ARTIFICIAL INTELLIGENCE & MACHINE LEARNING

UNIT – 4

Dr. G. Ravi & Dr. S. Peerbasha
PG & Research Department of Computer Science
Jamal Mohamed College (Autonomous)
Trichy-620 020.

UNIT - 4

LEARNING

LEARNING:

- Learning is the process by which a system improves its performance from experience or environment. Learning may also be defined as the art of acquiring new or modifying existing knowledge and skills.
- In short, we can define learning as a change in behaviour. Learning has many roles in artificial intelligence applications. For example, in robotics or expert system development, learning plays a separate module in the system.
- Machine learning is a computerized process which gives an improved performance in the application development with human computer interaction.

What is learning?

- Learning is an area of artificial intelligence that focuses on the processes of self improvement.
- Information processes improve their performance or enlarge their knowledge bases. Computers do the learning process through the following steps.

1. One way of learning is data acquisition of new knowledge. Any program that does such data acquisition is a learning program.

2. The problem solving is another component of learning.

Learning is hard because of the following reasons:

1. Intelligence implies that an organism or machine must be able to adapt to new situations.

2. It must be able to learn and must do new things.

3. This requires knowledge acquisition, inference, updating/refinement of knowledge base, acquisition of heuristics, applying faster searches, etc.

How computers learn?

- Many approaches have been attempted to pro vide a machine with learning capabilities. This is because learning tasks cover a wide range of phenomena. A few examples of how one may learn as follows:
- ***Skill refinement:*** One can learn by practice, e.g., playing the guitar.
- ***Knowledge acquisition:*** Learning through experience and storing it in a knowledge base is called knowledge acquisition. Rote learning can be cited as an example of this.

- ***Taking advice:*** It is similar to rote learning. But the input knowledge may need to trans form effectively by an order.
- ***Problem solving:*** If we solve a problem systematically, we will get experience to solve similar problems. For the next time, we can solve similar problems more efficiently.
- ***Induction:*** One can learn from examples. Humans often classify things in the world without knowing explicit rules. It usually involves a teacher or a trainer to aid the classification.
- ***Discovery:*** Here one learns knowledge without the aid of a teacher or a supervisor. New knowledge elements are inferred from the knowledge base obtained from a supervisor.
- ***Analogy:*** It may be able to transfer little knowledge for getting a better solution of the task in hand a system can use. This system finds and stores the similarities in the information.

7.1 TYPES OF LEARNING

- Learning is acquiring new or modifying new or modifying existing knowledge, behaviours, skills, values or preferences and may involve synthesizing different types of information. The ability to learn is possessed by humans , animals and some machines.

7.1.1 Rote Learning:

- The simplest kind of machine learning is recording of data. Data storage programs can be used for making the machines learn. Such a type of learning is known as rote learning. A few more capabilities for rote learning can be included in the following.

1. ***Organized storage:*** There is organized storage of information. In order to use the stored value, the board position should be organized in such a way that the process of retrieval is the fastest. This is done by indexing the board position by the number of pieces on the board.
2. ***Generalization:*** To keep the number of stored objects at a manageable level, some kind of generalization is necessary.
3. ***Direction:*** The program should try to find the required stored value intelligently for a given board position. So, we have to focus the attention to a single processing direction.

7.1.2 Learning by Parameter Adjustment:

- Game play determines the best moves by using static evaluation function or by MIN IMAX procedure. Pattern classification does this by assigning weights to the various features that a particular pattern contains in chess game.

- This is done by considering a polynomial of the form $C_1t + C_2t^2 + C_3t^3 + \ldots + C_{16}t^{16}$, where one can use 16 features and determine the 16 weights.
- The adjustment of $C_1, C_2 \ldots, C_{16}$ depending on the configuration is known as parameter adjustment. The value will be set with the best values obtained by experience.

