Evaluation Methodologies for Intelligent Tutoring Systems

MARY A. MARK AND JIM E. GREER

ARIES Laboratory
Department of Computational Science
University of Saskatchewan
Saskatoon, CANADA
S7N 0W0

Phone: (306) 966-8655
Fax: (306) 966-4884
Email: greer@cs.usask.ca

Evaluation Methodologies for Intelligent Tutoring Systems

MARY A. MARK AND JIM E. GREER

ARIES Laboratory

University of Saskatchewan, Saskatoon, Saskatchewan, CANADA

ABSTRACT

As intelligent tutoring system (ITS) issues are investigated and intelligent tutoring systems are developed, evaluation methodology becomes important. Basic researchers, system developers, and educators working with ITS all have motives for becoming involved in ITS evaluation. In formative evaluation, researchers examine a system under development, to identify problems and guide modifications. By contrast, summative evaluation is carried out to support formal claims about the construction, behaviour of, or outcomes associated with a completed system. Different methodologies are suitable for different types of evaluation, some focusing on internal considerations, such as architecture and behaviour, others on external considerations, such as educational impact.

This paper draws upon the areas of intelligent tutoring systems research, expert systems design, computer-based instruction, education, and psychology to identify techniques for the formative and summative evaluation of ITS. Evaluation techniques are discussed in terms of the motivations for using them, situations for which they are particularly suitable, and their strengths and weaknesses. Techniques are illustrated with reference to actual studies whenever possible.

1. Introduction

Until recently, little attention has been paid to evaluation of intelligent tutoring systems (ITSs). Most ITS researchers have concerned themselves with envisioning the potential of ITSs and investigating the implementation issues involved in constructing actual components and systems (Sleeman & Brown, 1982; Wenger, 1987). A substantial research base now exists in areas such as knowledge representation, search, and planning, contributing to the design and implementation of ITSs. As more ITSs are developed and introduced into business, government, and educational settings, the question of how ITSs can be evaluated is becoming increasingly important. This paper reviews evaluation techniques from different disciplines and discusses their applicability to the evaluation of ITSs.

According to Cooley and Lohnes (1976, p. 3), "An evaluation is a process by which relevant data are collected and transformed into information for decision making". Frye, Littman and Soloway (1988) have applied Scriven's (1967) distinction between formative and summative evaluations of instructional materials, according to purpose, to intelligent tutoring systems. *Formative* evaluation occurs during design and early development of a project and is oriented to the immediate needs of developers who are concerned with improving the design and behaviour of a system. It frequently addresses the ITS evaluation question "What is the relationship between the architecture of an ITS and its behaviour?" (Littman & Soloway, 1988). By contrast, *summative* evaluation is concerned with the evaluation of completed systems and the making of formal claims about those systems. Summative evaluations tend to address another important question for ITS evaluation: "What is the educational impact of an ITS on students?" (Littman & Soloway, 1988).

Formative evaluation is often seen as part of a computer programming methodology, which characterizes development in terms of cycles of design, implementation, and formative

evaluation (McGraw & Harbison-Briggs, 1989). Formative evaluation is used to obtain detailed information that can be used to modify and improve the functioning of an ITS. Primary questions when designing formative evaluation are "What are the major types of data to obtain and the major sources of such data?" and "What is the impact of evaluation data upon the modification of programs?" The first question relates to the identification of significant concerns, ranging from details of interface design to overall system goals. The second question relates to the specification and use of information related to those concerns. Instructional designers recommend that formative evaluation should begin early in development, before substantial investments in time and resources occur, and continue throughout development of a system (Gagné, Briggs & Wager, 1988). Because these results are used as a basis for further development, not for claims about the system, informal or ad hoc methods are generally acceptable for formative evaluations.

By contrast, summative evaluation attempts to prove or disprove some formal claim about a system or the techniques used in a system. Generally, claims relate to system goals. Specific questions may be "What does a particular implemented ITS do? " "Does an ITS fulfill the purpose for which it was designed?" "Does an ITS result in predicted outcomes?" "What is the effect of one type of system or component relative to a comparable system or component?" Because they deal with formal claims, summative evaluations are expected to meet more rigorous methodological standards than formative evaluations (Rosenberg, 1987).

There are few agreed upon standards within the ITS community to guide investigators who wish to evaluate systems. However, other fields have developed evaluation techniques which may be applicable to ITSs. The closest computational analog to an ITS is an expert system, while the closest educational analog to an ITS is a computer-assisted instructional (CAI) system. ITS researchers may also find it useful to draw upon the fields of education and psychology. This paper briefly reviews techniques from ITS research, expert systems design, computer-based instruction, education, and psychology in an attempt to assess their applicability to formative and

summative evaluation of ITSs. Section 2 briefly introduces some evaluation techniques, noting their suitability for formative and summative evaluation of ITSs. Section 3 relates evaluation techniques to generic ITS architecture and evaluation of ITS components. Section 4 focusses on the evaluation of ITSs' educational impact. Techniques are illustrated with reference to actual studies whenever possible.

2. Evaluation Techniques

Several principles of expert system evaluation (Gaschnig, Klahr, Pople, Shortliffe & Terry, 1983) are relevant to ITS evaluation. It has been suggested that complex objects or processes cannot be meaningfully evaluated in terms of a single measure and that a large number of distinct criteria should be examined to evaluate a complex system. Complex systems, like ITSs, can be considered in terms of a complete and total system, in terms of system components, or in terms of specific features. Techniques which are suitable for evaluating complete ITSs may not be well suited to evaluation of particular ITS components or features (and vice versa). Also, researchers will find different evaluation criteria useful, depending on their interests and concerns. Different evaluation methods will be suitable for different purposes, at different times, with different components or with overall systems. For example, Vasandani and Govindaraj (1991) report that evaluation of the intelligent tutor Turbinia-Vyasa has included expert review, pilot testing, and an experimental study. It should be kept in mind that the development of an evaluation is a complex process. Attempting to identify and address all the issues and concerns that are relevant to the design of ITS evaluation is beyond the scope of this paper, and is not attempted.

