

Chapter 11

Expert Systems and Applied Artificial Intelligence

11.1 What is Artificial Intelligence?

The field of artificial intelligence (AI) is concerned with methods of developing systems that display aspects of intelligent behaviour. These systems are designed to imitate the human capabilities of thinking and sensing.

Characteristics of AI Systems

Characteristics of AI systems include:

1. Symbolic Processing

In AI applications, computers process symbols rather than numbers or letters. AI applications process strings of characters that represent real-world entities or concepts. Symbols can be arranged in structures such as lists, hierarchies, or networks. These structures show how symbols relate to each other.

2. Nonalgorithmic Processing

Computer programs outside the AI domain are programmed algorithms; that is, fully specified step-by-step procedures that define a solution to the problem. The actions of a knowledge-based AI system depend to a far greater degree on the situation where it is used.

The Field of AI

Artificial intelligence is a science and technology based on disciplines such as computer science, biology, psychology, linguistics, mathematics, and engineering. The goal of AI is to develop computers that can think, see, hear, walk, talk and feel. A major thrust of AI is the development of computer functions normally associated with human intelligence, such as reasoning, learning, and problem solving.

How the AI Field Evolved [Figure 11.2]

1950 Turing Test - a machine performs intelligently if an interrogator using remote terminals cannot distinguish its responses from those of a human.

Result: General problem-solving methods

1960 AI established as research field.

Result: Knowledge-based expert systems

1970 AI commercialization began

Result: Transaction processing and decision support systems using AI.

1980 Artificial neural networks

Result: Resembling the interconnected neuronal structures in the human brain

1990 Intelligent agents

Result: Software that performs assigned tasks on the users behalf

11.2 Capabilities of Expert Systems: General View

The most important applied area of AI is the field of expert systems. An *expert system* (ES) is a knowledge-based system that employs knowledge about its application domain and uses an inferencing (reason) procedure to solve problems that would otherwise require human competence or expertise. The power of expert systems stems primarily from the specific knowledge about a narrow domain stored in the expert system's *knowledge base*.

It is important to stress to students that expert systems are assistants to decision makers and not substitutes for them. Expert systems do not have human capabilities. They use a knowledge base of a particular domain and bring that knowledge to bear on the facts of the particular situation at hand. The knowledge base of an ES also contains *heuristic knowledge* - rules of thumb used by human experts who work in the domain.

11.3 Applications of Expert Systems

The text outlines some illustrative minicases of expert systems applications. These include areas such as high-risk credit decisions, advertising decision making, and manufacturing decisions.

Generic Categories of Expert System Applications

Table 11.1 outlines the generic areas of ES applications where ES can be applied. Application areas include classification, diagnosis, monitoring, process control, design, scheduling and planning, and generation of options.

Classification - identify an object based on stated characteristics

Diagnosis Systems - infer malfunction or disease from observable data

Monitoring - compare data from a continually observed system to prescribe behaviour

Process Control - control a physical process based on monitoring

Design - configure a system according to specifications

Scheduling & Planning - develop or modify a plan of action

Generation of Options - generate alternative solutions to a problem

11.4 How Expert Systems Work

The strength of an ES derives from its **knowledge base** - an organized collection of facts and heuristics about the system's domain. An ES is built in a process known as **knowledge engineering**, during which knowledge about the domain is acquired from human experts and other sources by knowledge engineers.

The accumulation of knowledge in knowledge bases, from which conclusions are to be drawn by the inference engine, is the hallmark of an expert system.

Knowledge Representation and the Knowledge Base

The knowledge base of an ES contains both factual and heuristic knowledge. **Knowledge representation** is the method used to organize the knowledge in the knowledge base. Knowledge bases must represent notions as actions to be taken under circumstances, causality, time, dependencies, goals, and other higher-level concepts.

Several methods of knowledge representation can be drawn upon. Two of these methods include:

1. Frame-based systems

- are employed for building very powerful ESs. A frame specifies the attributes of a complex object and frames for various object types have specified relationships.

2. Production rules

- are the most common method of knowledge representation used in business. ***Rule-based expert systems*** are expert systems in which the knowledge is represented by production rules.

A production rule, or simply a rule, consists of an IF part (a condition or premise) and a THEN part (an action or conclusion). IF condition THEN action (conclusion).

The ***explanation facility*** explains how the system arrived at the recommendation. Depending on the tool used to implement the expert system, the explanation may be either in a natural language or simply a listing of rule numbers.

Inference Engine [Figure 11.4]

The inference engine:

1. Combines the facts of a specific case with the knowledge contained in the knowledge base to come up with a recommendation. In a rule-based expert system, the inference engine controls the order in which production rules are applied (A fired@) and resolves conflicts if more than one rule is applicable at a given time. This is what A reasoning@ amounts to in rule-based systems.
2. Directs the user interface to query the user for any information it needs for further inferencing.