7.1.3 Learning by General Problem Solving:

- Learning can be thought of as a form of problem solving methods. General problem solver (GPS) provides a design of a learning program.
- It is necessary to have a table that operates relevant to reducing the important differences between a given state and the goal state.
- Suppose we decide that rather than providing such a table to GPS, we want the program to teach the table itself from its own experience, then we say that GPS is another type of learning program.
- GPS was an attempt to construct a general problem solving mechanism that could solve problems in a new area given domain-specific knowledge about that area.
- GPS is a generalized state-space search mechanism. It has the following components:

1. A set of states and operators that change states. Specification of start and goal states.

2. A procedure for identifying differences between states.

3. A table of connections, which connects observed differences with opera tors that may be relevant for reducing those differences.

- GPS works by determining the difference between the current state and the goal state and selecting operators relevant to reducing that difference. Hopefully, this will give direction to the search and make it more efficient than a blind state-space search.

- To apply GPS, the problem of discovery of a good set of differences used by another copy of GPS working in a given problem domain, it is necessary to do the following:

1. Input a set of operators that will be available in the application domain.

2. Define a goal for the learning task. This goal is to provide a good set of differ ence for the application task.

3. Define a set of differences used by the learning task.

4. Define a set of operators for the learning task.

7.1.4 Concept Learning:

- Classification is the process of assigning to a particular input the name of the class to which it belongs. It is an important component of problem solving task.
- The idea of producing a classification programme that can evolve its own definition and construct these class definitions is called concept learning. The technique used depends on the way that classes are desired.

Suppose ***if:***

- The current goal is to go from place A to place B and there is a wall separating the two places

then:

- look for a doorway in the wall and go through it.
- To use this rule successfully, the systems matching outlived must be able to identify an object as a "wall". Without recognizing the object to be the "wall", the rule can never be invoked.

7.1.5 Learning by Analogy:

- Analogy involves a complicated mapping be tween what might appear to be two dissimilar concepts.
- Analogy a powerful inference rule. It allows similarities between the objects to be stated.
- An analogical learning system uses frame structure.

- For example, suppose we wish to say that "John is like a turbo-jet".
- This means that the person is 'John' who's movement or speech is very fast. Here, the turbo-jet is an object that maps to the person 'John'.
- Analogical problems are giving learning process which maps in two different ways. These two methods of analogical problem solving that have been studied in artificial intelligence are transformational analogy and derivational analogy.

➢ ***Transformational analogy:***

- Look for a similar solution and copy it to the new situation making suitable substitutions wherever appropriate (example, geometry). In 1983, Carbonell described this method.
- The idea is to transform a solution to a previous solved problem into a solution for the cur rent problem (Figure 7.1). Transformational analogy does not look at how the problem was solved; it only looks at the final solution.
- For example, we know how to solve a problem related to an isosceles triangle. The same methodology can be adapted to solve a problem related to an equilateral triangle.

Derivation analogy:

- The history of the problem solution, the steps involved, i often relevant. In 1986, Carbonell showed that derivation analogy is a necessary component in the transfer of skills in complex domains (Figure 7.2).
- For example, we know how to find the area of a triangle and a square. Derivation analogy will help to solve the volume of a pyramid from this knowledge.

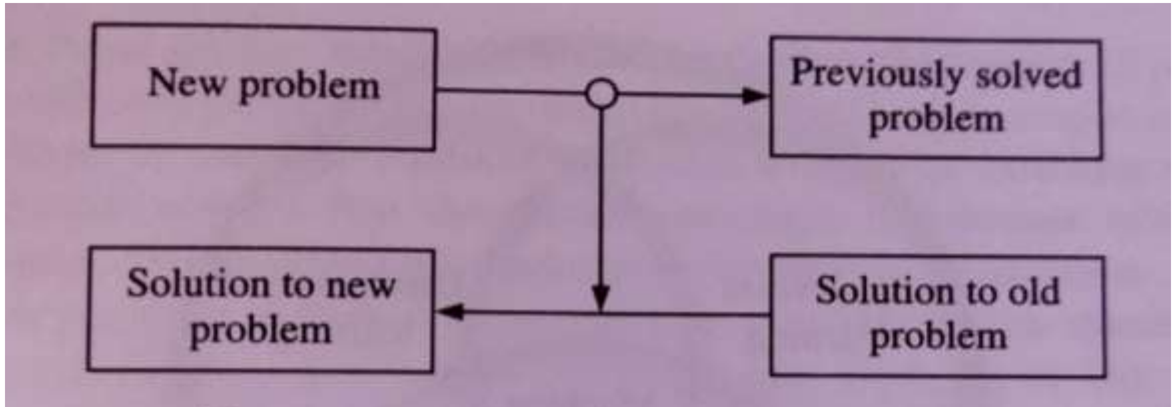


Figure 7.1 Transformational analogy.

