excess of 75% of the selections or duration spent in learner control), and (3) students who did not show a clear preference pattern.

Classifications were also derived from the way in which students chose to study the five lesson modules. Four types of lesson order strategies were defined: (1) students who exhibited a depth-first lesson selection strategy, as manifested by selection of two or fewer lessons for the first five lesson choices; (2) students who chose to look at four or more lessons within their first five lesson choices, thus exhibiting a breadth-first search strategy; (3) students who chose three lessons within their first five lesson choices and thus did not show a clear strategy preference; and (4) students who remained in system-guided mode throughout their interaction time and thus followed the system recommended lesson strategy.

Two parallel methods were used to classify learners based on their trace data: the first was statistical clustering and the second method used a Kohonen neural network to group students. Both methods used the same inputs to generate three classifications: one based on time data, one based on frequency data and one based on both time and frequency data. This was then repeated for the trace data after the first lesson visited was omitted.

The similarities between the different types of groups were analyzed first. Some correlations are to be expected as grouping methods were redundant (e.g., cluster analysis groups based on duration data and those based on frequency data). There was some correlation between the groupings produced by the constrained neural network classification and the groupings produced by a cluster analysis on the same data. This suggests that the neural network is classifying the data in a similar but not identical way. When the first lesson visited is omitted from the analysis, the correlation between the constrained and unconstrained neural network groups is quite high (r^2=0.78) which indicates that the neural network classification is likely to be more useful if samples of learner data are obtained after a certain amount of time has elapsed. This allows the system to exclude any behavior patterns that may be due to novelty effects of the learning environment. Alternatively, it may require a certain amount of time before useful learner differences are manifested in learning environments, as most learners may act in a similar way when using a learning tool for the first time.

Groups based on guidance preference. No significant differences were found on the pretest, posttest, gain and time on task variables for the groupings based on guidance preference (both using frequency data, duration data, for all lessons and when the first lesson visited was omitted). The groups based on guidance preference do not appear to explain any additional variance in the posttest, gain and time on task variables. This is probably due to the fact that the majority of students (78%) showed a clear preference for the system-guided mode which meant there was not enough diversity in learner behaviors.

Students in the different groups did show some differences in terms of their use of learning environment options. Students who showed a preference for guided mode spent less time changing their guidance mode, in elaborate information, condense information, notes and self-test. Their behavior was somewhat more passive or less active in that they most likely expected the system to take them through the various screens and options.

Students who preferred self-guided learning made significantly greater use of the elaborate information and notes options. This demonstrates a more reflective approach in that students spent a lot of time reading supplementary information on a given topic and making their own notes rather than interacting with the system (i.e. making a lot of selections within a given lesson).

Those students in a third group did not show a clear preference for either guidance mode. Consequently, they spent more time in changing their guidance mode. They also spent significantly longer in the elaborate information, condense information, notes and self-test options in addition to using the general text option more frequently. These students exhibited higher levels of interaction throughout their learning (i.e. they chose more options and chose options more frequently than students in the other groups).

When the first lesson visited was omitted from the analysis, the only significant difference across the three groups was the frequency with which help was used. Guided students used help an average of three times, unguided students seven times and those without a clear preference, nine times. This is expected as students following the guided mode benefit from the instructional design whereas students striking out on their own or those who kept switching back and forth will be more likely to require assistance.

Guidance preference appears to be related to a facet of learner behavior which may be described in terms similar to the learning style constructs: active, passive, and reflective. However, rather than describing long-term learner traits, these characteristics are most likely based on learner expectations of the role of the system. Once a student selects an initial guidance mode, they expect the system to behave in a certain manner and they may adjust their own learning behaviors accordingly.

Groups based on lesson sequence strategy. When the lesson sequence based groups were analyzed, no significant main effects were found for the pretest, gain and time on task variables. A significant main effect was found for the posttest scores of these groups. Students who exhibited a depth-first lesson order strategy had posttest scores that were significantly lower than students who chose to remain in system-guided mode throughout their interaction with the learning environment, both when all lessons were included and when the first lesson visited was omitted. It thus appears that the choice of lesson topic strategy was related to posttest achievement.

These groups were created in order to study student learning strategies which may prove to be similar to those strategies described as deep vs. surface learning (Marton, 1981, 1988) or holist vs. serialist learning (Pask,1972, 1976, 1988). As such, no differences were expected with respect to learning achievement between the depth-first and breadth-first strategies. Students who were guided throughout their learning learned better in this environment since they benefited from the instructional design. Students who adopted a depth-first strategy may not have covered all of the topics in sufficient detail in order for them to learn effectively, in terms of posttest requirements. This implies that some type of instructional or system feedback to students exhibiting this lesson order strategy may help them to maximize their learning success. This feedback could be in the form of prompts, hints or other types of advice that would encourage students to make sure they looked at all five lesson modules and not to spend too much time in the initial lesson topics selected. This feedback would be particularly warranted in the case of any time constraints on students.

Students in the lesson sequence strategy groups were found to differ with respect to only one trace variable - notes. Those students who chose to remain in guided mode throughout the learning interaction made much more extensive use of the notes option than students in the other groups. This small group of students, eight in total, thus exhibited distinct learning behaviors that led to high achievement in this learning environment, characterized by a preference for system guidance and reliance on self-generated notes during the learning process.

Cluster analysis classification based on trace variables. Trace variables were then used to classify learners using statistical cluster analysis. Only learner choices served as input into the cluster analysis (the *a priori* measures and achievement data were not used in generating the clusters). Three types of learner trace data were used and each generated four groups of learners: the duration data, the frequency data and combined frequency and duration data on all the learning environment variables. The classifications based on both time and frequency data showed the greatest number of significant differences in trace variable means for the groups, when all lessons were included in the analysis. This method of grouping students thus appears to be, as expected, more comprehensive than groups based on frequency data alone or on frequency data alone as this classification better captures both dimensions of student learning behaviours. The results from this combined grouping are discussed as follows.

One cluster of students (Group 1) consisted of learners who exhibited both high posttest scores and low time on task. Learners in this group interacted extensively with the learning environment but did not stay too long in any one option. In particular, they made less use of the guidance mode selection, elaborate information, condense information, general text, self-test and notes options than students in other groups. These students thus exhibited a "balanced" interaction pattern - they appeared to have spent roughly equivalent periods of time on all the options and showed approximately the same frequency of use of all the options offered by the learning environment.

Students in a second cluster also had a high posttest score but they spent longer periods of time, thus exhibiting lower learner efficiency. They spent longer in the elaborate information, condense information, and general text options. They also made more frequent use of the condense information option. Their learning behaviours thus showed a focus on content; they used those options which presented, summarized and expanded upon course content. The instructional implications for this group are that if learning efficiency is a requirement (i.e. there are time constraints) then these students should be given feedback to the effect that they should spend less time in the options they are selecting. If efficiency is not an issue, then these students should be allowed to proceed at their own pace as their pattern of interaction does lead to good achievement within this learning environment.

Students in a third cluster also did well on the posttest and had a time on task that was comparable to the second cluster. However, they were characterized by much greater frequency of use of the elaborate information, condense information, general text, notes and hypertext options. They did not necessarily spend a long time in these options, but over the course of their interaction with the learning materials, they selected these options with greater frequency than students in other groups. These students also made significantly less use of the help option. These students thus appear to be exploring the learning environment fully and with minimal assistance. They also made greater use of two options, notes and hypertext, which was not observed with the previous two groups.

Students in a fourth cluster group exhibited both poor achievement and poor learning efficiency. Their trace data showed a very different pattern of interaction than those found with the previous three groups. These students made little use of general text while making more extensive use of self-test and help options. These students also changed their guidance preference more often. Thus although these students spent just as long on the learning task, they spent that time in less productive options. They focused on achievement as evidenced by their high levels of self-test use and yet spent little time on the general text screens of the lessons.

The cluster analysis groups thus appear to provide additional information beyond that provided by the posttest and time on task variables beyond what can be obtained from a priori measures such as pretests and learning style inventories. These results demonstrate that learner interaction patterns with this learning environment can be assessed fairly rapidly (i.e. within a short period of interaction) and they appear to be useful in predicting learner achievement and learner efficiency. Trace data based on both duration and frequency information appear to discriminate well between different learner profiles. The critical variables distinguishing the groups appear to be the amount of time spent in general text, condensing information and hypertext, together with the frequency of help use, condense information and general text options. These data are useful in updating the existing stereotype learner model values (adding, deleting or modifying them) as well as in providing new values to "fill in the remaining blanks."

The findings related to the frequency and duration of help option use support earlier research that used neural networks to distinguish between experts and novices (Beale and Finlay, 1989, 1992; Sorenson, 1993). However, these experiments focused on domain expertise at a more global or macro level. Students were classified in a binary fashion - they were either novices or experts based on how much help they required throughout their entire interaction with the learning environment. The trace variables analyzed from this learning environment can be used to conduct a similar classification but in a more dynamic fashion and at a more micro or detailed level. Students may exhibit novice-like behavior or expert-like behavior and this may change depending on the lesson and a number of other factors. For example, students used help very frequently but sometimes only in a specific lesson or while studying a specific topic in that lesson. In the cluster analysis groups there were no significant differences in pretest scores of the four groups. Thus the cluster of students which had significantly lower achievement, did not simply consist of students who started off with less prior knowledge. Use of the help facility, together with patterns of usage of the other trace variables, appear to distinguish between students in a manner that is not limited to how much domain knowledge they may have, or have acquired.

<u>Neural network classifications based on trace data - Constrained neural network.</u> As was done with cluster analysis, learner choices (trace data) were used as inputs, this time into a

neural network pattern recognition system. Once again, no *a priori* measures, such as pretest or learning style instrument scores, were included in the input data set, nor were posttest achievement, gain scores and time on task. The constrained neural network was used to classify learners into a maximum of four groups based on their trace data. These groups were then analyzed with respect to learning achievement, efficiency and differences in learning patterns. The four groups of students were quite similar to the four groups found using cluster analysis in the previous section.

One group of students (Group 3) showed both high posttest scores and low time on task and were similar to the first group (Group 1) found using cluster analysis classification. These students spent less time spent in elaborate information, general text, and hypertext. They made infrequent use of condense information, general text and hypertext and they showed significantly more frequent use of the help and notes option.

Students in another group (Group 1) had high posttest scores with moderate time on task. This group is similar to the second cluster analysis group. These students spent longer in general text and made infrequent use of notes.

Students in Group 2 were quite similar to Group 3 as classified by cluster analysis. They too had high posttest scores but only moderate time on task. These students differed, however, in that they made more frequent use of condense information and hypertext, very infrequent use of help and they spent longer periods of time in hypertext.

Group 4 students were quite similar to Group 4 found using cluster analysis. They exhibited poor posttest scores. They spent significantly longer periods of time in elaborate information and showed a high frequency of general text. Unlike the cluster analysis Group 4, however, this group did not show significantly longer time on task.

Thus the four groups formed by a constrained neural network classification appear to explain some of the posttest variance and the time on task variance. The use of a neural network to form these groups appears to be at least comparable to groups obtained using the better known and more widely used technique of cluster analysis. Similar types of learner behaviors appear to be distinguished using both methods. It is possible to identify students who will spend significantly longer interacting with the learning materials. It is also possible to identify students who will do better and those who will not fare as well on the posttest. The greatest advantage offered by a neural network tool over a conventional statistical analysis package is that the neural network can collect and analyze these data during the process of learning. Statistical analyses, in contrast, must be done off-line. Neural networks thus offer the possibility of true dynamic modeling and systems that can adapt to the student in real time.

Unconstrained neural network. The analysis was repeated for the unconstrained neural network groups of which there were six. Two of these groups had very few individuals (four in Group 4 and six in Group 6) which may reflect outliers or highly idiosyncratic learner behaviors. If these groups of possible outliers are omitted, then the unconstrained neural network groups are similar to the groups obtained using the constrained neural network and, consequently, with groups obtained using cluster analysis.

Students in two of the neural network groups showed high posttest scores and low time on task. They were thus similar to one of the constrained ANN group (3) and one of the cluster analysis groups (1). Students in the first group were distinct in that they spent longer in the elaborate information option and less time in the notes option. Students in the second group spent most of their time in the general text option and little time in notes. Thus the unconstrained ANN differentiated between these two groups whereas the constrained ANN and cluster analysis simply lumped them itno the same category. These students exhibit different learning interaction patterns that both lead to comparable learning success.

Another group of students (Group 3) achieved a good posttest scores with moderate time on task but they differed in that they spent significantly more time in general text and showed a marked preference for the hypertext option. This group is similar to a constrained ANN group (1) and a cluster analysis group (2).

A separate group of learners again showed good posttest scores and moderate time on task but they exhibited a different interaction pattern: they spent much more time in notes and showed very infrequent use of the help option. This group is similar to two of the unconstrained neural network groups (1 and 2), and a cluster analysis group (3). These students preferred to personalize their learning experience by actively constructing their own annotations and summaries as they learned, through use of the notes option.

As with the constrained neural network group 4 and the cluster analysis group 4, one group of students showed both poor posttest scores and high time on task. These students spent very little time in general text, condense text and hypertext and hence, spent very little time interacting with the actual course content. The instructional implications for these students are to intervene as soon as this profile is detected as students exhibiting this pattern of learning behaviour need to be helped early on in the learning process. Feedback directing them to spend more time in learning the concepts presented in the general text options of the lesson modules would be particularly appropriate.

A final group of students had moderate posttest and time on task scores and did not appear to show any similarities with either the constrained ANN groups, nor the cluster analysis groups. These students spent less time in elaborate information and hypertext and more time in both general text and notes. This group appears to be a distinct group that is classified only by the unconstrained neural network. They appear to be attending to the course content but their time spent in interacting with the materials does not appear to be effective, nor particularly efficient. This group of students would likely benefit from direct human tutorial intervention, as soon as this profile is detected. This may help determine the cause of the difficulties learners are experiencing. The new information obtained by the human tutors can then be incorporarated for subsequent system learner diagnoses. Human intervention and "manual learner diagnosis" is required whenever a new category of learners is detected, or whenever an individual student cannot be classified as belonging to one of the existing groups. In this way, the more the system is used, the more its diagnostic capabilities (and adaptive capabilities) will be expanded and fine-tuned.