The facts of the given case are entered into the ***working memory***, which acts as a blackboard, accumulating the knowledge about the case at hand. The inference engine repeatedly applies the rules to the working memory, adding new information (obtained from the rules conclusions) to it, until a goal state is produced or confirmed.

Figure 11.5 One of several strategies can be employed by an inference engine to reach a conclusion. Inferencing engines for rule-based systems

generally work by either forward or backward chaining of rules. Two strategies are:

1. *Forward chaining*

- is a data-driven strategy. The inferencing process moves from the facts of the case to a goal (conclusion). The strategy is thus driven by the facts available in the working memory and by the premises that can be satisfied. The inference engine attempts to match the condition (IF) part of each rule in the knowledge base with the facts currently available in the working memory. If several rules match, a conflict resolution procedure is invoked; for example, the lowest-numbered rule that adds new information to the working memory is fired. The conclusion of the firing rule is added to the working memory.

Forward-chaining systems are commonly used to solve more open-ended problems of a design or planning nature, such as, for example, establishing the configuration of a complex product.

2. *Backward chaining*

- the inference engine attempts to match the assumed (hypothesized) conclusion - the goal or subgoal state - with the conclusion (THEN) part of the rule. If such a rule is found, its premise becomes the new subgoal. In an ES with few possible goal states, this is a good strategy to pursue.

If a hypothesized goal state cannot be supported by the premises, the system will attempt to prove another goal state. Thus, possible conclusions are reviewed until a goal state that can be supported by the premises is encountered.

Backward chaining is best suited for applications in which the possible conclusions are limited in number and well defined. Classification or diagnosis type systems, in which each of several possible conclusions can be checked to see if it is supported by the data, are typical applications.

Uncertainty and Fuzzy Logic

Fuzzy logic is a method of reasoning that resembles human reasoning since it allows for approximate values and inferences and incomplete or ambiguous data (fuzzy data). Fuzzy logic is a method of choice for handling uncertainty in some expert systems.

Expert systems with fuzzy-logic capabilities thus allow for more flexible and creative handling of problems. These systems are used, for example, to control manufacturing processes.

11.5 Expert System Technology [Figure 11.6]

There are several levels of ES technologies available. Two important things to keep in mind when selecting ES tools include:

1. The tool selected for the project has to match the capability and sophistication of the projected ES, in particular, the need to integrate it with other subsystems such as databases and other components of a larger information system.
2. The tool also has to match the qualifications of the project team.

Expert systems technologies include:

1. *Specific expert systems*

- These expert systems actually provide recommendations in a specific task domain.

2. *Expert system shells*

- are the most common vehicle for the development of specific ESs. A shell is an expert system without a knowledge base. A shell furnishes the ES developer with the inference engine, user interface, and the explanation and knowledge acquisition facilities.

Domain-specific shells are actually incomplete specific expert systems, which require much less effort in order to field an actual system.

3. *Expert system development environments*

- these systems expand the capabilities of shells in various directions. They run on engineering workstations, minicomputers, or mainframes; offer tight integration with large databases; and support the building of large expert systems.

4. *High-level programming languages*

Several ES development environments have been rewritten from LISP into a procedural language more commonly found in the commercial

environment, such as C or C++. ESs are now rarely developed in a programming language.

11.6 Roles in Expert System Development

Three fundamental roles in building expert systems are:

1. **Expert** - Successful ES systems depend on the experience and application of knowledge that the people can bring to it during its development. Large systems generally require multiple experts.
2. **Knowledge engineer** - The knowledge engineer has a dual task. This person should be able to elicit knowledge from the expert, gradually gaining an understanding of an area of expertise. Intelligence, tact, empathy, and proficiency in specific techniques of knowledge acquisition are all required of a knowledge engineer. Knowledge-acquisition techniques include conducting interviews with varying degrees of structure, protocol analysis, observation of experts at work, and analysis of cases.

On the other hand, the knowledge engineer must also select a tool appropriate for the project and use it to represent the knowledge with the application of the *knowledge acquisition facility*.

3. **User** - A system developed by an end user with a simple shell, is built rather quickly and inexpensively. Larger systems are built in an organized development effort. A prototype-oriented iterative development strategy is commonly used. ESs lend themselves particularly well to prototyping.