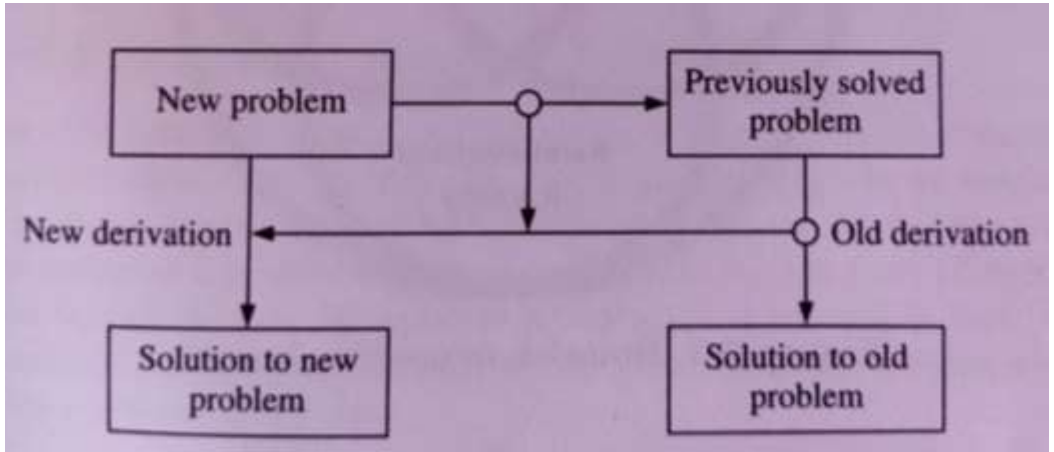


Figure 7.2 Derivational analogy.

7.2 MACHINE LEARNING:

- Machine learning is the branch of artificial intelligence concerned with the design and development of algorithms.
- It allows the computer to behave in a way based on the empirical data, such as from electronic sensors or databases.
- A major focus of machine learning research is to automatically train the system for recognizing complex patterns and to create intelligent decisions based on the provided inputs.
- The real set of possible data will be too large when compared to the inputs that we supply. Hence, it is always difficult to produce an effective output.

7.2.1 Why Machine Learning?

- Machine learning is the field of study that gives computers the ability to learn without being explicitly programmed.
- The main aim of machine learning is to develop systems: that are too difficult or expensive to construct manually. The systems must automatically adapt and customize themselves to individual users like personalizing news, mail filtering, etc.

- Computers are used in various fields of science and for various tasks. Normally, the programmers design and implement software that they need for particular functions. How ever, there are many tasks for which the implementation is difficult or impossible.

