A SURVEY ON ARTIFICIAL INTELLIGENCE AND EXPERT SYSTEM

Naveen Kumar Singh, A.P Department of Computer Application,ABESEC ,Ghaziabad
Dr Rakesh Kumar, Professor, Accurate Engineering college Ghaziabad

ABSTRACT

Artificial intelligence is the simulation of human intelligence processes by machines, especially computer systems. These processes include learning (the acquisition of information and rules for using the information), reasoning (using the rules to reach approximate or definite conclusions), and self-correction. Particular applications of AI include expert systems, speech recognition, and image recognition.

Keywords*: Artificial, System, Intelligence, Applications, Learning, Inference, Image*

INTRODUCTION

Expert Systems are computer programs that are derived from a branch of computer science research called *Artificial Intelligence* (AI). AI's scientific goal is to understand intelligence by building computer programs that exhibit intelligent behavior. It is concerned with the concepts and methods of symbolic inference, or reasoning, by a computer, and how the knowledge used to make those inferences will be represented inside the machine. of course, the term *intelligence* covers many cognitive skills, including the ability to solve problems, learn, and understand language; AI addresses all of those. But most progress to date in AI has been made in the area of problem.

The solving -- concepts and methods for building programs that *reason* about problems rather than calculate a solution.

AI programs that achieve expert-level competence in solving problems in task areas by bringing to bear a body of knowledge about specific tasks are called *knowledge-based* or *expert systems*. Often, the term expert systems is reserved for programs whose knowledge base contains the knowledge used by human experts, in contrast to knowledge gathered from textbooks or non-experts. More often than not, the two terms, expert systems (ES) and knowledge-based systems (KBS), are used synonymously. Taken together, they represent the most widespread type of AI application. The area of human intellectual endeavor to be captured in an expert system is calledthe *task domain*. *Task* refers to some goal-oriented, problem-solving activity. *Domain* refers to the area within which the task is being performed. Typical tasks are diagnosis, planning, scheduling, configuration and design. An example of a task domain is aircraft crew.

The Problem Outlined

Research of the applications of AI is coming to a point where the approaches, algorithms and models need to be applied for further understanding of their usefulness and applicability towards anygiven project for a UAV [6], [7]. Most of the research papers reviewed gave excellent theory but did not give enough information on the mechanics or steps for applying the methods outlined to begin a research effort.

The focus of this research has been to outline some of the A.l techniques used for UAV flight control and discuss some of the tools used to apply.

EXPERT SYSTEMS AND ARTIFICIAL INTELLIGENCE

Expert Systems are computer programs that are derived from a branch of computer science research called *Artificial Intelligence* (AI). AI's scientific goal is to understand intelligence by building computer programs that exhibit intelligent behavior. It is concerned with the concepts and methods of symbolic inference, or reasoning, by a computer, and how the knowledge used to make those inferences will be represented inside the machine.

Of course, the term *intelligence* covers many cognitive skills, including the ability to solve problems, learn, and understand language; AI addresses all of those. But most progress to date in AI has been made in the area of problem solving -- concepts and methods for building programs that *reason* about problems rather than calculate a solution.

AI programs that achieve expert-level competence in solving problems in task areas by bringing to bear a body of knowledge about specific tasks are called *knowledge-based* or *expert systems*. Often, the term expert systems is reserved for programs whose knowledge base contains the knowledge used by human experts, in contrast to knowledge gathered from textbooks or non-experts. More often than not, the two terms, expert systems (ES) and knowledge-based systems (KBS), are used synonymously. Taken together, they represent the most widespread type of AI application. The area of human intellectual endeavor to be captured in an expert system is called the *task domain*. *Task* refers to some goal-oriented, problem-solving activity. *Domain* refers to the area within which the task is being performed. Typical tasks are diagnosis, planning, scheduling, Building an expert system is known as *knowledge engineering* and its practitioners are called *knowledge engineers*. The knowledge engineer must make sure that the computer has all the knowledge needed to solve a problem. The knowledge engineer must choose one or more forms in which to represent the required knowledge as symbol patterns in the memory of the computer -- that is, he (or she) must choose a *knowledge representation*. He must also ensure that the computer can use the knowledge efficiently by selecting from a handful of *reasoning methods*. The practice of knowledge engineering is described later. We first describe the components of expert systems. Building Blocks of Expert Systems.

Every expert system consists of two principal parts: the knowledge base; and the reasoning, or inference, engine.

The *knowledge base* of expert systems contains both factual and heuristic knowledge. *Factual knowledge* is that knowledge of the task domain that is widely shared, typically found in textbooks or journals, and commonly agreed upon by those knowledgeable in the particularfield.

Heuristic knowledge is the less rigorous, more experiential, more judgmental knowledge of performance. In contrast to factual knowledge, heuristic knowledge is rarely discussed, and is largely individualistic. It is the knowledge of good practice, good judgment, and plausible reasoning in the field. It is the knowledge that underlies the "art of good guessing."