2.1 Proofs of correctness

Program architecture and behaviour are often approached from the viewpoints of verification and validation (McGraw & Harbison-Briggs, 1989). In program verification, the

system is evaluated in terms of the correspondence between its structure and behaviour and its specifications. Does it do what it is said to do? In program validation, the system is evaluated on whether or not its behaviour fulfills desired requirements or goals. Does it do what it should? Conventional computer programs are sometimes verified and validated through formal proofs of correctness. However, this technique is unsuitable for AI programs which deal with analytically intractable problems, represented as incompletely specified functions (Partridge, 1986). This category includes ITSs. Proofs of correctness are therefore inapplicable to ITSs.

2.2 Criterion-based evaluation

In another approach to evaluation, a system is considered successful or "adequate" if it displays no major inadequacies within its intended application environment (Partridge, 1986). The system evaluator is left to determine what "major inadequacies" are, based on the degree to which the system is subjectively felt to meet its requirements and specifications. The care with which these criteria have been developed is critical. McGraw and Harbison-Briggs (1989) argue that it is extremely difficult to develop specific objectively-measurable criteria for knowledge-based systems, implying that this type of evaluation has limited usefulness for ITS. It is best suited to formative development, particularly early development where developers are concerned with general characteristics rather than precise detail or for evaluation of specific aspects of a system, like interface design, where criteria can be specified and measured precisely.

General guidelines for judging program construction, behaviour, and characteristics are sometimes recommended. Popular for CAI (cf. Hativa, 1988), this approach also has been used with ITS (Ford, 1988). Guidelines can raise interesting and challenging concerns about the design and use of learning environments (Bork, 1988). However, the use of general checklists and review guidelines in assessment has been seriously criticized. Reviewers' ratings have not

correlated well with each other or with actual student achievement. (Jolicoeur & Berger, 1986, 1988; McGraw & Harbison-Briggs, 1989).

Well-validated research in cognition, perception, and learning (Jay, 1983; Jonassen & Hannum, 1987; Larsen, 1985) can suggest ways to design and improve educational programs, particularly interface and user-related features. However, a basis in theoretical principles does not guarantee desired results. Such guidelines may be useful in directing development of ITSs, but the effectiveness of a program or its features should be verified in other ways.

2.3 Expert knowledge and behaviour

Often, an expert's knowledge is used as an explicit standard for judging a program (Gaschnig et al., 1983). In education and with CAI systems, it is common to have an expert examine all relevant aspects of surface behaviour during formative evaluation (Alessi & Trollip, 1985; Steinberg, 1984). Expert inspection is possible when behaviour is consistent and predictable. It has limited applicability for ITS evaluation because ITS behaviour is complex and dynamic, and underlying representations may not be inspectable or intuitively understandable. Inspection is sometimes applied to ITS components, where it can be quite useful for formative evaluation (Hofmeister, 1986).

The behaviour of human experts can also be used to evaluate knowledge-based systems. The Turing test is a well-known technique for comparing human and computer behaviour (cf. Parry & Hofmeister, 1986). In a Turing test, a system is considered to be successful if its behaviour is indistinguishable from or superior to the behaviour of human experts. This technique is desirable if (1) humans provide a good standard of comparison for the desired behaviours (Gaschnig et al., 1983) and (2) either there are explicit objective criteria for measuring and comparing behaviours of the human and the system or humans are considered to be competent

judges of the behaviours examined. Since teaching is not well understood and human and machine tutors differ in the resources they offer for interaction with students, measures for comparing the overall behaviour of ITSs and humans are difficult to find. Furthermore, humans are not necessarily good judges of what constitutes good teaching. Therefore, a Turing test is unlikely to offer the conclusive proof of a system's overall merits or failings desirable for summative evaluation. However, a Turing test may be useful in examining the behaviours of specific ITS components, as part of formative or summative evaluation.

2.4 Certification

Another possibility would be to base techniques for identifying competent teaching systems on those for identifying competent human teachers. Perhaps ITSs could be appraised by independent human teachers, given feedback on their strengths and weaknesses (formative evaluation), and rated as to their adequacy (summative evaluation). Such certification would be "an authoritative endorsement of the correctness of a program" (McGraw & Harbison-Briggs, 1989, p. 303). However, this recalls questions about the standards which would be appropriate for judging programs, the criteria which could be used for evaluating systems and components, and the accuracy with which humans can in fact identify effective educational programs. At present, these questions have not been clearly answered. If these concerns can be resolved, certification may provide a desirable approach for ITS evaluation.

2.5 Sensitivity Analysis

With sensitivity analysis (Gaschnig et al, 1983), a component or system could be examined to see how responsive its behaviour is to differences in the information given to it. This could be particularly relevant when evaluating ITSs, which are supposed to offer individualized instruction. The sensitivity of an ITS to different learner characteristics might indicate whether

additional teaching expertise needs to be incorporated into the system (formative). A system which displayed similar responses to significantly different input might be considered less desirable than one displaying greater variation in response (summative). Whether measures of the sensitivity of an ITS would really be appropriate for formative or summative evaluation is likely to depend on whether measures of sensitivity could be assigned meaning (what range of behaviors are desirable, and when should they be invoked, given a system's goals), on how precisely measures of sensitivity could be determined, and on whether they could be compared.

2.6 Pilot Testing

If developers can assume that a system's prospective users will have levels of experience and expertise similar to their own, they can design a system to meet their own needs and feel reasonably sure that the needs of prospective users will also be met. However, users may differ in knowledge of tutors and tutoring domains and in basic cognitive abilities. To identify users' problems and concerns, investigation of actual system use by members of the intended user population is often recommended as part of formative evaluation of systems under development (Hofmeister, 1986; Patterson & Bloch, 1987). To determine whether systems are used as anticipated, and to ensure that formal claims cannot be called into question due to unexpected outcomes, it may also be desirable for pilot testing to be carried out in cooperation with summative testing of completed systems.

