CHAPTER 3
LITERATURE REVIEW

In this chapter, the study of previous research is presented. Section 3.1 discusses the previous intelligent system's. Section 3.2 introduces the technology of education. Section 3.3 discusses different types of student modeling architecture such as differential modeling and perturbation modeling. Section 3.4 discusses the reasoning method for knowledge representation such as rule-based reasoning and case-based reasoning.

3.1 A REVIEW OF INTELLIGENT SYSTEM
3.1.1 ISIS Tutor: An Intelligent Learning Environment (ILE) for CDS/ISIS Users

[Brusilovsky & Pesin, 1996] build ISIS Tutor for learning the print formatting language of an information retrieval system, CDS/ISIS for IBM PCs. ISIS Tutor is written in embedded ISIS-Pascal. The print formatting language is the key to many CDS/ISIS operations and mastering the language is important for effective use of the system. To display or print the result of a search or the content of a database need a sequence of print formatting command, which is a more complex program in the print formatting language. Print formatting commands can type a field of the current record or a part of field, manipulate the current output position, type a constant character string and so on. It is also used in indexing and sorting. There are over 50 different commands and modifiers in the print formatting language, and the tutoring system for the language is really helpful.

Brusilovsky and Pesin (1996) found that intelligent tutoring systems are able to control learning on various levels but do not provide tools to support free exploration. However, hypermedia system can provide the basis for an exploratory learning system but this need to be supplemented by more directed guidance. The guidance can be provided by an intelligent tutoring component. They suggest therefore that it is possible to achieve a good result by developing an educational computer system, which integrates the capabilities of hypermedia, learning environments and intelligent tutoring. According to their research at the Moscow State University, they found that there are two problems of creating integrated Intelligent Learning Environment which are the problem of adaptation and integration.
Most of the intelligent tutoring system and intelligent learning environment components can adapt its work in tutoring a given student but very few environment and manual components can do that. As for integration, it requires the continuity of student work. The result of student work with any of the components during the session should be taken into account by other components to adapt their performance to the changed knowledge level and current interest of the particular student. They solve these problems by designing a knowledge-based approach and a simple student model centered architecture for building an integrated intelligent learning environment. Using this approach, all the components of ILE use the same central knowledge base to coordinate its behavior and adapt it to the given student.

The following are the main components of the ISIS Tutor:-

- Domain model and Student model
It makes the system integrated and adaptive. ISIS Tutor uses the student model to adapt their work and update it to reflect the student progress. It uses overlay model which contain one integer counter for each subject knowledge element measuring student understanding. The overlay model is powerful to be used by different components of ILE.

- Tutor component
This component deals with three kinds of teaching operations which are concept presentation, examples and problems. Using the student model, the tutor can select an optimal teaching operation for the given student in each stage of learning.

- Hypermedia component
It supports student-driven acquisition of conceptual knowledge. Here, the component uses the student model to provide an adaptive navigation support for the given student and updates the student model to reflect the results of the student's work with the component. It uses colors and specials marks to annotate the set of links leading from the current node to related nodes according to the current user knowledge and educational goals.

- Learning environment
It allows the user to play and experiment with print formatting commands. It provides step-by-step execution and extended visualization. ISIS Tutor uses the adaptive navigation support in more general context adaptation in hypermedia.
3.1.2 Intelligent Web-Based Help Desk System

This system is implemented for a NASA help desk system, the user support organization of the Earth Observing System Data and Information System (EOSDIS). EOSDIS manages data from NASA Earth science and field measurement programs. It provides data archiving, distribution and information services to wide range of users including scientists, educators and student. It stores data collected from science satellites and experiments, develops and distributes tool for data manipulation and generates standard data products known as ‘data sets’ [Thurman, 1997].

The goal of intelligent web-based help desk system is to provide users direct access to the knowledge in the case base and enable the case-based reasoner to respond to request for assistants. If the request cannot response, it provides a medium through which help desk staff can concurrently answer a user request and create and index a new case into the help desk database. Several tools are used in designing and developing a web based help desk system. These include cased-based reasoning, cognitive engineering and the design of intelligent system.

**Case-based Reasoning.** A case-based reasoning system for a help desk begins with defining the structure and content of a case. After that, indexing structure and the means for adding new cases is addressed.