11.7 Development and Maintenance of Expert Systems [Figure 11.7]

Steps in the methodology for the iterative process of ES development and maintenance include:

1. Problem Identification and Feasibility Analysis:
 - the problem must be suitable for an expert system to solve it.
 - must find an expert for the project
 - cost-effectiveness of the system has to be established (feasibility)
2. System Design and ES Technology Identification:

- the system is being designed. The needed degree of integration with other subsystems and databases is established
- concepts that best represent the domain knowledge are worked out
- the best way to represent the knowledge and to perform inferencing should be established with sample cases

3. Development of Prototype:

- knowledge engineer works with the expert to place the initial kernel of knowledge in the knowledge base.
- knowledge needs to be expressed in the language of the specific tool chosen for the project

4. Testing and Refinement of Prototype:

- using sample cases, the prototype is tested, and deficiencies in performance are noted. End users test the prototypes of the ES.

5. Complete and Field the ES:

- the interaction of the ES with all elements of its environment, including users and other information systems, is ensured and tested.
- ES is documented and user training is conducted

6. Maintain the System:

- the system is kept current primarily by updating its knowledge base.
- interfaces with other information systems have to be maintained as well, as those systems evolve.

11-8 Expert Systems in Organizations: Benefits and Limitations

Expert systems offer both tangible and important intangible benefits to owner companies. These benefits should be weighted against the development and exploitation costs of an ES, which are high for large, organizationally important ESs.

Benefits of Expert Systems

An ES is no substitute for a knowledge worker's overall performance of the problem-solving task. But these systems can dramatically reduce the amount of work the individual must do to solve a problem, and they do leave people with the creative and innovative aspects of problem solving.

Some of the possible organizational benefits of expert systems are:

1. An Es can complete its part of the tasks much faster than a human expert.
2. The error rate of successful systems is low, sometimes much lower than the human error rate for the same task.
3. ESs make consistent recommendations
4. ESs are a convenient vehicle for bringing to the point of application difficult-to-use sources of knowledge.
5. ESs can capture the scarce expertise of a uniquely qualified expert.
6. ESs can become a vehicle for building up organizational knowledge, as opposed to the knowledge of individuals in the organization.
7. When use as training vehicles, ESs result in a faster learning curve for novices.
8. The company can operate an ES in environments hazardous for humans.

Limitations of Expert Systems

No technology offers an easy and total solution. Large systems are costly and require significant development time and computer resources. ESs also have their limitations which include:

1. Limitations of the technology
2. Problems with knowledge acquisition
3. Operational domains as the principal area of ES application
4. Maintaining human expertise in organizations

11-9 Overview of Applied Artificial Intelligence

Expert systems are only one area of AI. Other areas include:

1. Natural language processing

2. Robotics
3. Computer vision
4. Computerized speech recognition
5. Machine learning

Natural Language Processing

Being able to talk to computers in conversational human languages and have them understand us is a goal of AI researchers. Natural language processing systems are becoming common. The main application for natural language systems at this time is as a user interface for expert and database systems.

Robotics

AI, engineering, and physiology are the basic disciplines of robotics. This technology produces robot machines with computer intelligence and computer-controlled, human like physical capabilities, robotics applications

Computer Vision

The simulation of human senses is a principal objective of the AI field. The most advanced AI sensory system is computer vision, or visual scene recognition. The task of a vision system is to interpret the picture obtained. These systems are employed in robots or in satellite systems. Simpler vision systems are used for quality control in manufacturing.

Speech Recognition

The ultimate goal of the corresponding AI area is computerized speech recognition, or the understanding of connected speech by an unknown speaker, as opposed to systems that recognize words or short phrases spoken one at a time or systems are trained by a specific speaker before use.

Machine Learning

A system with learning capabilities - machine learning - can automatically change itself in order to perform the same tasks more efficiently and more effectively the next time.

A number of approaches to learning are being investigated. Approaches include:

1. Problem Solving Learning - accumulate Aexperience@ about its rules in terms of their contributions to correct advice. The rules that do not contribute or those that are found to provide doubtful contributions could be automatically discarded or assigned low certainty factors.
2. Case-Based Learning - collecting cases in a knowledge base and solving problems by seeking out a case similar to the one to be solved.
3. Inductive Learning - learning from examples. In this case, a system is able to generate its knowledge, represented as rules.

11-10 Neural Networks

Neural networks are computing systems modelled on the human brain's mesh-like network of interconnected processing elements, called neurons. Of course, neural networks are much simpler than the human brain (estimated to have more than 100 billion neuron brain cells). Like the brain, however, such networks can process many pieces of information simultaneously and can learn to recognize patterns and programs themselves to solve related problems on their own.

A **neural network** is an array of interconnected processing elements, each of which can accept inputs, process them, and produce a single output with the objective of imitating the operation of the human brain. Knowledge is represented in a neural network by the pattern of connections among the processing elements and by adjusting weights of these connections.

The strength of neural networks is in applications that require sophisticated pattern recognition. The greatest weakness of neural networks is that they do not furnish an explanation for the conclusions they make.

In summary, a neural network can be trained to recognize certain patterns and then apply what it learned to new cases where it can discern the patterns.