Gagné et al. (1988) and Golas (1983) identify three types of pilot testing: one-to-one testing, small-group testing, and field testing. In *one-to-one testing*, observers make detailed observations of how a student interacts with the instructional materials being developed. Investigators can observe student capabilities; identify inappropriate expectations; detect unclear directions, questions, and information; and note unexpected features of the instructional situation (Gagné et al, 1988). One-to-one pilot tests are usually carried out early in development to

minimize inappropriate development. Richer and Clancey (1987) use one-to-one evaluation to identify problems in their knowledge-base interface Guidon-Watch. Similarly, Fischer, Lemke, McCall, and Morch (in press) and Girgensohn (1992) propose modifications to JANUS and MODIFIER based on observation of individual users. *Small group testing* is usually carried out later in the development process, once the format of the program and its content have begun to stabilize. A small group of students, representative of the target population, are questioned before and after system use to assess their understanding of the content taught. Such information indicates whether specific aspects of content or program use are learned or understood by students. *Field testing* examines system use in actual instructional settings with real instructors and students. This type of evaluation attempts to identify problems which occur when a program is introduced into the type of environment in which it will be used and to determine whether students will exhibit anticipated behaviour and learning outcomes when a program is used under near-typical conditions (Hoecker & Elias, 1986). Field testing also enables researchers to expand the focus of an evaluation and gather information about possible unanticipated outcomes (Schofield, Evans-Rhodes & Huber, 1990). Sensitivity to the possibility of unexpected effects, positive or negative, can prevent misleading conclusions from being reached. Sensitivity to the unexpected can also result in important discoveries about teaching and learning.

2.7 Experimental Research

Common in psychology and education, experimental research is suited to educational systems, including ITSs, because it enables researchers to examine relationships between teaching interventions and student-related teaching outcomes, and to obtain quantitative measures of the significance of such relationships. The formal power of experimental techniques is rarely needed in formative research, but may be desired if designers wish to quantitatively examine or compare the effects of specific system features. Experimental techniques are often used for summative

research, where formal power is desired and where overall conclusions, rather than acquisition of information, are desired.

An evaluation answers the questions for which it was designed, so the first step in research design is identification of a research question to be examined. Waugh and Currier (1986) outline concerns which they see as relevant research questions for computer-based education, suggesting that research is needed to the examine interrelationships between computer programs, instructional methods, and student characteristics and activities involved in learning. Given a research question, hypotheses can be formed. An hypothesis must be testable, concerned with specific conditions and outcomes, and possible to confirm or deny on the basis of those conditions and outcomes. A research design is then developed to enable the researcher to examine the hypothesis. When a practical, suitable design has been found to answer the research question, the researcher can carry out the study and data from the study can be analyzed. Ideally, if results do not confirm the research hypothesis, researchers should be able to suggest possible explanations for their results. For example, Gallagher (1981) gives an excellent analysis of several features of a blocks world tutor which could have interfered with student learning.

A variety of different experimental designs exist, including single group designs, control group designs, and quasi-experimental designs. Each design has its own strengths and weaknesses. Well-designed CAI experiments include Emihovich and Miller (1988), Johnson, Gersten and Carnine (1987), and Rowland and Stuessy (1988). Recent studies also apply experimental methodology to ITS evaluation. Lajoie and Lesgold (1991) have carried out a single-factor control group experiment to examine the effectiveness of the ITS system, SHERLOCK. While pre-training performance did not differ, post-tests of on-the-job avionics performance showed that Air Force trainees who worked with SHERLOCK performed better than trainees who worked in the avionics shop. Control group studies have also compared variants of specific tutoring systems, to determine the effects of particular factors or aspects of the tutor. Vasandani

and Govendaraj (1991) have compared simulator use, passive tutoring, and active tutoring in teaching troubleshooting of a marine power plant. They conclude that the simulator alone was inadequate, while the two training versions helped students to develop troubleshooting skills. Mark and Greer (1991) have compared four versions of a VCR Tutor, to determine the effectiveness of different tutorial approaches utilizing different types of knowledge. They found that subjects using a knowledgeable tutor learned to program a VCR simulation using fewer steps and with fewer errors and types of errors than subjects who used a prompting version of the tutor. Versions of the Lisp Tutor have been used to experimentally examine the effects of a variety of factors. Corbett, Anderson, and Fincham (1991) report that menu-driven data entry inhibits learning, compared to type-in code entry, even though students in the menu condition tend to like their tutor more. Corbett, Anderson, and Patterson (1990) compared immediate vs. student-controlled presentation of feedback. They found that students who received immediate feedback completed their exercises more quickly than students who controlled the presentation of feedback. However, students who controlled the presentation of feedback seemed to be able to identify and correct their own errors. Similarly, in examining immediate vs. delayed feedback Schooler and Anderson (1990) report that delayed feedback seemed to promote the acquisition of desirable skills such as error detection and self-correction. While these studies tend to examine the effects of single factors, more complex research designs are also being used to examine multiple factors and their interactions. Shute has examined relationships between computer learning environments and learner characteristics, using Smithtown (Shute & Glaser, 1990).

3. Evaluating Architecture and Behaviour

Since ITSs are complex systems, different components of an ITS may require different evaluation approaches (Gaschnig et al, 1983). It is important to consider the suitability of evaluation techniques for components of ITS architecture. The general architecture discussed here contains six common ITS components (McCalla & Greer, 1990). The *domain knowledge*

component of an ITS is concerned with storing, manipulating, and reasoning with knowledge of some subject domain. The *teaching* component of an ITS uses knowledge of how to teach. The *communication* component presents information to the student and acquires responses from the student. A *student knowledge* component describes the system's understanding of the student's knowledge and needs through diagnosis, i.e. understanding of the student's behaviour, and through modelling, i.e. maintaining a coherent view of student-related knowledge. An ITS may also be able to monitor and adjust its own behaviour with a *learning* component. Overall operation of an ITS is managed by a *control* component.

3.1 Domain Knowledge

A task force for the National Science Teachers Association (NSTA) (Klopfer, 1986) has identified subject matter standards as one of the major considerations for educational assessment of of computer-related instructional materials. In an ITS, information about a subject or domain may be presented to the student through textual descriptions, examples, problems to solved, questions to be answered, active or interactive graphic displays, and feedback responding to the student's actions. The underlying representation and manipulation of domain knowledge is controlled by the domain knowledge component. Ideally, accuracy of this component should be ensured before a system is completed and assumed thereafter. Formative evaluation is of primary importance for ITS developers: summative claims about accuracy are only interesting if the domain is inaccurate!