Knowledge representation formalizes and organizes the knowledge. One widely used representation is the *production rule*, or simply *rule*. A rule consists of an IF part and a THEN part (also called a *condition* and an *action*). The IF part lists a set of conditions in some logical combination. The piece of knowledge represented by the production rule is relevant to the line of reasoning being developed if the IF part of the rule is satisfied; consequently, the THEN part can be concluded, or its problem-solving action taken. Expert systems whose knowledge is represented in rule form are called *rule-based systems*.

Another widely used representation, called the *unit*(also known as *frame*, *schema*, or *list structure*) is based upon a more passive view of knowledge. The unit is an assemblage of associated symbolic knowledge about an entity to be represented. Typically, a unit consists of a list of properties of the entity and associated values for those properties.

Since every task domain consists of many entities that stand in various relations, the properties can also be used to specify relations, and the values of these properties are the names of other units that are linked according to the relations. One unit can also represent knowledge that is a "special case" of another unit, or some units can be "parts of" anotherunit.

The *problem-solving model*, or *paradigm*, organizes and controls the steps taken to solve the problem. One common but powerful paradigm involves chaining of IF-THEN rules to form a line of reasoning. If the chaining starts from a set of conditions and moves toward some conclusion, the method is called *forward chaining*. If the conclusion is known (for example, a goal to be achieved) but the path to that conclusion is not known, then reasoning backwards is called for, and the method is *backward chaining*. These problem-solving methods are built into program modules called *inference engines* or *inference procedures* that manipulate and use knowledge in the knowledge base to form a line of reasoning.

The *knowledge base* an expert uses is what he learned at school, from colleagues, and from years of experience. Presumably the more experience he has, the larger his store of knowledge. Knowledge allows him to interpret the information in his databases to advantage in diagnosis, design, and analysis.

Though an expert system consists primarily of a knowledge base and an inference engine, a couple of other features are worth mentioning: reasoning with uncertainty, and explanation of the line of reasoning.

Knowledge is almost always incomplete and uncertain. To deal with uncertain knowledge, a rule may have associated with it a *confidence factor* or a weight. The set of methods for using uncertain knowledge in combination with uncertain data in the reasoning process is called *reasoning with uncertainty*. An important subclass of methods for reasoning with uncertainty is called "fuzzy logic," and the systems that use them are known as "fuzzy systems."

Because an expert system uses uncertain or heuristic knowledge (as we humans do) its credibility is often in question (as is the case with humans). When an answer to a problem is questionable, we tend to want to know the rationale. If the rationale seems plausible, we tend to believe the answer. So it is with expert systems. Most expert systems have the ability to answer questions of the form: "Why is the answer X?" Explanations can be generated by tracing the line of reasoning used by the inference engine (Feigenbaum, McCorduck et al.1988).

The most important ingredient in any expert system is knowledge. The power of expert systems resides in the specific, high-quality knowledge they contain about task domains. AI researchers will continue to explore and add to the current repertoire of knowledge representation and reasoning methods. But in knowledge resides the power. Because of the importance of knowledge in expert systems and because the current knowledge acquisition method is slow and tedious, much of the future of expert systems depends on breaking the knowledge acquisition bottleneck and in codifying and representing a large knowledge infrastructure.

Knowledge engineering

It is the art of designing and building expert systems, and knowledge engineers are its practitioners. Gerald M. Weinberg said of programming in *The Psychology of Programming*: "'Programming,' -- like 'loving,' -- is a single word that encompasses infinitude of activities" (Weinberg 1971). Knowledge engineering is the same, perhaps more so. We stated earlier that knowledge engineering is an applied part of the science of artificial intelligence which, in turn, is a part of computer science. Theoretically, then, a knowledge engineer is a computer scientist who knows how to design and implement programs that incorporate artificial intelligence techniques. The nature of knowledge engineering is changing, however, and a new breed of knowledge engineers is emerging. We'll discuss the evolving nature of knowledge engineering later.

Today there are two ways to build an expert system. They can be built from scratch, or built using a piece of development software known as a "tool" or a "shell." Before we discuss these tools, let's briefly discuss what knowledge engineers do. Though different styles and methods of knowledge engineering exist, the basic approach is the same: a knowledge engineer interviews and observes a human expert or a group of experts and learns what the experts know, and how they reason with their knowledge. The engineer then translates the knowledge into a computer-usable language, and designs an inference engine, a reasoning structure, that uses the knowledge appropriately. He also determines how to integrate the use of uncertain knowledge in the reasoning process, and what kinds of explanation would be useful to the end user.

Next, the inference engine and facilities for representing knowledge and for explaining are programmed, and the domain knowledge is entered into the program piece by piece. It may be that the inference engine is not just right; the form of knowledge representation is awkward for the kind of knowledge needed for the task; and the expert might decide the pieces of knowledge are wrong. All these are discovered and modified as the expert system gradually gains competence.

The discovery and cumulation of techniques of machine reasoning and knowledge representation is generally the work of artificial intelligence research. The discovery and cumulation of knowledge of a task domain is the province of domain experts. Domain knowledge consists of both formal, textbook knowledge, and experiential knowledge -- the *expertise* of the experts.