**Cognitive Engineering Models.** It is used to support the initial design and long-term maintenance of a case-based help desk. It define both the structure and content of help desk case-base and the interfaces for both users who seek assistance and the help desk staff who maintain and enhance the computer-based help system.
Intelligent System. It is defined as a system, which does not make the same mistake again. When the case-based help desk requires human intervention to address a novel assistance request, the interface for the help desk staff allows the staff to concurrently answer a request and update the case-based reasoning system. The latter requires the software to automatically create a new case and new indices and allows the help desk staff to inspect, repair and validate the new addition to ensure that it is correct and consistent with other cases and indices. The available technologies which have been used in the development of web-based help desk system include the World Wide Web, the Java programming language and commercial cased-based reasoning systems and database management systems.

World Wide Web. WWW enables the distribution of information to a varied and widespread user community and makes it an attractive medium to provide help to geographically distributed users.

Java. Java can run on most platforms and with most Web browsers. It’s ability to create platform-independent, distributed applications makes it a critical component for Web-base help desk architecture.

Case-Based Reasoning Systems. There are several case-based reasoning tools including Inference Corporation’s CBR Express and CasePoint, Cognitive Systems’ REMIND and Esteem Systems’ ESTEEM. Some system are still implemented in Lisp, others stores cases in flat files rather than a database and few provide a distributed client-server architecture that can operate over the Internet.

Databases. The relational databases such as Oracle and Sybase are alternatives to the case storage feature of commercial case-based reasoning systems.
3.1.3 Case-based Mode Management Tutor (CB-MMT)

The case-based mode management tutor is designed to teach advanced mode management skills to Boeing 757/767 pilots. Its combines the benefits of intelligent tutoring system and case based teaching. Case-based scenarios focus instruction on known mode management problems and situations. It uses storytelling to teach. The stories provide information and illustrate the consequences or importance of that information [Chappell, 1995]. It is an intelligent tutoring system (ITS) that focuses on the more general problem of mode management. The design of the Mode Management Tutor takes advantages of an ITS capabilities to tailor instruction to individual student needs [Chappell & Mitchell, 1997].

The system is built in two phases. The first phase identifies specific knowledge and skills differences between trained novices and experts. The second phase build a training system to address those differences. By using computer to present knowledge within a simulation environment allows a training system to illustrate knowledge in context and to help pilots integrate knowledge through practice [Chappell & Mitchell, 1997]. Figure 3.1 [Chappell & Mitchell, 1997] shows the architecture of the Case-Based Mode Management Tutor.

![Figure 3.1: Design of the Case-Based Mode Management Tutor](image-url)
The student interacts with both the tutoring system and the aircraft simulation. The tutor presents case scenarios that demonstrate teaching objectives and then monitors the student's performance as the scenario is flown in the simulator [Chappell & Mitchell, 1997].

3.2 TECHNOLOGY EDUCATION

Over the last decade, people refer their problem using persons or book. With the development of computer technologies, these reference channels are becoming less in demand. In addition, this technology is widely used in education not only at the secondary level, but also in high school, colleagues and universities. Not everybody however found the technology helpful. Sometimes, users only understand the importance of technology after finishing some courses.

The aim of technology education is to prepare students to become technologically literate citizens. To achieve this goal, technology education must prepare students to understand, control and use technology. The students believe that technology was more difficult to work and female students consistently perceived technology to be less interesting than male students did [Boser, et-al. 1998].

Otherwise, the students who getting better understanding about the computer technology found that it helps them effectively. The expansion of the computer technology enables them to obtain the lesson from the compact disk and an Internet. As [Webster & Todd, 1998] provide an interactive demonstration for statistics education on the World Wide Web, it can help students in introductory courses better understand some of the more difficult concepts that they encounter. It also allow a students to repeatedly practice their work and keep track of their results and allow student to see the mark of their performance.

Another example is a study by [Laviolette, 1994] where he describes some ways in which the computer, both computationally and graphically, can help provide students with an insight into basic regression concepts such as the underlying model with random error, outliers, influential observations, and interpretation of multiple regression coefficients. He found that the gap between theory and practice could be bridged by using software to illustrate the underlying concepts, rather than only as an analysis tool. Software as a teaching tool benefits students by replacing abstract lectures with concrete examples, thus contributing to greater understanding and better analysis.
3.3 STUDENT MODELING ARCHITECTURE

Student modeling architecture is used to record either misconceptions, missing conceptions or a combination of both. A misconception is knowledge that the student has but not the expert. Missing conceptions is knowledge which the expert posses but not the student.