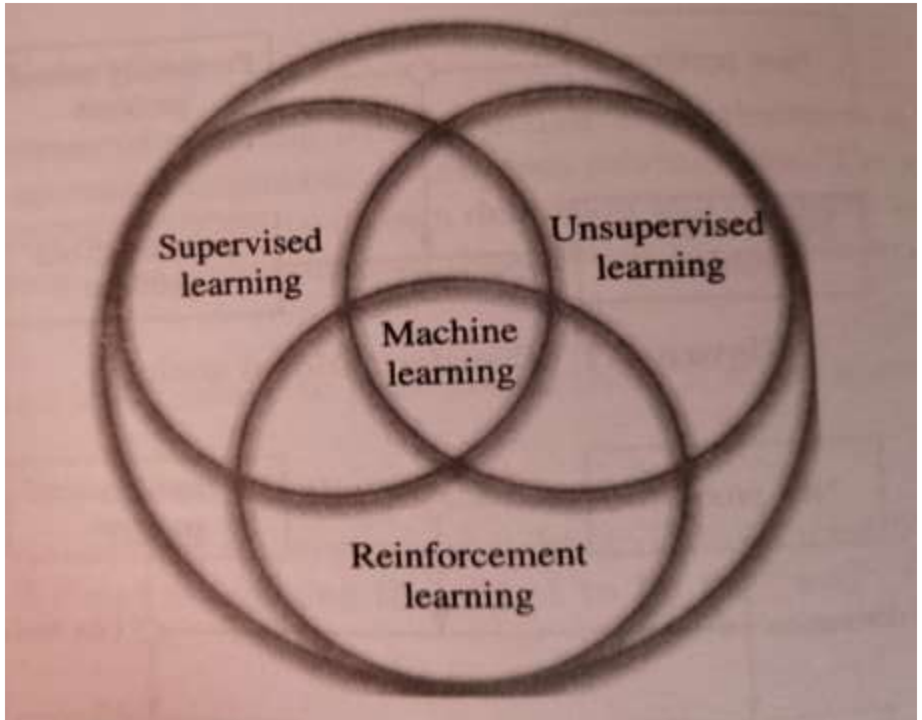


Figure 7.3 Branches of machine learning.

- Second, the problems where human experts exist, but they will be unable to explain their expertise. There are various tasks like speech recognition, hand-writing recognition, natural language understanding, etc.

7.2.2 Types of Problems in Machine Learning:

- Machine learning is all about algorithms and designing of algorithms for computers to train them to improve performance.
- It is not just the search of consciousness in machines but also comes close to the traditional goals defined by artificial intelligence.
- There are various machine learning algorithms avail able but selecting an apt algorithm for the kind of problem that we need to solve is the thumb rule in the process of learning.
- Learning is just the matter of finding statistical regularities or other patterns in the data. In a broad sense, we can classify the machine learning problems as follows:

1. Classification problems

2. Decision problems

3. Learning problems

- The types of learning algorithms fall into several classifications. One classification is the type of result expected from the algorithm.
- There are various hand written recognition software which give extremely good results. The neural network community works on the digit recognition problem. For the same problem, support vector machines offer a better solution classifier.