Possible techniques for formative evaluation of domain knowledge are expert inspection and the Turing test. Concerns about their applicability to overall system evaluation may also apply to domain knowledge evaluation, depending on the domain, the way it is represented, and the purposes for which domain knowledge is used. It is easier to verify facts and examine reasoning mechanisms in less complex, well understood domains. Similarly, if the domain knowledge base

can be assessed in terms of breadth or depth or if it has clearly defined standards of coverage, then it may be possible to determine whether or not those standards are met by the representation. If a domain is not well understood, or it is difficult to develop test cases or other measures to determine domain accuracy, or the underlying representation of knowledge is difficult to inspect, expert inspection may not be desirable. If humans are considered to provide a good standard of comparison for expertise in the area, then a Turing test may be suitable.

3.2 Teaching Knowledge Component

A second major area of concern for the NSTA task force was the instructional quality of computerized instructional programs (Klopfer, 1986) Ideally, the teaching component of an ITS should be an expert which designs and guides a tutorial session in accordance with instructional methods and conditions for their use. A goal of ITS developers is to build systems which will create individualized instruction, accommodating student characteristics such as background knowledge, level of cognitive development, and learning style (Ackerman, Sternberg & Glaser, 1989; Jones, 1988; Yang, 1987) to more effectively promote student learning.

The standards to which teaching knowledge can be compared are instructional theory and the expert human teacher. Teaching knowledge is not necessarily well understood or explicitly described, making it difficult to evaluate. Formative evaluation techniques such as observation of CAI surface behaviour (Alessi & Trollip, 1985) and use of instructional checklists (Jonassen & Hannum, 1987) apply poorly to evaluation of ITS teaching knowledge. If standards for assessing the significance of an ITS's teaching knowledge were developed, they might reflect the NSTA criteria (Klopfer, 1986), which include specific considerations such as the range of instructional methods offered by a program, the degree to which a program can adapt its behaviour to individual differences of students, and more general concerns such as the degree to which instruction is based upon educational and psychological research in teaching. How such criteria could be assessed,

formatively or summatively, is as yet unknown. Sensitivity analysis and certification might be possible approaches for investigation. Turing tests and experimental techniques are difficult to apply to an isolated tutoring component. Behaviour of a teaching knowledge component depends upon student and domain knowledge. It may be difficult to produce realistic teaching knowledge component behaviour free of sources of confounding from other components. One possibility is to experimentally compare teaching knowledge components of an ITS, while keeping other components identical (cf. Mark & Greer, 1991).

3.3 Student Knowledge

The student knowledge component of an ITS can carry out both diagnosis and modelling (Ohlsson, 1986; Sleeman & Brown, 1982; Wenger, 1987). Diagnosis examines a student's behaviour to obtain meaningful information from it. Modelling relates past and present information about the student in meaningful ways, and maintains that information for use by the system. The standard to which an ITS's student knowledge is compared is the actual student, whose knowledge may be be changeable, contradictory, and difficult to observe.

A student knowledge component is analogous to an educational or psychological test instrument in that it attempts to measure student characteristics. Conventional test instruments are judged on the basis of four main characteristics: validity, reliability, objectivity, and standardization (Mayer, 1987). A test is valid if evidence shows that it measures what it purports to measure. It is reliable if its results for a particular subject are consistent. It is objective if it is administered and scored in the same way for each person. It is referenced or standardized if test results can be translated into some meaningful description of student performance. Normative referencing describes a student's performance in relation to the performance of other students. Domain referencing (also called criterion referencing) describes a student's performance in terms of precisely-defined standards of performance for a particular domain. While the goals of a student

knowledge component are similar to the goals of a test instrument, there are important differences in their usage and goals which limit the extent to which standards for evaluating test instruments apply to student knowledge components.

The characteristics of test instruments most relevant to evaluation of a diagnostic component are validity and reliability. A diagnostic component, like a test instrument, is valid if it measures what it purports to measure at some time. A diagnostic component is reliable if it yields consistent results from comparable evaluations of a student. Objectivity and standardization are less applicable to ITSs. Objectivity, in the sense of identical administration, conflicts with ITSs' key goal of presentation of an individualized course of instruction to the student. Also, since the concern of the ITS is a particular student, normative referencing applies poorly to ITSs. Some form of domain referencing is often implicit in the construction and operation of the diagnostic component, since the component is concerned with interpreting student behaviour in meaningful ways using domain knowledge. However, this does not mean that a diagnostic component will produce an analysis of the student consistent with a domain-referenced standard. The types of information relevant to individualized instruction may differ from the types of information relevant to domain referencing.

The concerns of validity, reliability, objectivity, and standardization can also be related to student modelling. A student model is valid to the extent that it accurately reflects the student over time. In this, the student model differs from most test instruments, which do not deal with changes over time except in the sense that they may be used as discrete measures of the student at intervals over a period of time. Since the student model is based on the idea of monitoring changes in the student's behaviour and understanding over time, the criterion of reliability applies poorly to student modelling. Objectivity is not suited to the evaluation of student modelling for the same reasons that it is unsuitable to the evaluation of diagnosis. Since the important question for a student modelling component is the individualizing of instruction, standardization, in the sense of a

application of a set of standards for interpreting performance, also applies poorly to student modelling.

Validation is an important consideration during both formative and summative evaluation. Developers are more likely to be concerned about reliability during summative evaluation, after validity has been established. Validity and reliability are most effectively assessed through experimental studies. In determining diagnostic validity, diagnostic information obtained by the system is compared to diagnostic information obtained independently. Similarly, a student model can be validated by seeing whether repeated measures from the student model consistently agree with independent measures over time. To assess reliability, information from comparable sessions can be examined to determine whether consistent results are being produced by the system.

3.4 Communications Component

The third major area of concern identified by the NSTA is technical quality, referring to the facilities available to the human in understanding and operating the computer and the quality of the interface offered to the human by the computer program (Klopfer, 1986). Regardless of the quality of its underlying design and knowledge engineering, an ITS will be of little use if students cannot understand the information that it presents, or if they misinterpret directions for responding, or make errors because of an awkward interface. The interface features most commonly presented in instructional computer systems at present are graphics and prepared text, while the most common means of entering information are mouse-driven menus and graphics and the entry of domain-specific notation (numbers, formulae, limited text, etc) via the keyboard. Guidelines or reviews may be of use in directing early development. However, pilot testing with members of the population for whom the system is designed is more likely to identify problems and concerns for the formative evaluation of an interface (England, 1985; Frye et al, 1988).