Summary. The unconstrained neural network did not differ substantially from the constrained one. This is primarily due to the fact that there are very few students in two of the six groups in the unconstrained network. These students likely represent outliers. In a constrained network, they were "made to fit" into one of the four categories. Both types of neural network classifications yielded groups with characteristics similar to those obtained using cluster analysis. The neural network approach to learner modeling thus appears to be a useful one, both for the assessment of local usefulness of existing measures and sources of information about the learner (such as learning style inventories) as well as providing information about actual learner behaviors.

The trace variables that were used in distinguishing between learner groups were the following: general text duration, help frequency and condense information duration using multiple regression analysis on all trace variables, together with pretest, time on task and the Entwistle reproducing orientation score. A forward stepwise regression of all trace variables identified six variables, including the same three from the multiple regression analysis, and in addition: condense information frequency, general text frequency and hypertext duration. The highest amount was explained when the first lesson is omitted and with the variables general text duration, help use frequency and time spent in elaboration of information. Thus trace variables appear to contribute to a small percentage of the posttest variance, after the pretest, time on task and Entwistle reproducing orientation variables are taken into account.

Summary of Major Findings

The contributions of selected Educational Psychology, Cognitive Science and Artificial Intelligence variables to learner modeling were assessed in a computerized learning environment. A reasonably large sample size was used, based on a two-year data collection period using six separate groups. This resulted in a large number of data points for each individual learner. The instructional validity of the learning materials and the learning environments was demonstrated to be sound, through formative evaluations using a pilot study, and through the correlations of pretest and time on task variables with learning achievement as measured by a posttest. Thus learners interacted with the environment for a period of time that was sufficient for significant learning to have occurred. Very sophisticated analyses were then conducted on this data in order to assess the relative usefulness of each type of learner variable and learner behavior. In addition, the potential usefulness of ANN-based learner modeling was assessed for dynamic learner modeling in real-time.

Certain learner behaviours, or profiles, were linked to higher levels of achievement and learning efficiency. It is therefore possible to predict how well students do on a posttest of achievement and roughly how much time they will take to learn, based solely on their learner trace data. In this way, the learning system can be more adaptive to learners as they are learning, or in "real-time." Conventional analytical tools such as cluster analysis can only be used once students have completed a learning session ("off-line" learner diagnosis). Artificial neural networks can diagnose as soon as enough learner behaviour trace data has been collected and the system can base instructions decisions at a micro-adaptation level (e.g., within a lesson module or even within an option in a given lesson module).

Learner trace data thus provided additional information that could serve as a basis for adaptivity. The ANN-based learning environment presents a much more controlled test situation and one in which much more information can be gleaned as students learn (as compared to, for example, tasks used to assess student learning styles, such as reading long sections of prose; Entwistle, 1983, 1988). It is much easier to follow exactly what students are doing and for how long using the learner trace facility and do so in only 35 minutes of interaction time.

These findings have several implications for learning theory. It may be possible to use an ANN-based learning environment to rapidly and reliably assess the contextual usefulness of learning constructs and specific instruments such as learning style or personality type inventories. Researchers can extract expected user profiles and map these onto specific learning behaviours that can be manifested in a computerized learning environment and then track student behaviours to observe whether or not these hypotheses were confirmed. These profiles can also be used to better design learning environments, both when a diversity of learning behaviours is desired (e.g., individualized learning) or when convergence is the goal (e.g., mastery skill learning). In this way, instructors can select the most appropriate *a priori* measures for their target student groups, for a given learning task, goal or content.

The ANN-based tool provides a great deal of flexibility in changing the *a priori* measures, and in changing the system design, including the interface. The results of the data analysis can also be used in the formative evaluation of the learning system and its interface.

Another potential contribution to advances in learning theory is to use the ANN-classified learner groups to identify new learning constructs in an empirical fashion. Those behaviours which distinguish between different, stable learner interaction patterns can be further investigated in order to identify the underlying cognitive processes that give rise to these types of learning behaviours. The advantage is that these behaviours are already operationally defined and can be easily detected in a computerized learning environment.

In addition to a conceptual advance in general system design, a number of implications emerge for the design of intelligent or interactive learning environments. The results show that in addition to valuable information from sources such as pretest of domain knowledge and instruments such as the Entwistle ASI which purport to measure learner-preferred approaches to studying, it is also possible to track actual student learning behavior during the process of learning.

Variables which proved to be useful in distinguishing students based solely on an analysis of their trace data were found to be: the help option, the self-test option, the note-taking option, general text and hypertext. When the first lesson visited is omitted in an attempt to eliminate any non-typical learner behavior due to e.g., novelty effect of a new system, four variables are found to be useful in predicting achievement: pretest score, time on task score, Entwistle ASI reproducing orientation score, and time spent in (or frequency of use of) the following options: elaborate information and condense information.

These are good candidates as parameters of a dynamic learner model that can be updated continually as students interact with the learning environment. The pretest score and Entwistle ASI reproducing dimension scores are good candidates for providing default values for an initial learner model (Sollohub, 1989). The latter is envisioned as a stereotype or best-guess model that provides the system with something upon which to base initial adaptive decisions. This represents an improvement over starting off with a blank slate and building up a model exclusively from keystroke-level information, which may be trying to do too much with too little information.

The best approach to learner diagnosis thus appears to be a combination of *a priori* assessment of prior knowledge, preferred approach to studying and demographic information together with contextual information on actual learner behaviours during interaction with the learning environment. The *a priori* measures can be used to establish a default stereotype learner model of each learner (, categories such as novice/achievement-oriented/science background or intermediate/deep-processing approach/administrative sciences major). Thus, in the early phases of learning, system guidance can be quite high for students who are novices with respect to the content and the particular learning environment. Instruction can be based on a classic Gagne (Gagne *et al*, 1988) hierarchy of prerequisite concepts in order to maximize learning effectiveness and efficiency for all students. This would be particularly appropriate for lesson one, which introduces the basic concepts and vocabulary associated with neural networks. Students could be guided towards mastery learning of this lesson module.

As learning progresses, which may be measured in elapsed learning time, number of lessons attempted or number of options selected within a given lesson, then the *a priori* measures may become redundant and they lose much of their predictive usefulness. Students evolve from novices to expert as their level of knowledge and skills improves. At this point, they may be more likely to prefer self-guided interaction with the system and begin to exhibit differences in learning interaction patterns. The contextual trace information, on the other hand, should remain continually up-to-date and relevant for instructional prescriptions in subsequent lesson modules. This diagnostic information may potentially be used to individualize instruction, through accommodation of some learner preferences and through challenges that help learners increase their repertory of learning skills, in order to not only maximize learning effectiveness and efficiency for each individual learner, but also to optimize the learning experience for each learner.

Trace data can be used to derive learner models that are much finer-grained than the initial stereotype models. Even under what may be termed near-ideal test conditions, i.e. a highly controlled environment where learning behaviors can be directly observed, the Kolb LSI did not have any useful predictive value and the Entwistle had some value. These instruments are thus insufficient for valid instructional responses to be made on an individual basis to learners. ANNs thus appear to be a useful enabling technology for learner modeling in adaptive learning environments. They provide a means of coping with the evolution of learner knowledge states and learner behaviors. Learner data collected and analyzed using the systems approach to learner modeling thus serves to increase the bandwidth between learners and the learning environments.

The major advantage with ANNs is their dynamic nature which makes them much better suited to endowing systems with adaptivity in real time. Learner classifications created using the ANN tool proved to be quite similar to those created using conventional cluster analysis techniques. This supports the use of ANNs to group learners since the ANN appears to be using the same, if not identical, criteria to assign learners to a given group. Thus the two techniques are comparable as they both make use of the same subset of learner trace data in discriminating between groups. The ANN method can potentially be used to derive and update individual, fine-grained, learner-based learner models that are contextually valid and useful in adapting the computer based instruction so that it can be tailored to each learner.

The results lead to the conclusion that it may be possible to guide learners not based on general stable predispositions measured before learning takes place, such as those postulated by Kolb, but rather based on context sensitive information that can be obtained from learner behavior traces as learners are interacting with the learning materials.

Weaknesses

This study was not an attempt at a comprehensive analysis of the three disciplines of educational psychology, cognitive science and artificial intelligence. A subset of variables was selected from each of the three domains but these may not have been representative of the field. Although the variables selected have been shown to be relevant in extensive previous research, other variables (e.g., the interface design) may be equally or more potent factors affecting learning effectiveness and efficiency.

While the number of data points yielded statistically established robust distributions, subjects tested tended to be quite homogenous with respect to age, educational background and prior knowledge of the subject matter. It would be beneficial to test out the same course content with a wider range of test subjects.

The pretest and posttest had only a limited number of items and two-thirds of these were multiple choice in format. A more sophisticated test of learner knowledge, such as an evaluation of an actual implementation of a neural network at the end of the five course modules, would be more effective. It would have also been useful to have longer tests and to have spent more time perhaps in face to face interviews to assess some of the data that was gathered using a questionnaire. In particular, the motivation of learners was not well assessed by a single question on the questionnaire. A better instrument or method of assessing the underlying motivations of students with respect to the learning task at hand is required.

Very few individuals took the opportunity to use the neural network editor that was supplied within the learning environment. It would have been useful to directly guide students to try this module and then assess the results of their work. This would have provided information on how well students were able to apply the knowledge they acquired in the course.

A single subject matter (Introduction to Neural Networks) was utilized for data collection purposes. The results would have been more generalizable had more than one instructional treatment and content been used with the test subjects.

The total amount of time that the learners spent logged on the system was a function of the time they spent interacting with the content, minus any time they spent in briefing sessions and in pause mode. This total time ranged from a few minutes to over four hours for some students, with an average of 35 minutes. Although at first glance this appears to be a short period of time, in computer-based environments it has been found that significant learning can occur within fairly short bursts of interaction with the system (e.g. Corbett *et al*, 1995). There is a minimum amount of time required to attend to the information presented on the screen. In the case of this study, this time may in fact be better represented in terms of number of options selected per lesson module (e.g. at a minimum, having looked at all seven options available for each module). There are also maximum limits which are typically a function of the physiological and cognitive factors such as fatigue and information overload (Baeker *et al*, 1990) Ideally, data collection should span several learning sessions spread out over time, such a semester, in order to ensure a more representative and more comprehensive sampling of learner behaviors. A longer data collection time period would also yield data on how learner patterns evolve with time, as learners gain more domain expertise and more familiarity with the learning environment.

The learning tool was a prototype system only - it would be valuable to implement the trace analysis facility within an existing course that has already been well established with respect to educational effectiveness and efficiency (for example, the LISP tutor).

Finally, there was a good deal of subjective inferencing required in establishing profiles of expected learner behaviors in this learning environment, as projected from the two learning style constructs, and secondly, in assessing whether or not students did in fact exhibit these expected behaviors. Descriptions provided by the authors of the learning style instruments were mapped on to the functions available within the learning environment (e.g. help, self-test) to the best of my ability. Despite the fact that this was a subjective exercise, the cluster analysis and the ANN produced theoretical constructs or factors that were similar to those proposed by the authors. The classifications of learners were particularly well-matched with the theoretical constructs published by Entwistle, namely, the meaning and reproducing orientation to study. Thus, in an independent fashion, it was possible to find students who exhibited reproducing orientations and meaning orientations with respect to how they undertook to learn in this learning environment. These weak but suggestive links between the subjective inferences used and statistical outcomes obtained indicate that further research using the ANN tool for user modeling is warranted.

The amount of posttest variance accounted for by this subset of variables was low and the strength of the relationships observed was low, due mainly to the fact that the study looked at data from a single learning session which had a fairly short interaction time. In addition, there was a significant amount of inferencing required to map potential learning style constructs to concrete learner behaviors expected in the learning environment. It would have been useful to collect data over several learning sessions in order to produce stronger correlations and explain a greater portion of the achievement variance. Despite these limitations, however, statistically significant effects were observed. Clearly, further research is required where extended interaction with the learning environment could be tracked in order to replicate and validate the nature of the phenomena observed in this study.

This research was largely heuristic in nature in that it contributes to the design, hypothesis-testing and the use of new technologies in learning. New techniques were tried out under unique circumstances in order to uncover new questions rather than to produce answers as such. The testbed was used by a fairly homogenous group of students, the instructional design and course content was not varied, students interacted with the system an average of 35 minutes, with some students having spent in excess of four hours on the system, and lesson modules all had the same set of seven learning options available to learners. Interesting results were obtained in spite of the limitations of the research design. A critical area for future research lies in the use of ANNs to extract learner characteristics and online learning behaviors for user modeling over time. The observed interaction patterns will either increase in strength, stabilize or disappear altogether as students progress through the learning environment. This prototype environment can thus serve as the basis for future testing of learning variables for inclusion in valid, contextual learner models.

Implications for Future Work

Learner modeling can be augmented through the use of artificial neural networks as it becomes possible to integrate a large number of variables and perspectives emanating from different disciplines into a single learning environment testbed. The learning environment that was designed and developed for this study can be used to test a wide variety of variables in a similar manner such as: different learning style inventories or other instruments such as the MBTI; with a wide range of learners; for different skills and subject matters; and with different instructional designs in order to provide a highly controlled environment for research on learner behavior in computerized learning environment.

Learner behavioral data can be collected in real-time, both to test specific theoretical constructs, such as those found in learning style theories, and to identify new constructs that merit further study. Validated trace data that can distinguish between different types of learners and which can also be related to how well students learn and/or the efficiency of their learning processes may some day render instruments such as the learning style inventories redundant.

A variety of different types of data can be obtained and prove to be useful about learners when building learner models. These data can be drawn from different fields, each of which may use different theoretical constructs and different instruments to assess them. This learning environment can be used to critically assess the different sources of learner data and integrate them in a meaningful fashion in a systems approach to learner modeling. A different mix of educational psychology, cognitive science and artificial intelligence variables may be best for different learning tasks. *A priori* assessments are almost exclusively based on domain knowledge assessment. A learner model needs to take into account the context as well as the processes of learning. This tool may prove to be a useful testbed to evaluate the relative contributions of different learner variables.