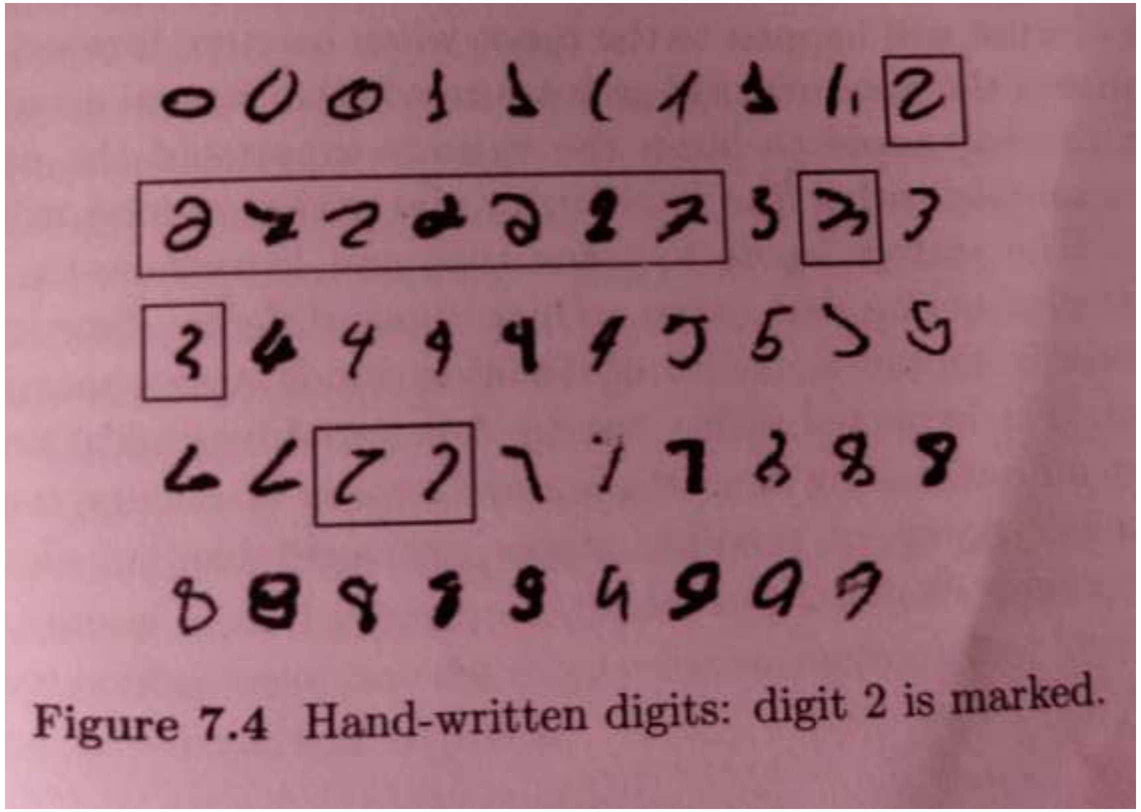


Figure 7.4 Hand-written digits: digit 2 is marked.

- We can have machine learning for prob lem solving too. It is mainly used in medical diagnoses and in computational biology.
- For instance, if the computer is trained to distinguish between cancer cells and normal tissues, then for a given a set of attributes, The system will be able to distinguish between likely cancer cells and normal tissues.

7.2.3 History of Machine Learning:

- 1950s can be termed as exploratory period during which many general techniques were born. It was during this time, Samuel Checkers' playing program, considered as the world's first self-learning program, was developed.
- It demonstrated the fundamental concept of artificial intelligence. Selfridge's pandemonium model proposed by Selfridge in 1959 also noted its success in modeling human pattern recognition.
- During 1960s the concept of neural net works was developed. Rosenblatt developed the perceptron, which was modeled after neurons. It was the precursor to later work in neural networks.
- In 1969, Minsky and Pa pert wrote a seminal paper proving that the perceptron was inherently limited, which discouraged research in this area for almost 20 years.

- 1970s were characterized by the development of more practical algorithms, often using symbolic techniques.
- Winston's important work on learning (Winston's arch learner) in blocks world domain happened in 1970. The idea of macro-operators was developed by Fikes during 1972.
- Mathematical discovery with AM by Lenat in 1977 and scientific dis covery with BACON by Langley in 1978 were the major discoveries.