3.3.1 Differential Modeling

The differential model is a modification of the overlay model. This model does not compare the knowledge of the student and the expert but their performance in the particular situation. It divides system knowledge into two classes:-

1. Knowledge the student should know
2. Knowledge the expert used to produce its behavior and the student could not be expected to know.

The differential student model partitions the domain knowledge into already presented knowledge and that which has not been presented to the student [Smith, 1998]. It does not cater for student misconception or bugs. The example of system that uses the differential modeling technique is the WEST game by Burton and Brown (1978). Figure 3.2 [Smith, 1998] shown a differential student model:-

![Diagram](image)

Figure 3.2: A Representation of a Differential Student Model.

WEST is one of the tutoring systems that employ a differential student model. It is a tutoring system that monitors a student while playing a game designed to encourage the use of basic arithmetic skills.
3.3.2 Perturbation Modeling

This model caters for knowledge possessed by the student that is not present in the expert domain knowledge. The perturbation model extends the experts knowledge with the addition of a bugs library. The process to create a bug library can be enumerative or generative [Smith, S. 1998]. The enumerative process lists all possible bugs usually via an analysis of the problem domain and the errors that students make. The generative process attempts to generate bugs from an underlying cognitive theory. The examples of system that use perturbation modeling are DEBUGGY, LMS and PROUST. Figure 3.3 [Smith, 1998] shows the perturbation modeling.

![Diagram of the perturbation model]

Figure 3.3: A Representation of the Perturbation or Buggy Student Model.
3.4 REASONING METHOD

3.4.1 Rule-based Reasoning

Rule-based reasoning is the most commonly used form of representation used in expert system. Knowledge is represented as IF THEN rules. These are essentially association pairs. For example, IF a particular fact is in the database, THEN the action is taken.

The following describe the characteristics of rule-based reasoning:
1. It includes general problem solving through interpreting and processing rules
2. Data of interest is contained on a database, a human user or online
3. The system operator decides which rules to review
4. Rules are of the form IF condition THEN action

A ruled-based reasoning consists of three components:
1. It contains discrete fragments of knowledge which are expressed in IF – THEN associations over variables.
2. The working memory which contains all facts that are true at any stage in the computational process
3. The inference engine which is the domain and knowledge independent processing mechanism.

The Inference Engine searches the rule base and matches variable values in the working memory to the preconditions and conclusions of rules in the rule base. When a match occurs, the rule is said to have fired. During the process, if a rule is traversed from left to right, trying to infer conclusions from known conditions, the strategy is called forward chaining. Otherwise if the rule is traversed from right to left, trying to evaluate conclusions by checking if the conditions are met, the strategy is called backward chaining. Forward chaining is useful when the objective is to find implications of new pieces of information while backward chaining is useful when the objective is to validate hypotheses. When more than one rule is applicable in a situation, the inference engine uses conflict resolution to decide which to fire.
3.4.2 Case-based Reasoning

Case-based reasoning can be defined as the way to solve a new problem by remembering a previous similar situation and reusing information and knowledge of that situation [Aamodt & Plaza, 1994]. It is an AI paradigm for solving problems. It is an intelligent systems method that enables information managers to increase efficiency and reduce cost by substantially automating processes such as diagnosis, scheduling and design. It stores its intelligence as a set of experiences or 'cases'. A case-based reasoning system stores memorable problem and their solutions in its knowledge base and retrieves them when it encounters a similar problem. It is based on the theory that decision-makers recall the past experiences in order to solve a new problems. It can be thought of as a computational implementation of recognition-primed decision making. Recognition-primed decision making is an emerging theory of human problem solving and decision making research [Thurman, 1997].
There are two parts of a case based that allow it to understand a problem based on past experience which are recalling an old experience and interpreting the new situation in terms of the recalled experiences. The first interpretation is called the indexing problem, which recalls the situation most similar to a new situation. The second interpretation includes comparison of the new situation with the previous situation and if necessary, adapting the old experience to fit the new situation. A case-based reasoning system learns in two ways, which are through the accumulation of new cases, and through the modification and extension of indices [Thurman, 1997].

A general CBR cycle may be described by the following four processes:-

1. *Retrieve* the most similar cases
2. *Reuse* the information and knowledge in that case to solve the problem
3. *Revise* the proposed solution
4. *Retain* the parts of this experience likely to be useful for future problem solving
This is diagrammatically shown in Figure 3.5.