Learner modeling requires both domain-based and learner-based approaches. This research can help move computerized learning environments towards a more constructivist approach, one which entails a different way of assessing and modeling learners beyond *a priori* global and static measures such as a learning style type. Earlier and more contextual learner assessment is required if learning systems are to adapt to individual learners in any meaningful fashion. Some means of contextually valid learner diagnosis is required in order for the learning system to respond in real time and in an appropriate manner. Thus not only would learners be corrected, in the sense of a domain overlay model of expertise, but they would also be guided in a way that is best suited for learning to take place for that particular individual.

Future system design enhancements could include continual or adaptive domain knowledge testing (e.g. Desmarais *et al*, 1988). In this way, the evolution of learner skill and knowledge, along with learner interaction patterns, can be used to classify different types of learners and tailor instructional feedback accordingly. The dynamic learner model could thus be both domain-based and learner-based in order to take into account both learner knowledge states and learner approaches to studying.

In the context of future research, it may also be useful to look at motivational issues. Motivational states may be continually monitored and appropriate interventions undertaken to help maintain high levels of student interest in the learning materials and in pursuing their learning goals. It may be possible to monitor how active students are in order to detect waning interest, fatigue or diminishing morale as they perceive they are not learning well or not learning fast enough. It may be possible to identify such a thing as a boredom signature much the same way as task signatures have been identified using ANN pattern analysis on human-computer interactions. The learning system feedback can then take the form of asking the student whether or not he is tired and suggesting he take a break. The adaptivity of the learning system can thus be expanded to take into account cognitive learning interaction preferences and patterns, level and rate at which domain knowledge and skills are being acquired, and affective factors such as learner motivation. The ANN tool can be used to collect data over longer periods of time as it may be necessary for a certain amount of time to elapse before distinct learning patterns emerge and it is likely that these patterns will change with subsequent learning. It would be valuable to monitor changes in learner knowledge states, motivational states and learning behaviours over long periods to make use of them in flexible instructional prescriptions throughout the learning interaction (Langer and Bodendorf, 1995).

Since the data of this study were analyzed, the research team and I continued to do research on the data. The use of a Markov chain (see Appendix E) to analyze the sequence data from learner traces proved to be a useful technique and some interesting results were obtained. In general terms, this method permits the construction of a probabilistic decision tree which shows the probability that a given choice will be made, given a particular sequence of previous choices. Preliminary work done on the trace data generated by this study show that this method is a robust and valuable means of modeling learner behaviours, especially over longer periods of time, with a high degree of accuracy. Sequences of consecutive learner choices lead to predictions as to how likely it is that the learner will select one of the seven available options. For example, it was found that learners who chose to defer the posttest chose to look at general text over 90% of the time. This is a promising area for future research. Markov chain analyses of long-term learning behaviours can be used to investigate the stability or persistence of particular learning profiles (intra-individual variability). It would be possible to study how much they changed with different interfaces, different course content and different instructional designs (inter-individual variability).

Studies to collect data on learner behaviors over extended periods of time, such as a semester, could be carried out in order to revisit the principles of the Kolb Theory of Experiential Learning. In his theory, Kolb discusses the four stages a learner is expected to progress through, much in a Piagetian sense. The Kolb LSI may thus measure the current stage a learner is in. In addition, longer term studies would enable the observation of learner progression in terms of competence, from novice to expert, as they evolve or become more competent, both with respect to the subject matter (domain-related) and with respect to learning skills (learner-related).

Since the data for this study were collected (1992-1993), many researchers have begun to question the need for deep, sophisticated learner models to explain the improper reasoning and misconceptions behind each type of learner error (Carbanaro *et al,* 1995; Ragnemalm, 1996). Researchers are focusing on the interaction with the system rather than endowing the system with greater intelligence (Collins, 1996). This is consistent with the constructivist school of thought which maintains that the intelligence of an Intelligent Learning Environment lies in the interaction between the learner and the system. This also moves away from the old domain-based approaches to learner modeling. A different role is being carved out for adaptivity and individualization of instruction - one that moves away from being domain-based to one that is based on the learner and specific, contextual learner characteristics. The tools and techniques created and implemented in this study will help researchers and developers to operationalize this new strategy for learner modeling. The ANN-based tool and approach will enable researchers to create learning environments and learner models that are in line with current interest in adaptive constructivst learning environments.

REFERENCES

Allen, M. and Szabo, M. (1990). Dimensions of authoring aids intelligence. in, Proceedings of the Thrid International Computer-Assisted Learning Conference, ICCAL'90, Hagen, FRG. June 11-13, 1990. NY: Springer-Verlag.

Allinson, L. (1991) Learning styles and computer-based learning environments. In, P. Goodyear (Ed.) Teaching Knowledge and Intelligent Tutoring. Norwood, NJ: Ablex Publishing Corp.

Allman, W. (1989) Apprentices of Wonder. NY: Bantam Books.

Allport, G. (1961) Pattern and Growth in Personality. NY: Holt, Rinehart and Winston.

Anderson, J. (1984) Cognitive psychology and intelligent tutoring. Proceedings of the Cognitive Science Society Conference. Boulder, CO. pp. 37-43.

Anderson, J., Boyle, C.F., Corbett, A. and Lewis, M.W. (1990) Cognitive modeling and intelligent tutoring. Artificial Intelligence, 42, 7-49.

Anderson, J., Corbett, A., Koedinger, and Pelletier, R. (1995) Cognitive tutors: lessons learned. The Journal of the Learning Sciences, 4, 2, 167-207.

Atkinson, G. (1988) Reliability of the learning style inventory - 1985. Psychological Reports, 62, 755-758.

Atkinson, R. and Schiffern, R, (1968) Human memory: A proposed control system and its control processes. In, K. Spence and J. Spence (Eds) The Psychology of Learning and Motivation. NY: Academic Press.

Ausburn, L.J. and Ausburn, F.B. (1978) Cognitive styles: some information and implications for instructional design. Educational Communications and Technology, 26, 4, 337-354.

Ausubel, D., Novak, D. and Hanesian, H. (1978) Educational Psychology: A Cognitive View. Second Edition. Cambridge, MA: MIT Press.

Baecker, R., Grudin, J., Buxton, W and Greenberg, S. (Eds.) (1995). Human-Computer Interaction: Towards the Year 2000. Second Edition. San Francisco, CA: Morgan Kaufmann Publishers Inc.

Beale, R. and Finlay, J. (1989) User Modeling with a Neural System. York, UK: University of York.

Beale, R. and Finlay, J. (1992) Neural Networks and Pattern Recognition in Human-Computer Interaction. Chicester, UK: Ellis Horwood Ltd.

Beaumont, I. (1995) Adaptive educational hypermedia: from ideas to real systems. In, Proceedings of ED-MEDIA'95, World Conference on Educational Multimedia and Hypermedia. Graz, Austria. June 17-21.

Benyon, D. (1993) Accommodating individual differences through an adaptive user interface. In, M. Schneider-Hufschmidt, T. Kume and U. Malinowski (Eds.) Adaptive User Interfaces. North-Holland. pp. 149-165.

Benyon, D. and Murray, D. (1993) Adaptive systems: from intelligent tutoring to autonomous agents. Knowledge-based Systems, 6, 4, 197-219.

Bertel, K. (1994) A dynamic view on cognitive student modeling in computer programming. Journal of Artificial Intelligence in Education, 5, 1, 85-105.

Bierman, D. (1991) To be intelligent or not to be intelligent: is that the question? In, Proceedings CALISCE'91, International Conference on Computer-Assisted Learning in Science and Engineering. Sept. 9-11, 1995. Lausanne, Switzerland.

Bierman, D., Kamsteeg, P. and Sandberg, J. (1993) Student models, scratch-pads and simulation. In E. Costa (Ed) New Directions for ITS. Berlin: Springer-Verlag.

Bloom, B. (1984) The Two Sigma Problem: The search for methods of group instruction as effective as one-to-one tutoring. Educational Researcher, 13, 4-16.

Bonham, A. (1988) Learning style instruments: Let the buyer beware. Lifelong Learning, 11, 6, 12-16.

Bork, A. (1981) Learning with Computers. Bedford, MA: Digital Press.

Bork, A. (1986) Learning with Personal Computers. NY: Harper and Row.

Bostrom, R., Olfman, L. and Sein, M. (1990) The importance of learning style in end-user training. MIS Quarterly, 14, 1, 101-119.

Boyd, G. and Mitchel, P.D. (1992) How can intelligent CAL better adapt to learners? Computers and Education, 18, 1, 23-28.

Breuker, J., Ed. (1990) EUROHELP: Developing intelligent help systems. Final Report on the P280 ESPRIT Project EUROHELP. Brussels: EC.

Brown, J. and Burton, R. (1978) Diagnostic models for procedural bugs in basic mathematical skills. Cognitive Science, 2, 155-191.

Brown, J., Collins, A. and Duguid, P. (1990) Situated cognition and the culture of learning. Educational Researcher, 18, 1, 32-42.

Bruner, J. (1961) The act of discovery. Harvard Educational Review, 31, 21-31.

Bruner, J. (1966) Toward a Theory of Instruction. Cambridge, MA: MIT Press.

Brusilovsky, P. (1992) A framework for intelligent knowledge sequencing and task sequencing. In Proceedings of the Second International Conference on ITS, Berlin: Springer-Verlag.

Brusilovsky, P. (1994a) Student model centred architecture for intelligent learning environments. In Proceedings, Fourth International Conference on User Modeling, Cape Cod, MA, August 1994.

Brusilovsky, P. (1994b) Student Models in Intelligent Tutoring Systems. Workshop. UM'94. International Conference on User Modeling. Cape Cod, MA.

Burns, H., Parlett, J. and Redfield, C. (1991) Intelligent Tutoring Systems. Hillsdale, NJ: Lawrence Erlbaum Assoc.

Carbonaro, A., Maniezzo, V., Roccetti, M. and Salomoni, P. (1995) Modeling the student in Pitagora 2.0. User Modeling and User-Adapted Interaction, 4, 233-251.

Carey, T. (1995) Analysis of feature usage in Access to Online Information. In, W. Schuler (Ed.) Usability Issues in Design of Hypermedia Interfaces. Berlin: Springer-Verlag.

Carey, T. and Edwards, D. (1991) ANNs for analyzing user interactions. In, Neural Networks and Pattern Recognition in Human-Computer Interaction. R.Beale and J. Finlay (Eds). Ellis Horwood.

Carr, B. (1977) WUSOR-II: a computer-aided instruction program with student modeling capabilities. AI Lab Memo 417. Cambridge, MA: MIT.

Carr, B. and Goldstein, I. (1977) Overlays: a theory of modeling for computer-aided instruction. AI Lab Memo 406. Cambridge, MA: MIT.

Carr, C. (1988) Hypertext: A new training tool? Educational Technology, 28, 8, 7-11.

Carrier, C.A. and Jonassen,, D.H. (1988) Adapting courseware to accomodate individual differences. In, D.H. Jonassen (Ed.) Instructional Design for Microcomputer Courseware. Hillsdale, NJ: Lawrence Erlbaum Assoc.

Clancey, W. (1984) Classification problem solving. Proceedings National Conference on Artificial Intelligence. Austin, Texas. pp. 49-53.

Clancey, W. (1986) Qualitative student models. In, J.F. Traub (Ed.) Annual Review of Computer Science. Volume I. pp. 381-450. Palo Alto, CA: Annual Reviews, Inc.

Clariana, R. and Smith, L. (1988) Learning style shifts in computer-assisted instructional settings. In, Proceedings, Annual Meeting of the International Association for Computers in Education. New Orleans, LA. April 5-9. 1988.

Clemson, B (1984) Cybernetics. Kent, UK: Abacus Books.

Clowes, I, Cole, I. and Arshad, F. (1985) User modeling techniques for interactive systems. In, P. Johnson and S. Cook (Eds) People and Computers: Designing the Interface. Cambridge, England: Cambridge University Press.

Collins, A., Brown, J.S. and Newman, D. (1989) Cognitive apprenticeship: Teaching the craft of reading, writing and mathematics. In, L Resnick (Ed.) Knowing, Learning and Instruction: Essays in Honor of R. Glaser. NJ: Lawrence Erlbaum Assoc.

Collins, A. (1996) Keynote address. ITS'96 Conference. Montreal, Canada. June, 1996.

Cooper, P. A. (1993) Paradigm shifts in designed instruction: from behaviourism to cognitivism to constructivism. Educational Technology, 33, 5, 12.

Corbett, A., Anderson, J. and Patterson, E. (1990) Student modeling and tutoring flexibility in the LISP Intelligent Tutoring System. In C. Frasson and G. Gauthier (Eds.) Intelligent Tutoring Systems. Norwood, NJ: Ablex. pp. 83-105.

Cornwell, J. and Dunlap, W. (1994) On the questionable soundess of factoring ipsative data: A response to Saville and Willson (1991). Journal of Occupational and Organizational Psychology, 67, 89-100.

Cornwell, J. and Manfredo, P. (1994) Kolb's Learning Style Theory Revisited. Educational and Psychological Measurement, 54, 2, 317-327.

Craik, K. (1993) The Nature of Explanation. Cambridge, England: Cambridge University Press.

Cronbach, L.J. and Snow, R.E. (1977) Aptitudes and Instruction: A Handbook for Research on Interactions. NY: Irvington.

CTGV, The Cognition and Technology Group at Vanderbilt. (1992) The Jasper series as an example of anchored instruction: Theory, program description, assessment data. Educational Psychologist, 27, 3, 291-315.

Curry, L. (1987) A critique of the research on learning styles. Educational Leadership, 48, 2.

Daniels, P. (1986) Cognitive models in information retrieval: An evaluative review. Journal of Documentation, 42, 4, 272-304.

Davis, D. S. and Schwimmer, P.C. (1981) Style - a manner of thinking. Educational Leadership, 38, 5, 376-377.

Desmarais, M., Giroux, L, Larochelle, S. and Leclerc, S. (1988) Assessing the structure of knowledge in a procedural domain. Proceedings of the Cognitive Science Society, pp. 475-481. Montreal, August 14-17.

Driscoll, M. (1994) Psychology of Learning for Instruction. Boston, MA: Allyn and Bacon.

Dubs, S. and Jones, M. (1991) Student modeling for intelligent tutoring systems. ARC Technical Report. Calgary, Alberta: Alberta Research Council.