Another approach suitable for formative or summative evaluation of interfaces is to compare versions of an interface experimentally to see which is more effective (Corbett, Anderson & Fincham, 1991)

3.5 Learning Component

The development of ITSs which could themselves learn would have significant implications for evaluation. If an ITS could monitor its actions as a teacher would, using information about its own actions and the behaviour of students to alter its teaching, what significance could be attached to evaluations of the ITS? Evaluations of components which were directly or indirectly affected by system learning, and of the system as a whole, could not be taken as indicative of a system's capabilities at any time other than when obtained. It would become essential to assess changes in system behaviour over time, to determine whether or not the learning component actually improved system behavior, not only in terms of the criteria which the learning component used as a basis for learning, but also in terms of other criteria relevant to other system components (O'Shea, 1982).

3.6 System Control

In an ITS, as in any complex system, interactions between components must be mediated. A control component has important implications for the behaviour of the system, both in meaningful interaction and speed. Appropriate evaluation techniques will depend in part upon the implementation of the control component. Since the control component is concerned with underlying features of system behaviour such as the scheduling of processes for different components, conventional computer performance evaluation techniques could be appropriate for this component. Such analysis is most likely to be of interest to formative evaluators concerned

with improving system performance. Summative evaluators are more likely to be concerned with the educational aspects of an ITS than with strict computational performance.

Table 1 summarizes the evaluation methods discussed in Section 2 as these apply to ITS architecture. Each method is described in terms of the degree to which it is recommended for ITS evaluation and whether it is preferable for formative or summative evaluation. Some indication also is given of the architectural aspects of an ITS to which each evaluation method is particularly likely to be suited.

insert Table 1 about here

4. Evaluating Educational Impact

Littman and Soloway's (1988) second evaluation question focuses on the educational impact of an ITS on students, recognizing that since the major goal of an instructional system is to teach, its major test is whether students learn effectively from it (Hasselbring, 1986; Waugh & Currier, 1986). Criteria used to judge the educational effects of ITSs include both achievement and affect. Achievement and achievement-related measures are concerned with the acquisition, understanding, performance, retention, and transfer of a learner's knowledge and skills (Haertel & Calfee, 1983). Affective measures are concerned with attitudes and emotions, which may impact upon students' use of and learning from ITSs (Malone, 1981). Educational effects are sometimes examined during formative evaluation of ITS and are a major concern in summative evaluation when evidence is desired for formal claims about whether or not an ITS achieves its teaching goals. Formative evaluation generally obtains qualitative or quantitative assessments of educational effects through pilot testing, while summative claims are based on experimental research.

4.1 Achievement Measures

The nature of knowledge and the nature of learning have been examined by philosophers, psychologists, educators, and cognitive scientists. Resulting characterizations of knowledge provide guidelines for the description and assessment of achievement. Knowledge can be categorized into types, based upon the degree of conscious awareness of knowledge and the purposes for which knowledge appears to be used. Knowledge of symbols and meanings can be called *referential* knowledge, verbal information (Gagné, 1974), or semantic knowledge (Mayer, 1987; McGraw & Harbison-Briggs, 1989). *Factual* knowledge is specifiable knowledge about objects and relationships between objects within the world. Mayer includes factual knowledge within his category of semantic knowledge, while McGraw and Harbison-Briggs classify it as part of a larger category of declarative knowledge, knowledge which can be readily described. *Procedural* knowledge of how to do things can be categorized as explicit or implicit. *Explicit* procedural knowledge can be consciously described in terms of an algorithm, rules, or procedures, and used to guide performance (Mayer, 1987). This has also been referred to as intellectual skill (Gagné, 1974) and declarative knowledge (McGraw & Harbison-Briggs, 1989). *Implicit* procedural knowledge is knowledge of how to do something that cannot be easily described verbally. Mayer refers to this as skills. McGraw & Harbison-Briggs reserve the term procedural knowledge to indicate implicit procedural knowledge. *Metacognitive* knowledge (Borkowski & Cavanaugh, 1979), also referred to as metaknowledge (McGraw & Harbison-Briggs, 1989), strategic knowledge (Mayer, 1987), or learning strategies (O'Neil, 1978), is increasingly believed to be important in the conscious monitoring of human information processing behaviour (Baron & Sternberg, 1987). What it means to learn something and how learning can be measured can be considered in terms of these different types of knowledge.

Learning objectives can also be considered in assessing educational effects. Bloom (Bloom, 1956; Bloom, Hastings, & Madaus, 1971) has elaborated a taxonomy of cognitive educational objectives, general categories of learning outcomes, which can be related to observable

performance indicators. He identifies six major types of objectives, which he calls knowledge, comprehension, application, analysis, synthesis, and evaluation. At Bloom's knowledge level, a student is able to recall a term, fact, or procedure, but not to understand or apply it. (Since Bloom's usage of the term "knowledge" is significantly restricted compared to its usage elsewhere in this paper, Bloom's "knowledge" level will be referred to here as *recall* level learning.) *Comprehension* suggests that a student can use material to some degree, enough to give definitions and draw direct conclusions. At the *application* level, a student begins to make use of knowledge in concrete situations, identifying conditions which make available knowledge relevant. *Analysis* implies that the student not only can identify underlying ideas, but also can examine and discuss their relationships. *Synthesis* implies that the student can also organize presented materials to generate new ideas. Finally, *evaluation* involves the ability to judge the value of knowledge.

Mark (1990) uses the categories of type of knowledge (Mayer, 1987) and level of learning (Bloom, 1956) as a framework for identifying possible measures for the assessment of achievement. Referential, factual, procedural, and metacognitive knowledge are terms designating categories of knowledge. Recall, comprehension, application, analysis, synthesis, and evaluation are terms designating knowledge outcomes or levels of understanding which people display. Together, these categories of knowledge and learning outcomes constitute a taxonomy of the types of behaviours which demonstrate whether or not a student has learned what is being taught. Such a categorization may be helpful in determining what measures to use to assess achievement. Ideally achievement measures should be valid and reliable and yield objective, measurable results.

4.2 Achievement-related Measures

Other criteria related to achievement which educators have found useful in assessing educational impact include transfer, retention, learning time, and completion rates. Transfer,

retention, and learning time can be assessed experimentally, while completion rates are sometimes observed during field trials.