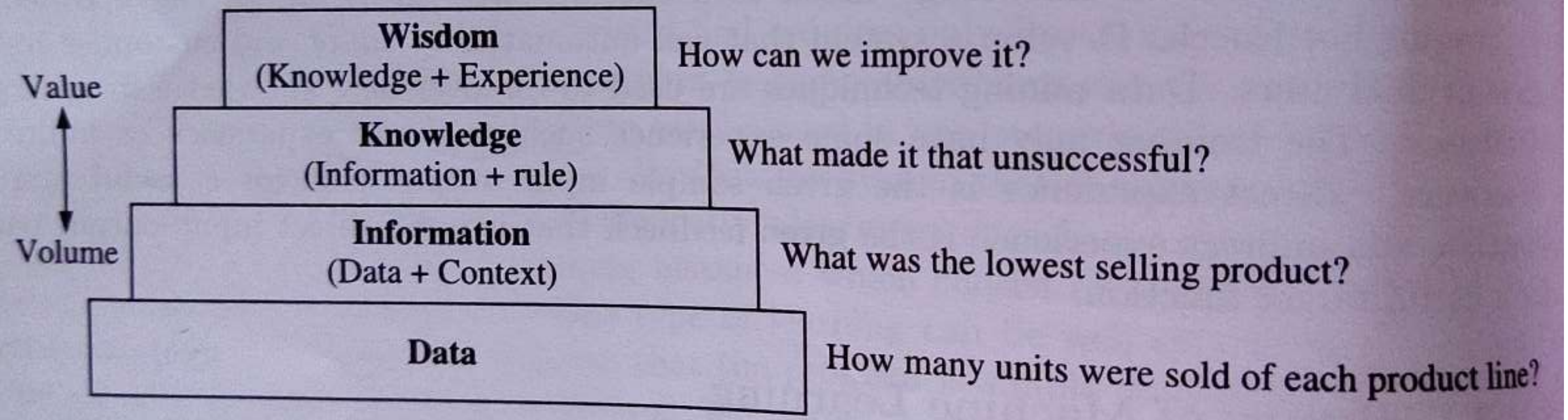


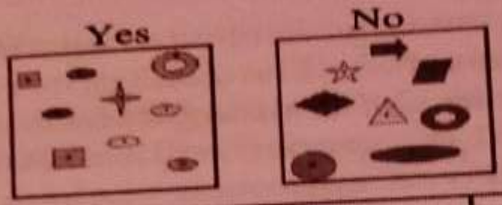
Figure 7.5 Data pyramid.

7.2.4 Aspects of Inputs to Training:

- The major classifications are supervised versus unsupervised training (class information on training example is available or not), incremental versus non-incremental training (training examples are given one time or con sequentially) and homogeneous versus heterogeneous training (described by only one type of data like numeric, symbolic, etc., or several types of data).

Supervised versus Unsupervised Training:

- As seen, supervised learning is employed when we have a sample data. So it is most commonly used in classification problems be cause the goal is to get the agent learn a classification system that is created.
- The digit recognition problem can be considered as an example of supervised learning. In the case of neural networks and decision trees, we have pre-determined classification and, hence, supervised learning is used commonly.
- In the case of neural networks, the classification is mainly used for determining the error of the network and to minimize it.



Color	Shape	Size	Output
Blue	Torus	Big	Y
Blue	Square	Small	Y
Blue	Star	Small	Y
Red	Arrow	Small	N

Figure 7.6 Positive and negative dataset.