Duchastel, P. (1992) Integrating hypermedia into intelligent tutoring. In, Proceedings, Hypermedia Courseware Structures of Communication and Intelligent Help. NATO Advanced Research Workshop. Germany: Springer-Verlag.

Egan, D, (1988) Individual differences in human-computer interaction. In, M. Helander (Ed.) Handbook of Human-Computer Interaction. Amsterdam, The Netherlands: North Holland.

Elsom-Cook, M. (1990) Guided Discovery Tutoring: A Framework for ICAI Research. London, U.K: Paul Chapman.

Entwistle, N. and Hounsell, D. (1975) How Students Learn. Bailrigg, Lancaster: Institute for Research and Development in Post-Computing Education, University of Lancaster.

Entwistle, N., Hanley, M. and Hounsell, D. (1979) Identifying distinct approaches to studying. Higher Education, 8, 1, 365-380.

Entwistle, N. (1981) Styles of Learning and Teaching. NY: Wiley

Entwistle, N. and Ramsden, P. (1983) Understanding Student Learning. London, U.K: Croom Helm.

Entwistle, N. (1987) Understanding Classroom Learning. London, U.K.: Hodder Stoughton.

Entwistle, N. (1988) Motivational factors in students' approaches to learning. In R. Schmeck (Ed.) Learning Strategies and Learning Styles. pp. 21-51. NY: Plenum Press.

Esichaikul, V., Smith, R and Madey, G. (1994) The impact of learning style on problem-solving performance in a hypertext environment. Hypermedia, 6, 2, 101-110.

Finlay, J. and Beale, R. (1991) Neural networks and pattern recognition in human-computer interaction. CHI'91 Workshop. New Orleans, Louisiana.

Fischer, B.B. and Fischer, L. (1979) Styles in teaching and learning. Educational Leadership, 36, 4. 245-254.

Fogarty, T. and Goldwater, P. (1994) Increasing student control through an expert system. Journal of Research on Computing in Education, 26, 2, 238-255.

Fox, R. (1994) Integrating learning theory into interactive instruction system design. SALT Newsletter, Summer 94, p.1.

Frasson, C. and Gauthier, G. (1990) (Eds.) Intelligent Tutoring Systems. Norwood, NJ: Ablex. pp. 83-105.

Freedman, R. and Stumpf, S. (1978) What can one learn from the learning style inventory? Academy of Management Journal, 21, 2, 275-282.

Freitag, E. and Sullivan, H. (1995) Matching learner preference to amount of instruction: an alternative form of learner control. Educational Technology Reserach and Development, 43, 2, 5-14.

Gagne, R.M., Briggs, L. and Wager, W. (1988) Principles of Instruction Design. Third Edition. NY: Holt, Rinehart and Winston.

Gagné, R.M. (1989) Some reflections on learning and individual differences. In, P.L. Ackerman, R.J. Sternberg and R. Glaser (Eds.) Learning and Individual Differences: Advances in Theory and Research. NY: Freeman and Co.

Gegg-Harrison, T. (1992) Adapting instruction to the student's capabilities. Journal of Artificial Intelligence in Eduation, 3, 169-181.

Gentner, D. (1979). Towards an intelligent computer tutor. In H. O'Neil (Ed.) Procedures for Instructional System Development. NY: Academic Press.

Gentner, D. and Stevens, A. (1983) (Eds.) Mental Models. Hillsdale, NJ: Lawrence Erlbaum.

Genesereth, M. (1978) Automated Consultation for Complex Computer Systems. Doctoral Dissertation. Harvard University. Cambridge, MA.

Gery, G. (1991) Electronic Performance Support Systems. Boston, MA: Weingarten Publications.

Gilmore, D. and Self, J. (1988) The application of machine learning to intelligent tutoring systems. In, J. Self (Ed) Artificial Intelligence and Human Learning: Intelligent Computer Aided Instruction. London, U.K.: Chapman and Hall.

Glaser, R. (1984) Education and thinking: the role of knowledge. American Psychologist, 39, 93-104.

Goldstein, I. (1979) The genetic graph: a representation for the evolution of procedural knowledge. International Journal of Man-Machine Studies, 11, 51-77.

Goodyear, P. (1991) Teaching Knowledge and Intelligent Tutoring. Norwood, NJ: Ablex Publishing Corporation.

Guild, P. and Garger, S. (1985) Marching to Different Drummers. Alexandria, VA: ASCD.

Guilford, J.P. (1980) Cognitive styles: What are they? Educational and Psychological Measurement, 40, 715-735.

Greenberg, S., Darragh, J., Maulsby, D. and Witten, I. (1994) Predictive interfaces: What will they think of next? In, A. Edwards (Ed.) Extra-Ordinary Human-Computer Interaction: Interfaces for Users with Disabilities. Cambridge, U.K.: Cambridge University Press.

Hambleton, R. (1984) Criterion-referenced measurement. In, T. Husen and T. Postlethwaite (Eds.) International Encyclopaedia of Education. NY: Pergamon Press.

Hannafin, M. (1984) Guidelines for using locus of instructional control in the design of computer-assisted instruction. Journal of Instructional Development, 7, 3, 6-10.

Hannafin, M. (1992) Emerging technologies, ISD and learning environments: critical perspectives. Educational Technology Research and Development, 40, 49-64.

Hannum, W. (1990) The Application of Emerging Training Technology. San Diego, CA: University Assoc.

Harp, S., Samad, T. and Villano, M. (1995) Modeling student knowledge with self-organizing feature maps. IEEE Transactions on Systems, Man, and Cybernetics, 25, 5, 727-737.

Hartley, J. and Sleeman, D. (1973) Towards an intelligent teaching systems. International Journal of Man-Machine Studies, 5, 215-236.

Hativa, C. and Lesgold, A. (1991) The computer as tutor - can it adapt to the individual learner? Instructional Science, 20, 49-78.

Hawkes, L.W., Derry, S. J. and Rundensteiner, E.A. (1990) Individualizing tutoring using intelligent fuzzy temporal relational databases. International Journal of Man-Machine Studies, 33, 409-429.

Hayes-Roth, B. (1995) An architecture for adaptive intelligent systems. Artificial Intelligence, 72, 329-365.

Hill, R. W., and Johnson, W. L. (1995). Situated plan attribution. Journal of Artificial Intellience in Education, 6, 1, 35-66.

Holt, P. (1990) Intelligent tutoring systems in the real world. Interactivity, 1, 1, 18-20.

Holt, P., and Wood, P. (1990) Intelligent tutoring systems: a review for beginners. Canadian Journal of Educational Communications, 19, 2, 107-123.

Holt, P., Dubs, S., Jones, M. and Greer, J. (1994) The state of student modeling. In J. Greer and G. McCalla (Eds.) Student Modeling: The Key to Individualized Knowledge-based Instruction. NY: Springer-Verlag.

IAKE (1990) Knowledge Engineering Certification Exam Study Manual. Washington, D.C.: International Association of Knowledge Engineers.

Kaplan, R. and Rock, D. (1995) New directions for intelligent tutoring. AI Expert, 10, 2, 30-40.

Karunanithi, N. and Alspector, J. (1996) Feature-based and clique-based user models for movie selection. In, Proceedings UM'96, Fifth International Conference on User Modeling. Kailua-Kona, Hawaii. January 2-5, 1996.

Katz, S., Lesgold, A., Eggan, G. and Gordin, M. (1992) Modeling the student in Sherlock II. Journal of Artificial Intelligence in Education, 3, 4, 495-518.

Kearsley, G. (1987) Artificial Intelligence and Instruction. Reading, MA: Addison-Wesley.

Keefe, J. (1979) Learning styles: an overview. In, Student Learning Styles: Diagnosis and Prescribing Programs. NASSP.

Kobsa, A. and Wahlster, W. (1988) User Models in Dialog Models. Berlin: Springer-Verlag.

Kolb, D. (1971) Individual Learning Styles and the Learning Process. Cambridge, MA: MIT Press.

Kolb, D. (1976) Learning Style Inventory, Self-Scoring Test and Interpretation Booklet. Boston, MA

Kolb, D. (1977) Learning Style Inventory: A Self-Description of Preferred Learning Modes. Boston: McBer.

Kolb, D., Rubin, I. and McIntyre, J. (1979) Organizational Psychology: An Experimental Approach. (3rd edition). Englewood Cliffs, NJ: Prentice-Hall.

Kolb, D. (1985) Learning Style Inventory. Boston, MA: McBer and Company.

Lajoie, S. and Lesgold, A. (1992) Dynamic assessment of proficiency for solving procedural knowledge tasks. Educational Psychologist, 27, 3, 365-384.

Landa, L. (1976) Instructional Regulation and Control. Englewood Cliffs, NJ: Educational Technology Publications.

Langer and Bodendorf (1995) Flexible user guidance in multimedia CBT applications using artificial neural networks. In, Proceedings ICAANGA'95, International Conference on Artificial Neural Networks and Genetic Algorithms, Arles, France. June 18-21, 1995.

Laurillard, D. (1988) The pedagogical limitations of generative student models. Instructional Sciences, 17, 3, 235-250.

Lesgold, A. (1994) An assessment of intelligent training technologies. In E. Barker and H. F. O'Neil, Jr. (Eds.) Technology Assessment in Education and Training. Hillsdale, NJ: Lawrence Erlbaum.

Letteri, C.A. (1977) Cognitive style: implications for curriculum. In, A. Molnar and J. A. Zahovik, (Eds.) Curriculum Theory. Washington, D.C. ASCD.

Levine, D.S. (1991) Introduction to Neural and Cognitive Modeling. Hillsdale, NJ: Lawrence Erlbaum Assoc.

Livergood, N. (1991) From computer-assisted instruction to intelligent tutoring systems. Journal of Artificial Intelligence in Education, 2, 3, 39-50.

Logan, E. (1990) Cognitive styles and online behaviour of novice searchers. Information Processing and Management, 26, 4, 503-510.

London, R. and Clancey, W. (1982) Plan recognition strategies in student modeling: prediction and description. Proceedings of the National Conference on Artificial Intelligence, Pittsburg, Pennsylvania. pp. 334-338.

Maes, P. (1994) Agents that reduce work and information overload. Communications of the ACM, 37, 7, 30-40.

Maguire, R.B., Chan, C.W. and Cai, F. (1995). ACEDIS: a user adaptive diagnostic system. The New Review of Applied Expert Systems, 1995, pp. 49-64.

Marshall, J. and Merritt, S. (1985) Reliability and construct validity of alternate forms of the learning style inventory. Educational and Psychological Measurement, 45, 931-937

Marton, F. (1981) On qualitative differences in learning. I. Outcome and Process. British Journal of Educational Psychology, 25, 159-169.

Marton, F. (1988) Describing and improving learning. In R. Schmeck (Ed.) Learning Strategies and Learning Styles. pp. 53-87. NY: Plenum Press.

McNeal, G.H. (1986) The effect of agreement or disagreement of student learning style and method of instruction. Doctoral Dissertation. University of Washington.

McTear, M. (1993) User modeling for adaptive computer systems: a survey of recent developments. Artificial Intelligence Review, 7, 3-4, 157-184.

Merrill, M. (1983) Component display theory. In, C. Reigeluth (Ed) Instructional Design Theories and Models: An Overview of their Current Status. Hillsdale, NJ: Lawrence Erlbaum Assoc.

Messick, S. (1984) The nature of cognitive styles: problems and promise in educational practice. Educational Psychologist, 19, 59-74.

Messick, S. (1987) Structural relationships across cognition, personality and style. In R. E. Snow and M.J. Farr (Eds.) Aptitude, Learning and Instruction. Volume III. Conative and Affective Process Analysis. pp. 35-75. Hillsdale, NJ: Lawrence Erlbaum and Assoc.

Messick, S. (1994) The matter of style: manifestations of personality in cognition, learning, and teaching. Educational Psychologist, 29, 3, 121-136.

Michalski, R., Carbonell, J. and Mitchell, T. (Eds.) (1986) Machine Learning: An Artificial Intelligence Approach. Volume II. Los Altos, CA: Morgan Kaufmann.

Millbank, G. (1996) Developing simulation-based aptitude evaluation tools: a new aptitude measurement paradigm. In, Proceedings, LETT'96, Fourth Annual Leading Edge Training Technologies Conference. Victoria, B.C. February, 1996.

Mingall, H. (1995) Neural networks: learning how to learn. Computing Canada, October 25, 1995, p. 30.

Mukhopadhyay, S., Mostafa, J., Palakal, M., Lam, W., Xue, C. and Hudli, A. (1996) An adaptive multi-level filtering system. In, Proceedings UM'96, Fifth International Conference on User Modeling. Kailua-Kona, Hawaii. January 2-5, 1996.

Newby, R. (1994) Test-retest reliabilities of standard and realigned forms of the learning style inventory. Perceptual and Motor Skills, 78, 214.

Newstead, S. (1992) A study of two "quick-and-easy" methods of assessing individual differences in student learning. British Journal of Educational Psychology, 62, 299-312.

Nichols, P., Chapman, S. and Brennan, R. (1995) Cognitively Diagnostic Assessment. Hillsdale, NJ: Lawrence Erlbaum Associates, Publishers.

Norman, D and Draper, S. (Eds.) (1986) User-Centred Design: New Perspectives on Human-Computer Interaction. Hillsdale, NJ: Lawrence Erlbaum.

Ohlsson, S. (1991) System hacking meets learning theory: reflections on the goals and standards of research in artificial intelligence and education. Journal of Artificial Intelligence in Education, 2, 3, 5-18.

Ohlsson, S. (1992) Constraint-based student modeling. Journal of Artificial Intelligence in Education, 3, 4, 429-447.

Orwant, J. (1995) Heterogenous learning in the doppelganger user modeling system. User-Modeling and User-Adapted Interaction, 4, 2, 107-120.

O'Shea, T. and Self, J. (1983) Learning and Teaching with Computers. Englewood Cliffs, NJ: Prentice-Hall.

Orey, M.A. and Nelson, W.A. (1992) Development principles for intelligent tutoring systems: Integrating cognitive theory into the development of computer-based instruction. Education Technology Research and Development, 41, 1, 59-72.

Papert, S. (1980) Mindstorms: Children, Computers and Powerful Ideas. NY: Basic Books.

Park, S and Tennyson, R. (1983) Computer-based instructional systems for adaptive education: a review. Contemporary Education Review, 2, 2, 121-135.