Transfer is the ability to apply information or skills learned in one context in a new and often unfamiliar context in which that knowledge is also relevant (Mayer, 1987; Perkins & Salomon, 1987; Salomon, 1984). Transfer indicates abstract understanding of material, and the ability to identify contextually relevant features of new situations. Transfer is a concern of evaluators when the ability to generalize knowledge is important, as in metacognition and the learning of procedural skills which must be applied to many different situations. It is also important when students must apply skills learned in a simulated environment in a real environment. Transfer is generally assessed by testing a student in the learning situation and in a situation to which the learned material is believed to be transferable. (Johnson et al,1987; Collins, Carnine and Gersten , 1987; Swan, 1989).

Retention is the ability to maintain learning over time (Mayer, 1987). The longer a student is able to recall and use learned information and skills, the better the student is considered to have learned that knowledge. Retention is assessed by taking comparable measures of the same phenomenon over time. For example, two parts of a mathematics test, which are known to be comparable, might be administered at different times (perhaps spring and fall) to determine whether knowledge has changed over time, independent of instruction. When learning has been demonstrated, educators may wish to assess retention of that learning as additional evidence for the benefits of their instructional approach.

Educators may also be concerned with the *time* that it takes to learn (Schmalhofer, Kuhn, Messamer & Charron, 1990). As a criterion for evaluating systems, time's value varies. Comparisons of learning times of students using different systems are sometimes used to argue that one system is superior to another. However, learning time should be considered in the context

of concerns such as achievement and motivation. One situation where learning time is an appropriate criterion for evaluation is mastery learning, in which a student must demonstrate predetermined levels of understanding of a topic or skill before learning more material (Hambleton, 1974).

A final concern for evaluators is the drop-out or *completion rate.* This can be observed during field tests in which students have the option of continuing or discontinuing their use of an instructional program (cf. Hoecker & Elias, 1986). Completion rates can be important when considering educational effects of systems. For example, an ITS which is difficult to use or which teaches at an inappropriate level may discourage students, leading them to drop out. This may make it a poor choice for classroom use, even if those students who do use it achieve well, because the population of students as a whole do not find it easy or desirable to use.

4.3 Affective Measures

When attitudes and feelings mediate learning, they are relevant to the effectiveness of ITS. One argument frequently presented in favour of computer use in education is that computers are intrinsically motivating. This term is used to cover a variety of factors which promote and inhibit individual involvement in particular behaviours (Ellis & Sabornie, 1986; Mayer, 1987; Smith & Keep, 1986). Most generally, it indicates people's willingness to be active and involved. Educators may be concerned with *academic motivation*, interest in engaging in academic activities; *achievement motivation*, interest in attaining a standard of excellence or some desired end; or *attitudes* towards education, computers, or specific programs (Thomas, 1979). The most common method of assessing motivation is to ask students to rate their agreement with specific attitudes, beliefs, and activities (Moore, 1985). Comparisons of time spent on task-related and unrelated activities during a session are another indicator of interest. Overall time is sometimes used as an indicator of motivation when an activity is voluntary, as is the drop-out or completion

rate for the program. Researchers have also suggested that computer use may foster *self-esteem* (Emihovich & Miller, 1988), which in turn may encourage educational achievement. Self-esteem can be assessed using specially constructed measures of self-esteem which are available from commercial and academic sources.

However, while affective measures may indicate how students feel about a system (Jacobson & Smith, 1990/91), they don't necessarily reflect achievement or achievement-related criteria accurately (Corbett & Anderson, 1990). In assessing the affective impact of an ITS, or in basing decisions upon it, it is important to consider whether there can be more than one interpretation of affective evidence. Frequently, the significance of affective evidence is disputable and, in any case, it must be weighed against other factors in an evaluation. Affective measures may be useful in the identification of problems or concerns during formative evaluation, but they are not sufficiently dependable to ensure that existing problems will be identified. They may suggest whether or not ITSs will be accepted and used. They may also be interesting as measures of the long term impact of ITS on attitudes and education. Finally, affective measures may be desired to supplement measures of achievement in summative evaluations. However, it should be kept in mind that they are rarely convincing evidence of effectiveness if considered in isolation.

5. Conclusions

This paper has reviewed evaluation techniques from different disciplines in an attempt to identify possible methods for formative and summative evaluation of ITSs. Special characteristics and goals of ITSs must be considered when critiquing evaluation methods from other disciplines. Because of the complexity of ITSs, evaluation techniques for traditional computer programs and simpler CAI systems are frequently inappropriate for ITS. Lack of precise knowledge about teaching makes specification and evaluation of ITSs difficult.

Some informal methods seem to be applicable to ITSs, although perhaps not ideal. Detailed information about ITS architecture and behaviour can be gathered during formative evaluation and used to guide system modification. For some aspects of architecture and behaviour, criterion-based or expert-based assessment may be appropriate, while for others pilot testing is preferred.

Unfortunately such informal techniques rarely have the rigour and the power necessary for summative evaluation. Formal methodological techniques, such as experimental designs, can be used to examine summative claims about ITSs. Evaluation techniques which focus on the educational effects of ITS on students are appropriate for summative evaluation of ITS since the major goal of an ITS is to teach. Another goal of ITS research is to develop systems that can provide personalized instruction, adapting to the educational needs of a range of students. While neither approach has been deeply examined in this paper, sensitivity analysis and certification may have potential for evaluating the adaptiveness of ITSs.

Formal techniques for summative examination of complex architecture and behaviour are not yet available. Evaluation techniques that yield detailed results rarely yield formal results; while techniques that yield formal results, usually examine some general aspect of a system or its educational effects. Current techniques do not formally examine complex systems in ways that retain and analyze that complexity. An ideal ITS evaluation technique would provide both the certainty of a formal analysis and the detail meaningful for analysis of a complex system. Whether such techniques can be developed and what they would involve, are challenging topics for future research.

References

Ackerman, P.L., Sternberg, R.J., & Glaser, R. Learning and individual differences. New York: W.H. Freeman , 1989.

Alessi, S.M. & Trollip, S. Computer-based instruction: Methods and development. Englewood Cliffs, N. J.: Prentice-Hall, 1985.

Baron, J. B. & Sternberg, R. J. Teaching thinking skills: Theory and practice. New York: W.H. Freeman, 1987.

Bloom, B.S. Taxonomy of educational objectives: Cognitive domain. New York: David Mackay, 1956.