Tools, Shells, and Skeletons

Compared to the wide variation in domain knowledge, only a small number of AI methods are known that are useful in expert systems. That is, currently there are only a handful of ways in which to represent knowledge, or to make inferences, or to generate explanations. Thus, systems can be built that contain these useful methods without any domain-specific knowledge. Such systems are known as *skeletal systems, shells*, or simply *AI tools*.

Building expert systems by using shells offers significant advantages. A system can be built to perform a unique task by entering into a shell all the necessary knowledge about a task domain. The inference engine that applies the knowledge to the task at hand is built into the shell. If the program is not very complicated and if an expert has had some training in the use of a shell, the expert can enter the knowledge himself.

Many commercial shells are available today, ranging in size from shells on PCs, to shells on workstations, to shells on large mainframe computers. They range in price from hundreds to tens of thousands of dollars, and range in complexity from simple, forward-chained, rule-based systems requiring two days of training to those so complex that only highly trained knowledge engineers can use them to advantage. They range from general-purpose shells to shells custom-tailored to a class of tasks, such as financial planning or real-time processcontrol.

Although shells simplify programming, in general they don't help with knowledge acquisition. *Knowledge acquisition* refers to the task of endowing expert systems with knowledge, a task currently performed by knowledge engineers. The choice of reasoning method, or a shell, is important, but it isn't as important as the accumulation of high-quality knowledge. The power of an expert system lies in its store of knowledge about the task domain -- the more knowledge a system is given, the more competent it becomes.

Bricks and Mortar

The fundamental working hypothesis of AI is that intelligent behavior can be precisely described as symbol manipulation and can be modeled with the symbol processing capabilities of the computer.

In the late 1950s, special programming languages were invented that facilitate symbol manipulation. The most prominent is called LISP (LISt Processing). Because of its simple elegance and flexibility, most AI research programs are written in LISP, but commercial applications have moved away from LISP.

In the early 1970s another AI programming language was invented in France. It is called PROLOG (PROgramming in LOGic). LISP has its roots in one area of mathematics (lambda calculus), PROLOG in another (first-order predicatecalculus).

PROLOG consists of English-like statements which are facts (assertions), rules (of inference), and questions. Here is an inference rule: "If object-x is part-of object-y then a component-of object-y is object-x."

Programs written in PROLOG have behavior similar to rule-based systems written in LISP. PROLOG, however, did not immediately become a language of choice for AI programmers. In the early 1980s it was given impetus with the announcement by the Japanese that they would use a logic programming language for the Fifth Generation Computing Systems (FGCS) Project. A variety of logic-based programming languages have since arisen, and the term *prolog* has becomegeneric.

Knowledge Based Expert Systems - An Overview

The Concept of Knowledge Based Expert Systems

To begin the development of an expert system it is relevant to address the following basic questions:

(i) What is an expert system?

Expert (Latin: *expertus*): *1. experienced; 2. having, involving, or displaying special skills or knowledge derived fromtraining or experience.*[50, page 409]

An expert system is defined in the IBM Dictionary of Computing [48, page252] as:

"A system that provides for solving problems in a particular application areaby drawing inferences from a knowledge base acquired by human expertise.

Synonymous with knowledge-based system."

"Expert systems are a class of computer programs that can advise, analyze,categorize, communicate, consult, design, diagnose, explain, explore,forecast, form concepts, identify, interpret, justify, learn, manage, monitor,plan, present, retrieve, schedule, test, and tutor. They address problems normally thought to require human specialists for their solution."

A concise description of an expert system is given by Luger & Stubblefield

[45, page 291]:Page 6

"An expert system is a knowledge-based program that provides "expertquality" solutions to problems in a specific domain."

An expert system is a computer program that uses expertise to assist people in performing a wide variety of functions, including diagnosis, planning, scheduling and design. It deals with everything from the diagnosis of human diseases to the diagnosis of a malfunction on a space shuttle. Its programmers use the expertise of one or several human specialists to create a tool that can be used by a layperson to solve difficult or ambiguous problems.

A chief advantage of expert systems is their low cost compared with the expense of paying an expert or team of specialists. A user-friendly interface to the system allows the user to specify symptoms and to clarify the problem in response to questions asked by the system. The goal is to lead the user to discover a solution to the problem.

CONCLUSION

The basic idea of this paper is to explore the artificial intelligence and to derived the expert system . How the artificial intelligence and expert system is correlated .what are further scope for the research of this field The two main components of an expert system are:! the knowledge base, which differs from a database in that it contains executable program code (instructions) and ! the inference engine, which interprets and evaluates the instructions e and data in the knowledge base.

REFERENCES

- Paul R. Watkins and Lance B. Elliot (Editors), (1993), "*Expert Systems in Business and Finance: Issues and Applications*, Chichester: John Wiley andSons.
- Miklos A. Vasarhelyi and B.N. Srinidhi (Editors), (1993) "*Artificial Intelligence in Accounting and Auditing: Using Expert Systems".* Volume II, Princeton, NJ: Markus Wiener Publishing,1993.

Authors would like to extend their heartfelt thanks to the academic and infrastructural support received from their respective Dept./University.