Pask, G. and Scott, B.C. (1972) Learning strategies and individual competence. International Journal of Man-Machine Studies, 4, 217-253.

Pask, G. (1976) Styles and strategies of learning. British Journal of Educational Psychology, 46, 128-148.

Pask, G. (1988) Learning strategies, teaching strategies and conceptual or learning style. In R. Schmeck (Ed.) Learning Strategies and Learning Styles. pp. 83-100. NY: Plenum Press.

Payne, S. and Squibb, H. (1987) Understanding algebra errors: the psychological status of mal-rules. Technical Report number 43. Lancaster, England: Centre for Research on Computers and Learning, University of Lancaster.

Pearl, J. (1988) Probabilistic Reasoning in Intelligent Systems. Los Altos, CA: Morgan Kaufmann.

Piaget, J. (1954) The Construction of Reality in the Child. NY: Basic Books.

Posey, C.L. and Hawkes, L.W. (1996) Neural networks applied to knowledge acquisition in the student model. Information Sciences, 88, 275-298.

Ragnemalm, E. (1996) Student diagnosis in practice: bridging a gap. User Modeling and User-Adapted Interaction, 5, 93-116.

Rich, E. (1979) User modeling via stereotypes. Cognitive Science, 3, 355-366.

Rich, E. (1983) Users are individuals: individualizing user models. International Journal of Man-Machine Studies, 18, 199-214.

Rich, E. (1989) Stereotypes and user modeling. In A. Kobsa and W. Wahlster (Eds.) User Models in Dialog Systems. Berlin: Springer-Verlag. pp. 35-51.

Rickert, F. (1987) Intelligent interactive learning systems. Interactive Learning, 4, 3, 50-54.

Ridgway, J. (1988) Of course ICAI is impossible.....worse though, it may be seditious. In J. Self (Ed.) Artificial Intelligence and Human Learning: Intelligent Computer-aided Instruction. pp. 28-48. London: Chapman and Hall.

Sack, W., Soloway, E. and Weingrad, P. (1992) Re: Writing Cartesian student models. Journal of Artificial Intelligence in Education, 3, 4, 381-399.

Sandberg, J. (1994) Educational paradigms: issues and trends. in R. Lewis and P. Mendelsohn (Eds.) Lessons From Learning. Amsterdam, The Netherlands: Elsevier Sciences B.V. (North Holland).

Scandura. J. (1983) Instructional strategies based on the structural learning theory. In, C. Reigeluth (Ed) Instructional Design Theories and Models: An Overview of their Current Status. Hillsdale, NJ: Lawrence Erlbaum Assoc.

Schank, R. (1990) Tell Me A Story. NY: Macmillan.

Self, J. (1987) The application of machine learning to student modeling. In, R. Lawler and M. Yazdani (Eds.) Artificial Intelligence and Education. Norwood, NJ: Ablex.

Self, J. (1990) Bypassing the intractable problem of student modeling. In C. Frasson and G. Gauthier (Eds.) Intelligent Tutoring Systems: At the Crossroads of Artificial Intelligence and Education. NJ: Ablex Publishing.

Self, J. (1992). Computational mathetics: the missing link in Intelligent Tutoring System research? In, E. Costa (Ed.) New Directions for Intelligent Tutoring Sytems. Berlin, Germany: Springer-Verlag.

Sewall, T. (1986) Learning Style Inventory: David Kolb. The Measurement of Learning Style: A Critique of Four Assessment Tools, ED267247.

Shute, V., Woltz, D. and Rejian, J. (1989) An investigation of learner differences in an intelligent tutoring environment. in, D. Bierman, J. Breuker and J. Sandberg (Eds.) Artificial Intelligence and

Education. Proceedings of the Fourth International Conference on Artificial Intelligence and Education. May 24-26. Amsterdam, The Netherlands: IOS Publications.

Shute, V. (1993) A macroadaptive approach to tutoring. Journal of Artificial Intelligence in Education, 4, 1, 61-93.

Shute, V. (1996) Workshop on Student Modeling. ITS'96 Conference. Montreal, Canada. June, 1996.

Simon, H. (1995) The information-processing theory of mind. American Psychologist, 50, 7, 507-508.

Sims, R., Veres, J, Watson, P. and Buckner, K. (1986) The reliability and classification stability of the Learning Style Inventory. Educational and Psychological Measurement, 46, 753-760.

Skinner, B.F. (1968) The Technology of Teaching. NY: Appleton-Century-Croft.

Sleeman, D. (1989) Is a student model necessary? Apprenticeship as a model for ITS. In, D. Bierman, J. Breuker and J. Sandberg (Eds.) Artficial Intelligence and Education. Proceedings of the Fourth International Conference on AI and Education. Amsterdam, The Netherlands. IOS Publications.

Snow, R.E. (1989) Aptitude-treatment interaction as a framework for research on individual differences in learning. In, P.L. Ackerman, R.J. Sternberg and R. Glaser (Eds.) Learning and Individual Differences: Advances in Theory and Research. NY: Freeman and Co.

Sollohub, C. (1989) Stereotypes as a basis for student models in computer based tutoring sytems. In, Proceedings of The Sixth International Conference on Technology and Education, Volume I. Orlando, Florida. March, 1989.

Sorensen, B. (1993) Personal communication. Human Factors Division, U.S. Air Force Armstrong Laboratory, San Antonio, Texas.

Stacey, D., Calvert, D. and Carey, T. (1991) Artficial neural networks for analysing user interactions. In, R. Beale and J. Finlay (Eds.) Neural Networks and Pattern Recognition in Human-Computer Interaction. NY: Ellis Horwood Publishers.

Stanton, N. and Stammers, R. (1990) Learning styles in a non-linear training environment. In, R. McAleese and C. Green (Eds.) Hypertext: State of the Art. Oxford, UK: Intellect.

Stumpf, S and Freedman, R. (1981) The learning style inventory: still less than meets the eye. Academy of Management Review, 6, 2, 297-299.

Sun, C-T., Ling, Y-T., and Lin, F-X. (1995) Modeling hypermedia navigation: an artificial intelligence approach. Paper submitted for presentation at AI-ED'95, August, 1995.

Suppes, P. (1979) Current trends in computer-assisted instruction. Advances in Computers, 18, 173-229.

Tennyson, R.D., Christensen, D.L. and Park, S.I. (1984) The Minnesota Adaptive Instructional System: an intelligent CBI system. Journal of Computer-Based Instruction, 11, 1, 2-13.

Uhr, L. (1969) Teaching machine programs that generate problems as a function of interaction with students. Proceedings of the National ACM Conference. pp. 125-135. NY: ACM.

Ur, S. and Van Lehn, K. (1995). STEPS: a simulated, tutorable physics student. Journal of Artificial Intelligence in Education, 6, 4, 405-437.

van der Veer, G. (1990) Human-Computer Interaction: Learning, Individual Differences and Design Recommendations. Alblasserdam: Haveka.

Van Lehn, K. (1987) Learning one subprocedure per lesson. Artificial Intelligence, 31, 1, 1-40.

Van Lehn, K., Ohlsson, S. and Nason, R. (1994) Applications of simulated students: An exploration. Journal of Artificial Intelligence in Education, 5, 2, 135-175.

Vassileva, J. (1990) A classification and synthesis of student modeling techniques in intelligent computer-aided instruction. In, D. Norrie and H. Six (Eds.) Proceedings of the Third International Conference on Computer-Assisted Learning. ICCAL'90. Hagen, Germany. NY: Springer-Verlag.

Veres, J., Sims, R. and Locklear, R. (1991) Improving the reliability of Kolb's revised Learning Style Inventory. Educational and Psychological Measurement, 51, 143- 150.

Von Glaserfeld, E. (1988) An exposition of constructivism: Why some like it radical. Journal of Research on Mathematics Education, Monograph 4, 19-30.

Vygotsky, L. (1962) Thought and Language. Cambridge, MA: MIT Press.

Wenger, E. (1987) Artificial Intelligence and Tutoring Systems. Los Altos, CA: Morgan Kaufmann.

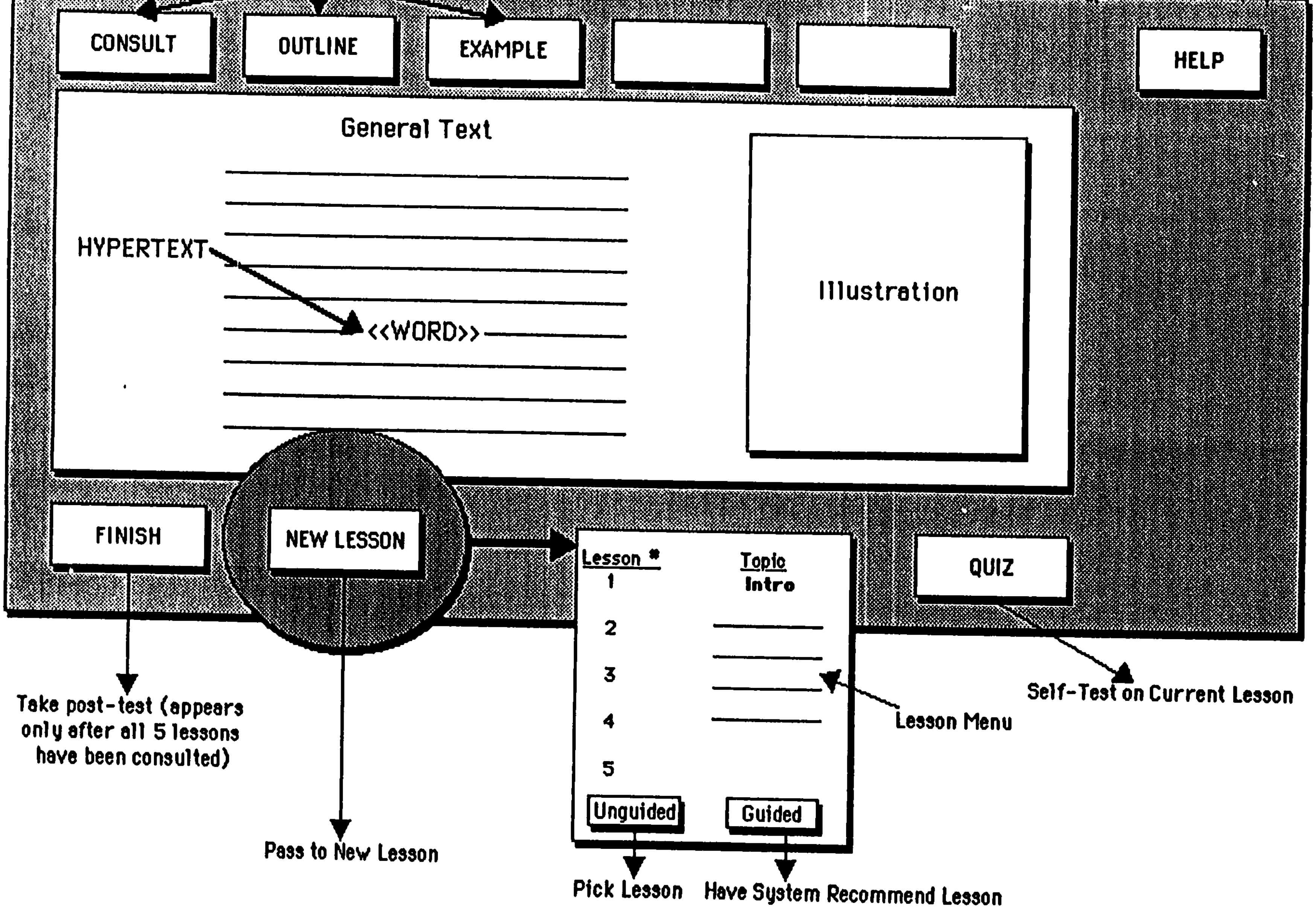
Wilkins, D., Clancey, W. and Buchanan, B. (1988) Using and evaluating differential modeling in intelligent tutoring and apprentice learning systems. In, J. Psotka, L.D. Massey and S.A. Mutter (Eds.) Intelligent Tutoring Systems: Lessons Learned. Hillsdale, NJ: Lawrence Erlbaum Assoc.

Winkels, R. (1990) User modeling in help systems. In, D. Norrie and H. Six (Eds) Computer-Assisted Learning. Berlin: Springer-Verlag.

Wohl, A. (1996) Keynote address. OSRA'96 Conference. February, 1996. Orlando, Florida.

Woolf, B. and Murray, T. (1992) Using machine learning to advise a student model. Journal of Artificial Intelligence in Education, 3, 4, 401-416.

Appendix A Explanatory User Interface Diagram



Interface Diagram to Explain How to Navigate Through Software and What the Menu Options Mean

Appendix B. Summary of Study I (Pilot Study)

Summary of Study I - Pilot Study Results

Sample size

The sample size for Study I was 69 with an attrition of 8, leaving 61 complete data sets. It was realized that both cluster analysis and ANN analysis required a much larger sample size and this was done for Study II.

Software

The software was modified following an ergonomic analysis of the interface design. The screens were made much more user-friendly and the look and feel was standardized by having options appear in the same order. Menus were made context-sensitive so as to reduce screen clutter. The sound option was turned off as students found it too noisy. The software was optimized so it ran much more rapidly, reducing waiting time for users. The entire code was debugged as system crashes had occurred periodically. Different versions of the system were created for different Macintosh platforms and the internal clock was weighted according to each type of computer's processing speed (in order to standardize the duration data collected).

Materials

The pretest and posttest were extensively modified. The original tests were multiple choice items only. Five short answer questions were added for Study II. The tests were tests of general knowledge only. This was changed so as to concentrate on course content presented in the modules. A ceiling effect was observed for the posttest. The average score on the pretest was 44% and that on the posttest 48%. Thus the tests were not adequately measuring the learning that took place. The average item difficulty was too high for posttest items (average of 37%) and the item discrimination index was less than 0.3 for over half of the test items. In Study II, a pool of 20 test items was created, two per lesson module. 10 items were randomly selected from each of the five pairs, to make up the pretest. Remaining items made up the posttest. Five short answer questions were selected from a pool of ten and assigned to the pretest. The remaining five were used on the posttest. An answer grid was developed for the short answer questions and more than one person was used to score the tests. In this way, the test instruments used in the second study were valid and could measure learning that occurred.