Bloom, B.S., Hastings, J.T. & Madaus, G.F. Handbook on formative and summative evaluation of learning. New York: McGraw-Hill, 1971.

Bork, A. Ethical issues associated with the use of interactive technology in learning environments. Journal of Research on Computing in Education, 1988, 21(2), 121-128.

Borkowski, J.G. & Cavanaugh, J.C. Metacognition and intelligence theory. In M.P. Friedman, J.P. Das, & N. O'Connor (Eds.), Intelligence and learning. New York: Plenum Press, 1979.

Collins, M., Carnine, D. & Gersten, R. Elaborated corrected feedback and the acquisition of reasoning skills: A study of computer-assisted instruction. Exceptional Children, 1987, 54(3), 254-262.

Cooley, W.W. & Lohnes, P.R. Evaluation research in education. New York: Irvington Publishers, 1976.

Corbett, A.T. & Anderson, J.R. Feedback timing and student control in the Lisp intelligent tutoring system. In D. Bierman, J. Breuker & J. Sandberg (Eds.), Artificial Intelligence and Education: Proceedings of the 4th International Conference on AI and Education, 24-26 May, 1989, Amsterdam, Netherlands. Amsterdam: IOS, 1989.

Corbett, A.T. & Anderson, J.R. The effect of feedback control on learning to program with the Lisp tutor. The Twelfth Annual Conference of the Cognitive Science Society, July 25-28, 1990, Cambridge, Massachusetts. Hillsdale, N.J.: Lawrence Erlbaum Associates, 1990.

Corbett, A. T., Anderson, J. A. & Fincham, J. M. Menu selection vs. typing: Effects on learning in an intelligent programming tutor. In. L. Birnbaum (Ed.), The International Conference on the Learning Sciences, Charlottesville, VA: Association for the Advancement of Computing in Education, 1991, p. 107-112.

Corbett, A. T., Anderson, J. A. & Patterson, E. G. Student modelling and tutoring flexibility in the Lisp intelligent tutoring system. In C. Frasson & G. Gauthier (Eds.), Intelligent Tutoring Systems: At the Crossroads of Artificial Intelligence and Education. Norwood, N.J.: Ablex Publishing Corporation, 1990.

Ellis, E.S. & Sabornie, E.J. Effective instruction with microcomputers: Promises, practices, and preliminary findings. Focus on Exceptional Children, 1986, 19(4), 1-16.

Emihovich, C. & Miller, G.E. Effects of LOGO and CAI on black first graders' achievement, reflectivity, and self-esteem. The Elementary School Journal, 1988, 88(5),473-487.

England, E. Interactional analysis: The missing factor in computer-aided learning design and evaluation. Educational Technology, 1985, 25(9), 24-28.

Fischer, G., Lemke, A. C., McCall, R. & Morch, A. I. Making argumentation serve design. HCI Journal, in press.

Ford, L. The appraisal of an ICAI system. In J. Self (Ed.), Artificial intelligence and human learning: Intelligent computer-aided instruction. London: Chapman and Hall, 1988.

Frye, D., Littman, D.C., & Soloway, E. The next wave of problems in ITS: Confronting the "user issues" of interface design and system evaluation. In J. Psotka, L.D. Massey, S.A. Mutter & J.S. Brown (Eds.), Intelligent tutoring systems: Lessons learned. Hillsdale, N. J.: Lawrence Erlbaum Associates, 1988.

Gagné, R.M. Essentials of learning for instruction. Hinsdale, Illinois: Dryden Press, 1974.

Gagné, R.M., Briggs, L.J., & Wager, W.W. Principles of instructional design. New York: Holt, Rinehart and Winston, 1988.

Gallagher, J.P. The effectiveness of man-machine tutorial dialogues for teaching attribute blocks problem-solving skills with an artificial intelligence CAI system. Instructional Science, 1981, 10, 297-332.

Gaschnig, J., Klahr, P., Pople, H., Shortliffe, E. & Terry, A. Evaluation of expert systems: Issues and case studies. In F. Hayes-Roth, D.A. Waterman, & D.B. Lenat (Eds.), Building expert systems. Reading, Massachusetts: Addison-Wesley, 1983.

Girgensohn, A. MODIFIER: Making systems end-user modifiable. Manuscript submitted for publication, 1992.

Golas, K. C. The formative evaluation of computer-assisted instruction. Educational Technology, 1983, 23(1), 26-28.

Haertel, E. & Calfee, R. School achievement: Thinking about what to test. Journal of Educational Achievement, 1983, 20(2), 119-132.

Hambleton, R. K. Testing and decision-making procedures for selected individualized instructional programs. Review of Educational Research, 1974, 44(4), 371-400.

Hasselbring, T.S. Research on the effectiveness of computer-based instruction: Review. International Review of Education, 1986, 32(3), 313-324.

Hativa, N. Differential characteristics and methods of operation underlying CAI/CMI drill and practice systems. Journal of Research on Computing in Education, 1988, 20, 258-270.

Hoecker, D., & Elias, G. User evaluation of the Lisp intelligent tutoring system. In Proceedings of the Human Factors Society, 1986, 182-185.

Hofmeister, A.M. Formative evaluation in the development and validation of expert systems in education. Computational Intelligence, 1986, 2, 65-67.

Jacobson, R. & Smith, G. Expert systems: Using artificial intelligence to support students doing algebra homework. Journal of Artificial Intelligence in Education, 1990/91, 2(2), 57-65.

Jay, T.B. The cognitive approach to computer courseware design and evaluation. Educational Technology, 1983, 23(1), 22-26.

Johnson, G., Gersten, R. & Carnine, D. Effects of instructional design variables on vocabulary acquisition of LD students: A study of computer-assisted instruction. Journal of Learning Disabilities, 1987, 20(4), 206-213.

Jolicoeur, K. & Berger, D.E. Do we really know what makes educational software effective? A call for empirical research. Educational Technology, 1986, 26(12), 7-11.

Jolicoeur, K. & Berger, D.E. Implementing educational software and evaluating its academic effectiveness: Part II. Educational Technology, 1988, 28(10), 13-19.

Jonassen, D.H. & Hannum, W.H. Research-based principles for designing computer software. Educational Technology, 1987, 27, 7-14.

Jones, M. Instructional systems need instructional theory: Comments on a truism. (ARIES Research Report 88-9). Saskatoon, Saskatchewan: University of Saskatchewan, Department of Computational Science, ARIES Laboratory, 1988.