The course content was changed from a training course for knowledge engineers to a less ambitious goal of imparting the fundamental concepts of artificial neural networks in five computerized lesson modules. General content on artificial intelligence was eliminated. The course content was evaluated by subject matter experts and practitioners in the field of neural networks. The revised content was used for Study II.

The pedagogical design was modified based on student feedback and subject matter experts' suggestions. In place of a the self-evaluation screen at the end of each module, a quiz option was added. Students could attempt to answer a given question a maximum of three times. A 'Defer Posttest' option was added as students wanted to be able to decide when to take the final test and to go back and study some more if they wanted to before selected the posttest option. A number of screen options were eliminated or renamed based on student usage. Those options which students did not use at all were eliminated or regrouped with others (for example, dictionary and definitions). A better help facility was designed for Study II, based on questions that were (repeatedly) asked during the test sessions.

Procedures

Students were given the opportunity to sign up for a test session time slot, as opposed to scheduling everyone to a time. This gave them more flexibility and resulted in less 'no shows.' The questionnaire and learning style instruments were initially computerized. However, students spent too long on these and became tired of using the computer. As a result, for Study II, these were handed out to students a week prior to the actual test sessions. This streamlined the process and ensured that when students were on the computer, they had only one system and one interface to use and that this was related only to the course content.

A debriefing session was added to the beginning of the test session. It was found that many students did not understand the concept of a pretest and posttest. They became discouraged because they felt they should have known more of the answers on the pretest. A 15-minute debriefing script was prepared and used for each test session. The purpose of the exercise was explained, students were thanked for their participation and so on. This was done by myself, the system designer and the professor of the course. In addition, a diagram was put up on the board at the front of the room, to explain the various options available on the system and what each one did (see Appendix A). For Study II, the professors decided to give students five bonus marks for taking part in the testing. Finally, based on the level of effort that was required for Study I test sessions, many more volunteers were on hand to answer student questions during the test sessions for Study II. Each took notes on observations of students, such as problems with computers, consulting with other students, and they took part in the exit interviews.

Appendix C Category Codes for Questionnaire Data

Category Codes used for Questionnaire Data

Age

1 = less than 20
2 = 20-25
3 = 26-30
4 = 31-40
5 = over 40

Educational Background

1 = DEC or DES
2 = DEC or DES and one or more certificates
3 = DEC or DES, one or more certificates, and BACC or BAA
4 = anything other than the above

Major

1= MIS
2 = Science (pure and applied sciences, math)
3 = Computer Science (software engineering)
4 = Administration (commerce, administrative sciences)
5 = Finance (accounting, economic science)
6 = other (engineering, telecommunications, social sciences)

Program

1 = computer science, MIS
2 = management
3 = human resource management
4 = accounting
5 = other

Future Career Plans

1 = programmer/analyst, computer scientist
2 = finish studies, obtain degree, do a master's
3 = manager, executive, project head, administration
4 = start own company, consultant
5 = other (accountant, financial analyst, mutual funds manager, sales representative)

Reason for Participation

1 = personal interest (want to learn, interested in content, interested in the exercise, want to learn)
2 = because it is mandatory, because of the 5 bonus points for participating
3 = mandatory and personal interest
4 = other or no answer

Kolb LSI

1 = accomodator
2 = assimilator
3 = converger
4 = diverger

Entwistle Quadrants

	reproducing orientation score	meaning orientation score
1 =	less than 12	less than 12
2 =	less than 12	more than 12
3 =	more than 12	less than 12
4 =	more than 12	more than 12

Requested learning style results?

1 = yes
2 = no

Guidance Preference Group

1 = greater than 75% spent in guided mode
2 = greater than 75% spent in unguided mode
3 = no clear preference

Lesson Order Sequence Group

1 = depth-first (1 or 2 lessons selected within the first five lesson choices)
2 = breadth-first (4 or 5 lessons selected within the first five lesson choices)
3 = mixed (3 different lessons selected within the first five lesson choices)
4 = remained in guided mode throughout the interaction (100% guided)

Appendix D Transcripts of Exit Interviews

Exit Interview Checklist - UQAM tests

° Introduction: I would like to ask you some questions concerning your reaction to the system you have just used. This will only take 5 minutes of your time now or, if you prefer, I can take down your phone number and contact you at a later date. To speed things up, I would like to record this interview, unless you have any objections...

° What are some of the features you liked about the system?

° What didn't you like about it?

° What did you think about the learning style tests?

° When you study, what are some of your habits? (example: do you cram before a test? do you like to spread out your papers? do you like to read everything at least once?....)

° Were you able to study in your usual manner when interacting with this system? is so, how were you able to do so? (example: used the notepad option, etc..)

° in general, do you prefer a completely unconstrained environment or do you like to have some guidance in navigating through study materials?

NOTE: try to interview 3-5 people per session and try to get a range of completion times (i.e. don't wait until the end of three hours but talk to people who leave after an hour, 2 hours, etc.

Three individuals were selected at random and interviewed from each test session. A total of individuals were interviewed. These interviews were taped, when students gave their consent. All students were asked the following questions: How did you feel about the experience? Were you able to study in your usual manner? Do you feel you learned something about neural networks? The following are extracts from the interview transcripts (several of these comments were repeated by more than one student):

It was a difficult subject. One that is not very well known. Difficult reading as the themes were not our usual course themes.

I liked it very much and found the subject interesting.
I liked the way the software was structured as this is how I like to study. For example, I like to first read about the topic then see many examples.

I liked the course content very much. There was a lot of information. I feel I learned a lot although I can't remember all of the definitions. You would have to memorize them to really learn them

I would have like to have had a book along with the software.

I particularly enjoyed the applications section. I learned that neural networks could be applied to the domain of speech recognition.

You mean like at home? Yes, pretty much. It was like a tutorial class. I still had to go through to find all the important materials.

I found it too dense. There was too much to learn. Not enough time to go through everything.

I like to proceed in a certain way but I felt the software allowed me to do this so yes, I studied the way I normally would.

I liked it very much. I learned quite a bit. It was fun. I am an instinctive person. I liked to be able to jump around a bit in the software.

I liked that we had so much freedom and that I could choose where I wanted to go. I like the action - for example, I liked asking for the definitions of terms I did not know.

I was able to go through things rapidly.

I learned yes but I already knew about neural networks - some of it was quite new though. I would like to see how the technology improves in the future. Now, in voice recognition, for example, it still isn't 100% successful and many characters are not understood.

Appendix E Kohonen Neural Networks and Markov Chain Analysis

Description du réseau de neurones Kohonen
Jean-Francois Arcand

Résumé

Ce document présente l'architecture du réseau de neurones artificiels (RNA) utilisé dans le cadre du projet HIT. Deux algorithmes furent utilisés pour classifier les traces :

- la première tâche consistait à classifier l'échantillon de traces (169 traces) en 4 groupes. L'algorithme utilisé sera présenté à la section 1 de ce document ;
- la seconde tâche consistait à découvrir le nombre exact de groupes contenus dans l'échantillon de traces. L'algorithme utilisé sera présenté à la section 2 de ce document.

Comme l'algorithme de la section 1 de même que l'algorithme de la section 2 utilise comme la méthode de Kohonen comme algorithme d'apprentissage, un résumé de la méthode sera présenté en introduction.

Introduction « KOHONEN FEATURE MAP »

Cet article présente la théorie concemant les réseaux de neurones de type Kohonen. On y définira donc les fonctions d'entrée, les fonctions d'activation, les fonctions d'apprentissage et l'algorithme à utiliser avec les réseaux de Kohonen.

ARCHITECTURE DU RÉSEAU DE KOHONEN

Le réseau de Kohonen est composé de deux couches de neurones : une couche d'entrée et une couche de sortie, appelée Kohonen. La couche d'entrée envoie le vecteur des données d'entrée à chacune des neurones de la couche de Kohonen(figure 1). Ainsi, toutes les neurones de la couche de Kohonen ont reçu toutes les informations d'entrée. Le vecteur d'entrée est pondéré, à chaque neurone, par un vecteur de poids qui représente la force des liens entre le neurone et le vecteur d'entrée. Ensuite, les neurones de la couche de Kohonen font la compétition : le gagnant sera celui dont l'activation est la plus forte. Seul le neurone gagnant produira un signal de sortie. La compétition a lieu à travers les connections qu'il y a entre les neurones de la même couche (figure 1). Les neurones de la couche de Kohonen sont donc elles-mêmes interconnectées.

Pour être cohérent avec le fonctionnement du cortex cérébral, on devrait définir les interactions (i.e. les poids) entre les neurones en fonction de la distance entre chacune d'elles. En effet, dans le cortex, un neurone central qui envoie un signal excite son voisinage environnant avec des connexions positives. Au fur et à mesure que la distance avec le neurone central diminue, le degré d'excitation diminue jusqu'à devenir inhibiteur, négatif. Cette inhibition continue sur une distance déterminée pour revenir à une faible excitation positive à une distance considérablement éloignée du centre[4]. Cela donne une fonction en forme de chapeau de mexicain et c'est ce qui fait que les neurones « compétitionnent » entre eux (figure2). Cependant, nous utiliserons une approximation de cette fonction afin de simplifier les calculs.

Dans les réseaux de type Kohonen, on n'utilise pas une bonne réponse donnée afin de modifier les poids, on procède en se basant seulement sur les données d'entrée. C'est donc de l'apprentissage non supervisé, « unsupervised leaming »[5].

FORMALISME MATHÉMATIQUE :

Notons le réseau de neurones utilisé comme suit : Soit $A=(a_1,a_2,a_3..a_i...a_m)$ qui représente la couche d'entrée et $B=(b_1,b_2,b_3, ...b_j, ...b_n)$ qui représente la couche de Kohonen. Soit $W=(w_{11}, w_{12},... w_{ij},...w_{mn})$ où

w_{ij} représente le poids reliant le neurone i de l'entrée au neurone j de la couche de Kohonen, W_j=(w_{1j}, w_{2j},... w_{ij},...w_{mj}) représente le vecteur des connections de la couche d'entrée au neurone j de la couche de Kohonen et V=(v_{11}, v_{12},... v_{kj},...v_{mm}) où v_{kj} représente le poids reliant le neurone k de la couche de Kohonen au neurone j de la même couche.

FONCTION D'ENTRÉE

À la première itération, chaque neurone reçoit le même signal, le même vecteur de données d'entrée. Pour l'instant, on ne tient pas compte des connexions entre voisins, parce qu'à la première itération, ils n'ont pas encore réagit aux données. Ainsi, le neurone j calcule une valeur d'entrée :

$S_j = \Sigma_i w_{ij} a_i = W_j \cdot A$ où les w_{ij} sont les poids et a_i les entrées.

Ensuite, les neurones s'activent, envoient des signaux puis en reçoivent des autres neurones. L'entrée sera donc constituée des valeurs d'entrée du réseau et des signaux provenant des autres neurones. Pour le neurone j, on aura :

$S_j = \Sigma_i w_{ij} a_i + \Sigma_k v_{kj} b_k$ où les v_{kj} sont les poids des connexions entre les neurones k et le neurone j et les b_k sont les valeurs de sortie des neurones k.

Cependant, un seul neurone aura une valeur de sortie différente de zéro. La sommation du deuxième terme de l'équation sera donc réduite à un terme.

FONCTION D'ACTIVATION : LA COMPÉTITION

On doit maintenant trouver le neurone dont l'activation est la plus grande, le neurone qui répond le plus fort au pattern d'entrée. Ce sera le neurone dont le vecteur poids est le plus près du vecteur d'entrée. Il y a plusieurs façons de déterminer le neurone dit « gagnant » dont le produit scalaire et la distance euclidienne.

Le produit scalaire :

Selon Freeman et Skapura[4], l'activation d'un neurone est définit par l'équation suivante :

$$y_j = -r_j(y_j) + S_j + \Sigma_l z_{lj} y_l$$

où $r_j(y_j)$ est le terme d'oubli, souvent exprimé par λy_j où λ est une constante $\in [0,\infty[$. $\Sigma_l z_{lj} y_l$ représente les interactions latérales et z_{lj} est une fonction qui représente les liens entre les neurones de la même couche. Lorsque z_{lj} prend la forme d'un chapeau de mexicain (comme dans la figure 2), alors le réseau développera une bulle d'activité autour du neurone qui a la plus grande valeur d'entrée, nommé le gagnant[4]. Cette bulle correspond au voisinage qui sera définit dans les prochains paragraphes. L'activation est déterminée en majeure partie par le terme S_j. Donc, le neurone gagnant sera celui qui a la plus grande valeur d'entrée, qui correspond au produit scalaire entre les entrées et les poids[6].

On peut aussi expliquer l'utilisation du produit scalaire de la façon suivante : le neurone gagnant sera celui dont le vecteur poids est le plus près du vecteur d'entrée. Le produit scalaire donne la distance angulaire entre deux vecteurs lorsque les deux vecteurs sont normalisés :

$X \cdot Y = |X||Y|\cos\theta$ où θ est l'angle entre les deux vecteurs.

Plus le produit scalaire sera grand, plus la distance angulaire entre les deux vecteurs sera petite.
Notons le neurone gagnant b_g. Il est déterminé par :

$$W_g \cdot A = \max_j \{ W_j \cdot A \}$$

Sur la figure 3, le neurone gagnant serait le #2 car il est celui dont le vecteur poids a le plus petit angle avec le vecteur d'entrée.

Normaliser les deux vecteurs fait en sorte qu'ils sont inclut dans la même hypershère et dans le même espace dimensionnel (figure 3)[2]. Cela élimine aussi toutes dimensions que l'on aurait pu donner aux composantes des vecteurs.

Certains affirment qu'on peut aussi utiliser le produit scalaire lorsque seul le vecteur d'entrée est normalisé, car le vecteur poids tendra à le devenir au fur et à mesure des itérations[10]. Lorsque les vecteurs ne sont pas normalisés, le produit scalaire donne alors une mesure de la projection du vecteur poids sur le vecteur d'entrée. Cependant, si on recherche le vecteur poids qui est le plus près du vecteur d'entrée, alors le produit scalaire ne sera pas exact. En effet, si un vecteur poids plus grand était plus loin du vecteur d'entrée qu'un autre vecteur poids plus petit et plus court (figure 4), alors le produit scalaire pour le premier vecteur poids pourrait être le plus grand et le mauvais neurone (en pointillé) serait déclaré gagnant.

La distance euclidienne :

La distance euclidienne entre deux vecteurs est donnée par :

$$||X\text{-}Y|| = (\Sigma_i (x_i - y_i)^2)^{1/2}$$ où x_i et y_i sont respectivement les composantes des vecteurs X et Y

Cette formule donne la distance entre les bouts des vecteurs X et Y. Si on l'utilise pour déterminer le neurone gagnant, alors on calcule la distance euclidienne entre chaque vecteur poids et le vecteur d'entrée, et le neurone dont la distance est la plus courte est le gagnant (figure 5) :

$$||A\text{-}W_g|| = \min_j ||A\text{-}W_j||$$

Ainsi, sur la figure 5, le neurone gagnant est celui dont le vecteur poids n'est pas en pointillé, dont la distance avec le vecteur d'entrée est la plus courte. Il est donc intéressant de l'utiliser lorsqu'on ne normalise pas les vecteurs[4].

Lorsque les vecteurs sont normalisés, alors :

$$||A\text{-}W_j|| = ((A\text{-}W_j)^2)^{1/2}$$
$$= ((A^2 - 2\ AW_j + W_j^2))^{1/2}$$

On a $A^2 = A \cdot A = |A||A|\cos 0 = |A|^2$ = une constante, car A est normalisé. Il en est de même pour W_j. On obtient donc,

$$||A\text{-}W_j|| = ((\text{constante} - 2\ A{\cdot}W_j))^{1/2}$$

La distance euclidienne dépend donc directement du produit scalaire lorsque les deux vecteurs concernés sont normalisés. Dans ce cas, il sera donc équivalent d'utiliser le produit scalaire ou la distance euclidienne[5][10]. Gallant[5], utilise la distance euclidienne parce qu'il normalise le vecteur d'entrée et tous les vecteurs poids au départ mais ne s'occupe pas de renormaliser les poids à chaque itération.

Les choix de la normalisation des vecteurs, de même que le choix de la façon de calculer la distance entre les vecteurs, varient d'un problème à l'autre.

Seul le neurone gagnant produira une sortie de 1, alors que tous les autres neurones de la couche auront une valeur de sortie égale à 0[3][6].

FONCTION D'APPRENTISSAGE

Une fois qu'on a trouvé le neurone gagnant, on doit définir ce qu'on appelle le voisinage. Ce sera sur cet ensemble de neurones que se feront les changements de poids. Il peut être en carré, en losange ou en hexagone[9], et sa grandeur peut varier (voir figure 6). En fait, tous les neurones dont le vecteur poids, sont assez près du gagnant, sur l'hypersphère, feront partie du voisinage[5][1].

Quelque soit la forme de cet ensemble de neurones, il est généralement très grand au début et il diminue au fur et à mesure des époques d'apprentissage jusqu'à ne contenir qu'un certain nombre de neurones. On peut, par exemple, commencer en utilisant 80% du réseau et diminuer jusqu'à 6 neurones [1], ou diminuer jusqu'à ce que l'ensemble ne contienne que le neurone gagnant[4].

La forme et la grandeur de cet ensemble de neurone varient beaucoup. Maintenant qu'on a déterminé le voisinage, on modifie les poids des neurones de cet ensemble en les rapprochant un peu du vecteur d'entrée, de façon à ce que ces neurones répondent plus fortement au pattern d'entrée (Figure 7).

Ainsi on a :

$$\Delta w_{ij} = \begin{cases} \alpha(t)\,[a_i - w_{ij}] & \text{si } j \in \text{voisinage,} \\ 0 & \text{sinon.} \end{cases}$$

Le changement de poids du voisinage est une très grossière approximation de la fonction en forme de chapeau. Pour un temps donné t, α joue le rôle d'une constante et on modifie tous les neurones de l'ensemble selon la même proportion (figure 8a), plutôt que de diminuer α selon la distance avec le neurone gagnant.

On pourrait aussi, dans notre approximation, tenir compte des liens inhibiteurs et considérer un deuxième ensemble de neurones dont les vecteurs poids sont plus loin du vecteur d'entrée que ceux du premier ensemble. Il faudrait alors utiliser un autre taux d'apprentissage β, qui serait négatif cette fois.

$$\Delta w_{ij} = \beta(t)\,[a_i - w_{ij}] \quad \text{si } j \in \text{du 2}^{e} \text{ voisinage,}$$

On éloignerait donc un peu ces vecteurs poids du vecteur d'entrée. Ce serait une autre approximation de la fonction en chapeau (figure 8b).

Dans la fonction des changements de poids, $\alpha(t)$ est un taux d'apprentissage. Sa valeur représente la vitesse d'apprentissage du réseau de neurones. Pendant une itération, si la valeur de α est grande, les vecteurs poids se rapprocheront beaucoup du vecteur d'entrée. Ils leur faudra donc moins de temps pour être très près du vecteur d'entrée que si la valeur de α était petite. Les valeurs de α doivent être plus petites que 1, afin d'assurer la convergence, et ensuite diminuer pour tendre vers zéro en fonction du nombre d'itérations. Il y a d'ailleurs plusieurs façon de définir α. On a suggéré :

$\alpha = \varepsilon \exp\left(- \frac{\|W_j - W_g\|^2}{\sigma^2}\right)$ où ε est une très petite constante (près de zéro), W_j représente les poids du neurone j, W_g représente les poids du neurone gagnant et où la variance $\sigma^2/2$ contrôle le rayon du voisinage[10].

$\alpha(t) = \left(1 - \frac{t-1}{T}\right)$ où t = 1 à T, T représentant le nb total d'itérations [5].

$\alpha(t) = t^{-1}$ où t = 1 à 5000 ou 1 à 10000, (nous sommes à la t^{e} itération) [11].

$\alpha(t) = .2\left(1 - \frac{t}{10\,000}\right)$ où t = 1 à 5000 ou 1 à 10000, (nous sommes à la t^{e} itération) [11].

ou encore simplement une constante [3].

Une autre option pourrait aussi être offerte à l'usager : celle de considérer 2 phases lors d'une époque, comme suggéré par Haykin[7]. Une première phase dans laquelle se définit l'ordre des neurones, la topologie de la carte de Kohonen. Pendant cette phase, α diminue jusqu'à environ 0.1 en même temps que le voisinage rétrécit jusqu'à ne contenir que les derniers neurones. La deuxième phase, nommée phase de convergence [7], sert au renforcement du gagnant. La valeur de α continue à diminuer et le voisinage ne contient plus que le neurones gagnant et ses voisins immédiats.

ALGORITHME :

On doit premièrement choisir les fonctions et les paramètres utilisés : la formule de la distance, la normalisation des vecteurs, la forme, la grandeur de départ et la limite inférieure de la grandeur du voisinage ainsi que le rythme auquel il rapetisse et le taux d'apprentissage.

On doit aussi initialiser les poids du réseau en leur donnant des valeurs aléatoires comprises dans un intervalle choisi par l'usager.

Ensuite, on présente un pattern d'entrée au réseau. On trouve le neurone dont le vecteur poids est le plus du vecteur d'entrée puis on modifie les poids du gagnant et du voisinage. Les deux dernières étapes doivent être répétées jusqu'à ce le neurone gagnant soit assez renforcé, c'est-à-dire jusqu'à ce qu'il soit assez près du vecteur d'entrée. En fait, on peut s'arrêter quand α est rendu très petit [11], quand le voisinage s'est rétrécit à sa plus petite grandeur [4] et quand les changements de poids du neurone gagnant sont presque nuls [9]. Autrement dit, on s'arrête lorsqu'on est rendu à un état stable.

ALGORITHME :

1. **choisir les fonctions utilisées**
2. **initialiser tous les poids du réseau en utilisant une variable aléatoire prise dans l'intervalle [·,·] choisie par l'utilisateur**
3. **pour chaque pattern d'entrée, on fait :**

   {

   3.1 présenter le pattern au réseau

   3.2 répéter

   {

   3.2.1 calculer la distance de chaque neurone avec le pattern d'entrée

   $W_j \cdot A = |W_j||A|\cos\theta$ ou $||A-W_j|| = (\Sigma_i (a_i - w_{ij})^2)^{1/2}$ ou une autre fonction

   3.2.1 trouver le neurone gagnant

   $W_g \cdot A = \max_j \{ W_j \cdot A \}$ ou $||A-W_g|| = \min_j ||A-W_j||$

   3.2.1 ajuster les poids du gagnant et des neurones de son voisinage

   $\Delta w_{ij} = \alpha(t)\,[a_i - w_{ij}]$ si $j \in$ voisinage,

   0 sinon.

   3.2.1 tant que la grandeur du voisinage n'a pas atteint sa limite inférieure, diminuer la grandeur du voisinage, s'il y a lieu

   } jusqu'à ce que le réseau converge

   }

Section 1

Un neurone est ici représenté par un groupe. Le réseau de neurones utilisé possède une couche d'entrée représentant les variables de la traces, de même qu'une couche de sortie (couche Kohonen) représentant la classification, c'est-à-dire le groupe. Les poids entre la couche d'entrée et la couche Kohonen sont fixé par les valeurs des traces représentant le centre des groupes. Par exemple, une trace représenté par le vecteur (0.4, 0.3, 0.6 , 0.4) aura un réseau de neurones de la forme :

L'algorithme utilisé se comporte de la façon suivante :

1. Prendre aléatoirement 4 traces de l'échantillon. Ces quatre traces possèdent chacune un vecteur représentant les valeurs des interactions de l'utilisateur avec le logiciel HIT. Ces 4

traces représenteront les centres de gravité des 4 groupes. Ces vecteurs seront appelés *vecteurs centres.*

2. Fixer un seuil de voisinage. Pour l'expérimentation, des valeurs situées entre 0.2 et 0.8 ont été utilisées. Une vérification de la stabilité des groupes a permis de déterminer qu'un voisinage de ? ? ?
3. Pour chaque trace de l'échantillon
   - on mesure la distance entre le vecteur centre et le vecteur échantillon (distance Euclidienne, voir annexe A pour explication)
   - on obtient ainsi quatre distances, car il y a quatre vecteur centre. On choisie la distance la plus petite, et cette trace est classifiée dans le groupe représentant le vecteur centre
4. On obtient comme résultat final une classification en quatre groupes.

Section 2

L'algorithme utilisé se comporte de la façon suivante :

1. A priori, on ne possède aucun groupe. Chaque vecteur trace représente un groupe. On commence donc avec 169 groupes.
2. Fixer un seuil de voisinage. Pour l'expérimentation, des valeurs situées entre 0.2 et 0.8 ont été utilisées.
3. Pour chaque trace de l'échantillon
   - On calcule la distance entre les autres traces de l'échantillon. Si la distance entre la trace et les traces de l'échantillon est inférieure au seuil de voisinage, on regroupe la traces avec le même groupe que la trace de l'échantillon.
   - On réduit ainsi le nombre de groupe jusqu'à ce que ce stabilise l'algorithme. Parfois, cela peut prendre 4 entraînement au réseau de neurones pour ce stabiliser.

On obtient comme résultat final une classification en *n* groupes.

RÉFÉRENCES :

[1] Anderson, J.A., (1995), An Introduction to Neural Networks, The MIT Press, 650 p.
[2] Caudill, M., (1993), A Little Knowledge is a Dangerous Thing, dans AI Expert, november 1993, pp.16-22.
[3] Caudill, M., (1993), Kohonen Competitive Learning, in Neural Networks Primer, 3rd ed. By AI Expert, pp.39-46.
[4] Freeman, J.A., Skapura, D.M., (1991), Neural Networks : Algorithms, Applications, and Programming Techniques, Addison-Wesley Publishing Company, 401 p.
[5] Gallant, S.I., (1993), Neural Network Learning and Expert Systems, The MIT Press, 365 p.
[6] Gurney, K., Competition and self-organisation : Kohonen nets, Brunel University Uxbridge, 10 p.
[7] Haykin, S., (1994), Neural Networks ; A Comprehensive Foundation, IEEE Press, McMaster University.
[8] Hiotis, A., (1993), Inside a Self-Organizing Map, dans AI Expert, april 1993, pp.38-43.
[9] Lin, X., Soergel, D., Marchionini, G., (1991), A Self-organizing Map for Information Retrieval, College of Library and Information Services, University of Maryland, 7p.
[10] Ritter, H., Self-Organizing Feature Maps : Kohonen Maps, dans The Handbook of Brain Theory and Neural Networks, (1995), pp.846-851.
[11] Simpson, P.K., (1990), Artificial Neural Systems, Pergamon Press inc., 183 p.
[12] The Handbook of Brain Theory and Neural Networks, (1995), Michael A. Arbib ed., The MIT Press, 1118 p.

Markov Analysis

Jean-Francois Arcand

> « A Bayesian network is a directed acyclic graph in which each node represents a random variable, that is, a set of mutually exclusive and collectively exhaustive propositions. Each set of arcs into a node represents a probabilistic dependence between the node and its parents (the nodes at the other ends of the incoming arcs). A Bayesian network represents, through its structure, the conditional independence relations among the variables in the network. These independence relations provide a framework within which to acquire probabilistic information. The conditional independences between variables also render inference tractable in a large number of real-world situations. » [Fung, Del Favro, p.]

The basic idea behind models such as Bayesian networks and Markov chains is to organize propositions into graphs of elements interrelated by probabilities representing their degrees of dependency. These networks provide a formal way to organize related evidences into graphs. Such graphs are easier to understand as compared to analyzing all the pieces of evidence in an independently. The main difference between the Bayesian networks and the Markov chains lies in the meaning of the arcs linking the nodes. A Markov chain, sometimes called a Markov network, is an undirected graph whose links represent symmetrical probabilistic dependencies between the linked nodes. On the other hand, a Bayesian network is a directed acyclic graph with arrows representing causal influences or class-property relationships. [Pearl]

Lets take the example of three events, A (interactor 1, e.g. a *General* button), B (interactor 2, e.g. a *Outline* button), and C (interactor 3, e.g. a *Consultation* button), and a relation r taking place among these interactors. Within a Markov Chain, r_N represents a symmetrical relation between the nodes being connected through. The value given to the arcs representing the r_N relationship can be interpreted as the probability level associated to the relation for the linked pair of events. For example, the probability of co-occurrence of two events A and B ($r_N(AB)$). (possible value of $r_N \in [0,1]$) A possible Markov chain expressing the dependancy relation r for the given set of events is:

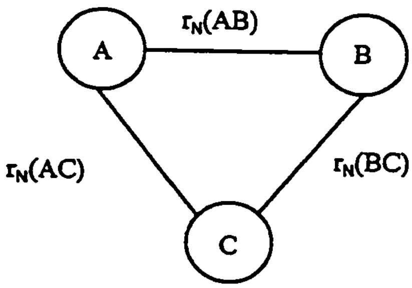


Figure 1.