Klopfer, L.E. Intelligent tutoring systems in science education: The coming generation of computer-based instructional programs. Journal of Computers in Mathematics and Science Teaching, 1986, 5, 16-32.

Lajoie, S. P. & Lesgold, A. M. The SHERLOCK experience: An evaluation of a computer-based supported practice environment for electronics troubleshooting training. Proceedings of the International Conference for Cognitive Science for the Development of Organizations (ICO'91) Montreal, May 2-4, 1991, 1991, 56-62.

Larsen, R. E. What communication theories can teach the designer of computer-based training. Educational Technology, 1985, 25(7), 16-19.

Littman, D. & Soloway, E. Evaluating ITSs: The cognitive science perspective. In M.C. Polson & J.J. Richardson (Eds.), Foundations of intelligent tutoring systems. Hillsdale, New Jersey: Lawrence Erlbaum Associates, 1988.

Malone, T. Towards a theory of intrinsically motivating instruction. Cognitive Science, 1981, 4, 333-369.

Mark, M.A. Evaluation of intelligent tutoring systems. (ARIES Research Report 90-2). Saskatoon, Saskatchewan: University of Saskatchewan, Department of Computational Science, ARIES Laboratory, 1990.

Mark, M.A. & Greer, J.E. The VCR Tutor: Evaluating instructional effectiveness. Proceedings of the 13th Annual Conference of the Cognitive Science Society, 1991, 564 - 569.

Mayer, R.E. Educational psychology: A cognitive approach. Toronto: Little, Brown and Company, 1987.

McCalla, G.I & Greer, J.E. The practical use of artificial intelligence in automated tutoring: Current status and impediments to progress. In C.K. Leong & B.S. Randhawa (Eds.),

Understanding literacy and cognition: Theory, research and application. New York: Plenum Publishing, 1990.

McGraw, K.L. & Harbison-Briggs, K. Knowledge acquisition: Principles and guidelines. Englewood Cliffs, N. J.: Prentice-Hall, 1989.

Moore, J.L. An empirical study of pupils' attitudes to computers and robots. Journal of Computer-Assisted Learning, 1985, 1, 87-98.

Ohlsson, S. Some principles of intelligent tutoring. Instructional Science, 1986, 14, 293-326.

O'Neil, H.F. Jr. (Ed.), Learning strategies. New York: Academic Press, 1978.

O'Shea, T. A self-improving quadratic tutor. In D. Sleeman & J.S. Brown (Eds.), Intelligent tutoring systems. London: Academic Press, 1982.

Parry, J.D. & Hofmeister, A.M. The development and validation of an expert system for special educators. Learning Disability Quarterly, 1986, 9(2), 124-132.

Partridge, D. Artificial intelligence: Applications in the future of software engineering. New York: Ellis Horwood, 1986.

Patterson, A. C. & Bloch, B. Formative evaluation: A process required in computer-assisted instruction. Educational Technology, 1987, 27(11), 26-30.

Perkins, D.N. & Salomon, G. Transfer and teaching thinking. In D.N. Perkins, J. Lochhead, & J. Bishop, (Eds.) Thinking: The second international conference. Hillsdale, N.J.: Lawrence Erlbaum Associates, Publishers, 1987.

Richer, M.H. & Clancey, W.J. Guidon-watch: A graphic interface for viewing a knowledge-based system. In R. W. Lawler & M. Yazdani (Eds.), Artificial intelligence and education: Volume one, Learning environments and tutoring systems. Norwood, N.J.: Ablex Publishing, 1987.

Rosenberg, R. A critical analysis of research on intelligent tutoring systems. Educational Technology, 1987, 27(11), 7-13.

Rowland, P. & Stuessy, C.L. Matching mode of CAI to cognitive style: An exploratory study. The Journal of Computers in Mathematics and Science Teaching, 1988, 7, 36-40,55.

Salomon, G. Computers in education: Setting a research agenda. Educational Technology, 1984, 24(10), 7-11.

Schofield, J.W., Evans-Rhodes, D., & Huber, B.R. Artificial intelligence in the classroom: The impact of a computer-based tutor on teachers and students. Social Science Computer Review 1990, 8(1), 24-41.

Schooler, L.J. & Anderson, J. R. The disruptive potential of immediate feedback. The Twelfth Annual Conference of the Cognitive Science Society, 1990, 702 - 708.

Schmalhofer, F., Kuhn, O., Messamer, P. & Charron, R. An experimental evaluation of different amounts of receptive and exploratory learning in a tutoring system. Computers in Human Behaviour, 1990, 6, 51-68.

Scriven, M. The methodology of evaluation. In R.E. Stake (Ed.), Curriculum evaluation. Chicago: Rand-McNally, 1967.

Shute, V. J. & Glaser, R. A large-scale evaluation of an intelligent discovery world: Smithtown. Interactive Learning Environments, 1990, 1, 51-77.

Sleeman, D.H. & Brown, J.S. (Eds.). Intelligent tutoring systems. New York: Academic Press, 1982.

Smith, D. & Keep, R. Children's opinions of educational software. Educational Research, 1986, 28(2), 83-88.

Steinberg, E.S. Teaching computers to teach. Hillsdale, New Jersey: Lawrence Erlbaum Associates, 1984.

Swan, K. Logo programming and the teaching and learning of problem solving. Journal of Artificial Intelligence in Education, 1989, 1(1), 73-92.

Thomas, D. B. The effectiveness of computer-assisted instruction secondary schools. AEDS Journal, 1979, 12, 103-116.

Vasandani, V. & Govindaraj, T. An experimental evaluation of an intelligent tutor for diagnostic problem solving. In. L. Birnbaum (Ed.), The International Conference on the Learning Sciences, Charlottesville, VA: Association for the Advancement of Computing in Education, 1991.

Waugh, M. L. & Currier, D. Computer-based education: What we know and need to know. Journal of Computers in Mathematics and Science Teaching, 1986, 5(3), 13-15; 18.

Wenger, E. Artificial intelligence and tutoring systems. Los Altos, CA: Morgan Kaufmann, 1987.

Yang, J. S. Individualizing instruction through intelligent computer-assisted instruction: A perspective. Educational Technology, 1987, 27(3), 7-15.