If the symmetry principle does not hold for the type of relation being expressed, then Markov chains cannot be used. However, such relations can be represented by Bayesian networks. Within the project HIT, we could have symmetrical relation as well as non-symmetrical relation. Possible Bayesian networks representing a non-symmetric relation r_{NS} for our set of nodes are:

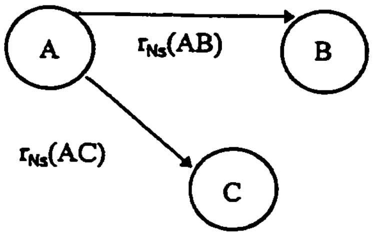


Figure 2.

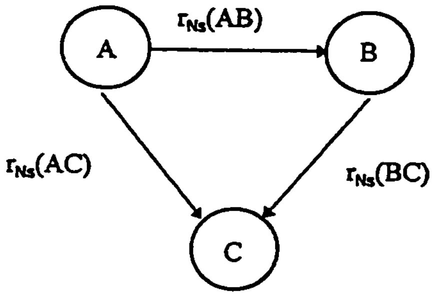


Figure 3.

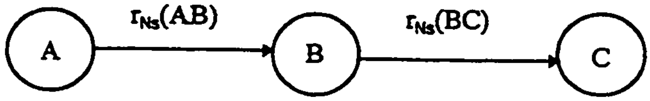


Figure 4.

Two observations are valid for all the three Bayesian networks presented in Figures 2 to 4: 1) all of their arcs have a direction, and 2) none of the networks contain cycles, which means that one cannot start from a node and return to it by following the arcs.

In both types of networks, Markov and Bayesian, the values attributed to the links are based on observations of how strongly the relation represented by the arc holds for a particular pair of nodes. In essence, such values can be based on statistical evidence, thus representing the likelihood attributed to r in regards to the linked nodes. For example, lets say that the non-symmetrical relation $r_{Success\text{-}rate}$ is represented by a Bayesian network, $r_{Success\text{-}rate}(\phi^{\beta}i, \phi^{\alpha}k)$ representing the probability of success of $\phi^{\beta}i$ given that it hires $\phi^{\alpha}k$, such a value being based on statistical evidence cumulated from past interaction between the two agents. Furthermore, let us say that the agents keep a cumulated value for their probability of success in all given context (the total probability or prior probability expressed as $o_{success}(\phi^{\beta}i)$ in section 4.1.3) From the expression of these likelihoods and prior probabilities, posterior probabilities can then be calculated. In effect, from the chain rule of basic probability theory it is established that every distribution function $P(x_1, \ldots, x_n)$ for a given relation r can be represented as a product:

$$P(x_1, x_2, \ldots, x_n) \quad = \quad P(x_n|x_1, \ldots, x_{n-1}) \ldots P(x_2|x_1)\, P(x_1)$$

Let us say that the following Bayesian network represents the relation $r_{success\text{-}rate}$ as described before between the three interactors ϕ^{β}_{i}, ϕ^{α}_{k}, ϕ^{ϖ}_{g}:

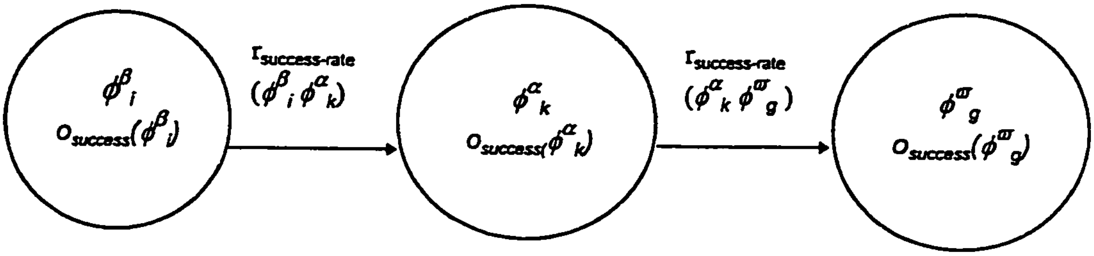


Figure 5.

From this network, we can then calculate the probability of success of ϕ^{β}_{i}, given that ϕ^{β}_{i} is followed by ϕ^{α}_{k} which in turn hires ϕ^{ϖ}_{g}.

$$P(H|e) = \frac{P(e|H)P(H)}{P(e)}$$

$$P(\text{success of } \phi^{\beta}_{i} \mid \text{success of } \phi^{\alpha}_{k}\phi^{\varpi}_{g}) = \frac{P(\text{success of } \phi^{\alpha}_{k}\,\phi^{\varpi}_{g} \mid \text{success of } \phi^{\beta}_{i})\; P(\text{success of } \phi^{\beta}_{i}) =}{P(\text{success of } \phi^{\alpha}_{k}\phi^{\varpi}_{g})}$$

$$= \quad P(\text{success of }\ )\; o_{success}(\phi^{\beta}_{i})$$

Markov's Analysis in HIT

This function allows the analyst to perform a Markov-type analysis. The analyst can find out about particular chains of operators that were activated. The analyst can add any operator to the existing one to form a chain by clicking on the selected operator from the top row. The chosen operator will be added before the ones found in the first column. The table is then refreshed and will display the number of chains of interactions formed with the new sequence shown in the first column preceded by the operator in the top row. The next operators to be added to the chain in the first column *is inserted to the left of the original operator.* The resulting analysis can be saved as a file in the project (See figure 6).

Appendix F. Pretest and Posttest Instruments

Nom:_______________________ Matricule:__________________

Pré test

Ne noircir qu'une seule réponse par question

1. Quelle caractéristique les systèmes «naturellement intelligents» possèdent-i

O A. raissonnement dit par association
O B. calculs efficaces
O C. traitement rapide d'un grand volume d'information
O D. aucune de ces réponses

2. Lequel parmi les termes suivants a le même sens qu'un réseau de neurones?

O A. système d'aide à la décision
O B. système expert
O C. système à base de connaissances
O D. système connexionniste

3. Laquelle parmi les suivantes est une propriété des réseaux de neurones?

O A. ils justifient leur solution à un problème
O B. ils fournissent une réponse pour un échantillon de données qu'ils n'ont jamais rencontré
O C. ils fournissent une réponse pour un échantillon de données incomplet
O D. b et c

4. Quelle est l'une des plus grandes lacunes des réseaux neuronaux?

O A. les algorithmes d'apprentissage ne sont pas très puissants
O B. ils ne s'appliquent qu'à des domaines possédant un patrimoine d'expérience varié et étendu
O C. la capacité de traiter les connaissances floues est très faible
O D. a et b

5. Pourquoi dit-on que les reseaux de neurones sont intelligents?

O A. ils ont la capacité de générer leurs propres règles
O B. ils traitent l'information avec dynamisme et auto-organisation
O C. ils sont capables d'apprendre à partir de leurs propres expériences
O D. toutes ces réponses sont bonnes

60%

30/50 = R 40%

M

PRE

6.Les poids associés aux interconnexions: R

O A. sont des nombres réels
O B. servent à donner l'emphase sur certains signaux d'entrée en multipliant leur valeur
O C. servent à minimiser l'emphase de certains signaux d'entrée en multipliant leur valeur
O D. toutes ces réponses sont bonnes

7.Quel est la première étape dans la modélisation des réseaux neuronaux?

O A. préanalyser les données
O B. évaluer la taille du problème à résoudre
O C. diviser le problème à résoudre en sous-problèmes
O D. vérifier si le problème à résoudre est bien adapté à une résolution par réseaux de neurones

8.L'algorithme d'apprentissage: M

O A. nécessite un très grande nombre d'échantillons de données
O B. est un processus d'amélioration progressive par essais et erreurs
O C. est capable de fournir une réponse aux cas qu'il n'a jamais rencontrés
O D. toutes ces réponses sont bonnes

9.Les applications fructueuses de réseaux de neurones ont généralement les caractéristiques suivantes: M

O A. une partie des données est incomplète
O B. les données contiennent des erreurs
O C. les humains ne savent pas comment résoudre le problème
O D. a et b

10.Lequel parmi les suivants pourra être une application de réseaux de neuron

O A. la reconnaissance de caractères
O B. le traitement de demandes de crédits
O C. l'établissement d'horaires de vols pour une compagnie aérienne
O D. toutes ces réponses sont bonnes

Donner des réponses courtes et précises pour les cinq prochaines questions

11. Qu'est-ce qui distingue les réseaux de neurones des systèmes à base de connaissances en ce qui a trait au stockage et au traitement des connaissan

12. Comment définiriez-vous le concept «réseau neuronal»?

13. Décrivez la structure et le fonctionnement d'un réseau neuronal

14. Quelles sont les étapes de modélisation d'un réseau neuronal?

15. Quels sont les critères à partir desquels on peut juger si un problème est propice à l'utilisation des réseaux de neurones?

Post test

Ne noircir qu'une seule réponse par question

1.L'approche informatique conventionnelle s'adresse surtout aux:

O A. traitements rapides d'information
O B. calculs qui dépassent les capacités des humains
O C. tâches cognitives de bas niveau
O D. a et b

2.Lequel parmi les termes suivants est interchangeable avec le terme «réseau neuronal»?

O A. système connexionniste
O B. traitement parallèle
O C. naturellement intelligent
O D. toutes les réponses sont bonnes

3.Quel énoncé caractérise les réseaux de neurones?

O A. ils réagissent en parallèle aux stimulis auxquels ils sont soumis
O B. ils exécutent une série d'instructions
O C. ils justifient leurs solutions à un problème
O D. ils font des déductions à partir d'une base de données

4.Laquelle parmi les suivants est l'une des plus grandes contraintes des réseau neurones?

O A. le filtrage des données demande une intervention humaine
O B. ils nécessitent un nombre impressionnant d'exemples pour apprendre et généraliser
O C. la plupart des modèles de réseaux de neurones peuvent être simulés sur des ordinateurs standards
O D. toutes les réponses sont bonnes

5.Quelle caractéristique, en comparaison avec les systèmes experts, rend les réseaux de neurones intelligents?

O A. l'expert humain n'est plus nécessaire
O B. leur capacité à apprendre et à généraliser
O C. ils peuvent déterminer eux-mêmes quelles parties des données sont significatives
O D. leur capacité à fournir une réponse précise dans un laps de temps réduit

Post R= 40%
M= 60%

6.À quoi se réfère «la somme pondéréé des potentiels d'entrée»? R

O A. un signal d'entrée multiplié par la valeur du poids associé au lien d'interconnexion
O B. la seuil de réponse
O C. l'algorithme d'apprentissage
O D. aucune de ces réponses

7.Quels sont les caractéristiques qui distinguent les problèmes bien adaptés à solution par réseaux de neurones

O A. il existe des solutions technologiques courantes
O B. le problème nécessite une grande rapidité de traitement (i.e. en temps réel)
O C. le problème peut évoluer
O D. b et c

8.Ce qui fait la puissance d'un réseau de neurones est que: M

O A. il traite l'information qu'il reçoit de façon parallèle
O B. l'abondance des liens d'interconnexion permet de créer des redondances
O C. chacun des neurones agit indépendemment des autres
O D. toutes ces réponses sont bonnes

9.Les réseaux de neurones représentent une solution idéale pour résoudre des problèmes:

O A. comportant une large partie d'appariement de modèles
O B. pour lesquels on dispose d'une quantité considérable d'exemples
O C. pour lesquels un expert humain est disponible
O D. a et b

10.Lequel parmi les suivants représente un exemple d'application de réseaux d neurones?

O A. la reconnaissance vocale
O B. l'optimisation des routes de camionnage
O C. une aide à la tâche pour les mécaniciens
O D. toutes ces réponses sont bonnes

Donner des réponses courtes et précises pour les cinq prochaines questions

11. Pour quels genres de tâches les réseaux de neurones sont-ils reconnus pour obtenir de meilleurs résultats que les systèmes informatiques conventionnels

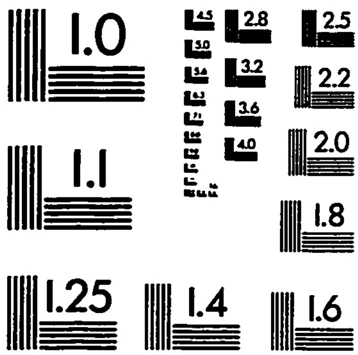
12. Comment définissez-vous le concept «système expert» (ou système à base de connaissances)?

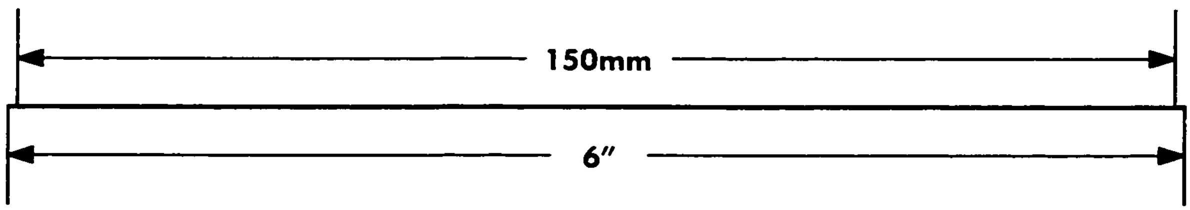
13. Quels sont les avantages à combiner les réseaux de neurones avec d'autres technologies?

14. Quels sont les trois rôles que peut prendre la sortie d'un neurone?

15. Quels sont les principaux domaines d'application des réseaux de neurones?

TEST TARGET (QA-3)





APPLIED IMAGE . Inc
1653 East Main Street
Rochester, NY 14609 USA
Phone: 716/482-0300
Fax: 716/288-5989