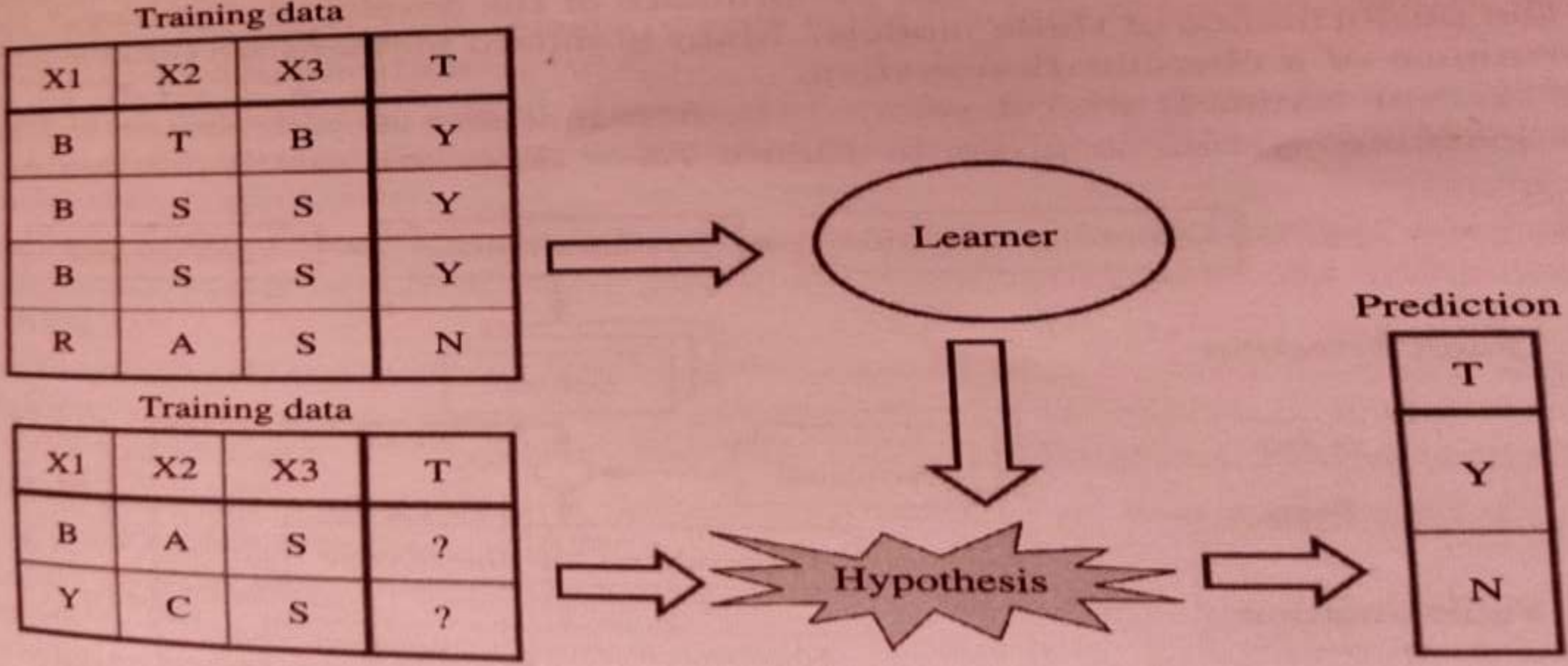


Figure 7.7 Supervised learning.

7.2.5 Learning Systems:

- For decision making, classification and prediction problems, we can use machine learning approaches. How can we evaluate the performance of the developed algorithms? How will we quantify the performance of these models? Many standard methods are suitable for measuring the performance of a classification system.
- It was Arthur Samuels who developed the first machine learning system at IBM in 1959. A typical learning system is given in Figure 7.8. There are two approaches to develop a learning system:

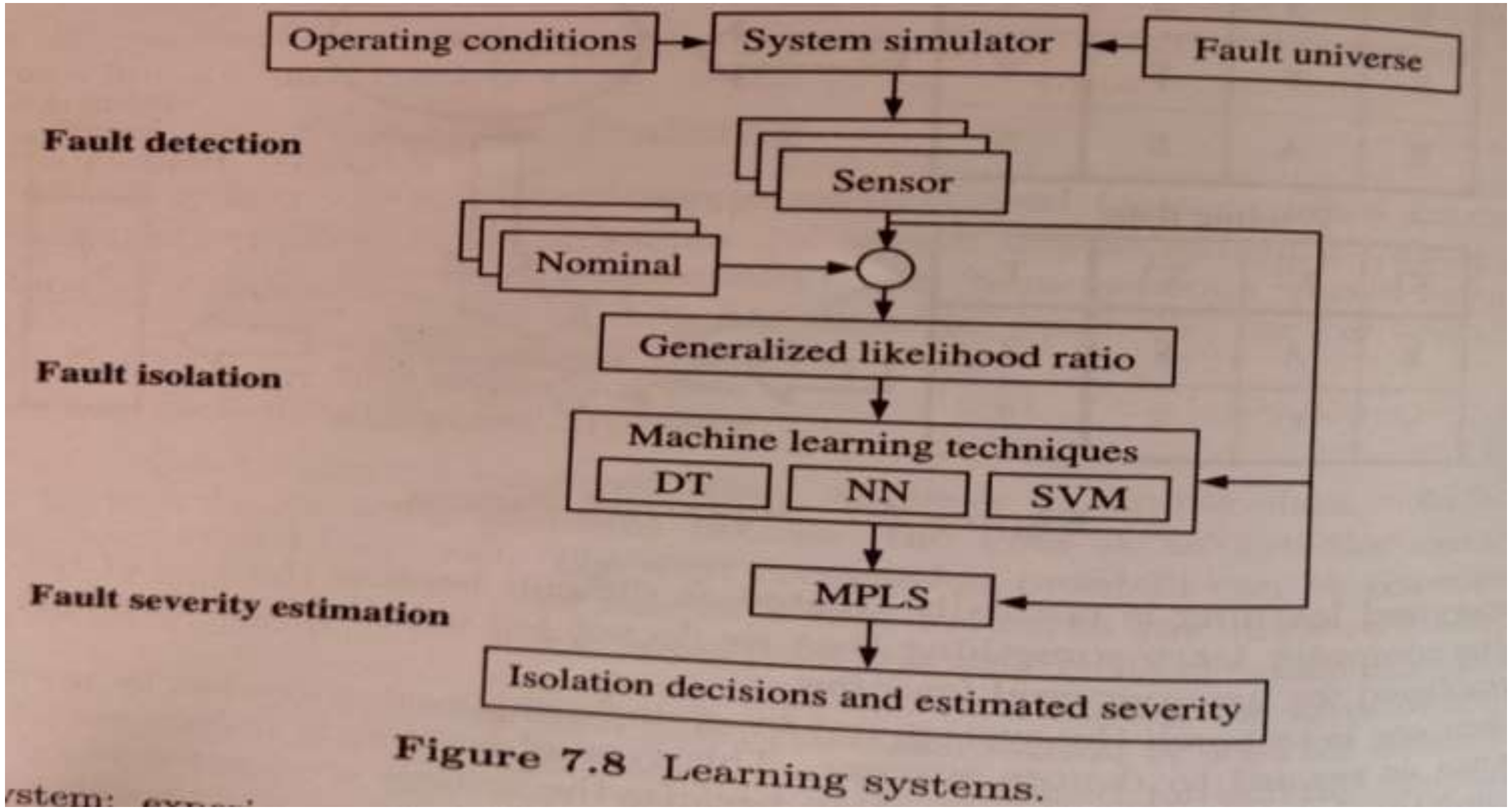


Figure 7.8 Learning systems.

- Experimental and theoretical. Experimental approach con ducts controlled cross-validation experiments to compare various methods on a variety of benchmark datasets and gather data on their performance.
- For example, test accuracy, training-time and testing-time. This method analyses differences for statistical significance in the results. Algorithms are analyzed mathematically using theoretical approaches.
- Prove theorems in terms of computational complex it and calculate the ability to fit the training data and sample complexity.
- One of the major challenge is that it is highly dependent on quantity and quality of system operational data.

7.2.6 Machine Learning Applications:

Major applications of machine learning can be summarized as follows:

1. ***Optical character recognition:*** Categorizing the images of handwritten characters by the letters represented in a text. ANN and HMM are widely used for optical character recognition.

2. ***Face detection:*** Finding faces in images or indicating if a face is present. HMM is suitable for developing such application.

3. ***Spam filtering:*** Identifying e-mail messages as spam or non-spam from a bunch of e- mails. Any of the machine learning technique such as SVM, ANN or HMM can be used to develop such applications.

4. ***Topic spotting:*** In order to categorize news articles (say) as to whether they are about politics, sports, entertainment, etc., HMM or SVM can make an influence in this application.

5. ***Spoken language understanding:*** Within the context of a limited do main, determining the meaning of something uttered by a speaker to the extent that it can be classified into one of the fixed set of categories. For language classification, HMM is highly useful.

6. ***Medical diagnosis:*** For diagnosing a patient as a sufferer or non-sufferer of some disease, any rule-based system can be useful.

7. ***Customer segmentation:*** Predicting which customers will respond to a par ticular promotion in a domain task. This prediction problem can be in co-oporated with any of the prediction technique like SVM or ANN.

8. ***Fraud detection:*** For identifying a credit card transaction and identifying whether a fraud happens. Using this prediction system, any supervised ma chine learning can be applied to make a solution.

9. ***Weather prediction***: To predict whether or not it will rain tomorrow is a type of application that can be develop in HMM.

7.2.7 Quantification of Classification:

- A classifier results a set of real output values between a threshold value. Consider two classes of prediction problem. (binary classification).
- The outcome is labelled into positive (p) or negative (n) class. For the classification model, the outcome is labelled as actual class and predicted class {Y, N).
- For a binary classifier, there are four possible outcomes. If the outcome and the actual value are positive, then it is classified as true positive (TP), but the actual value is counted as negative, then it is classified as false positive (FP).
- Similarly, if the outcome and actual values are negative,

then it is counted as true negative (TN),

but if the actual value is positive,

then it is counted as false negative (FN).

Figure 7.9 shows the two-by-two contingency table representing the dispositions of the set of instances.

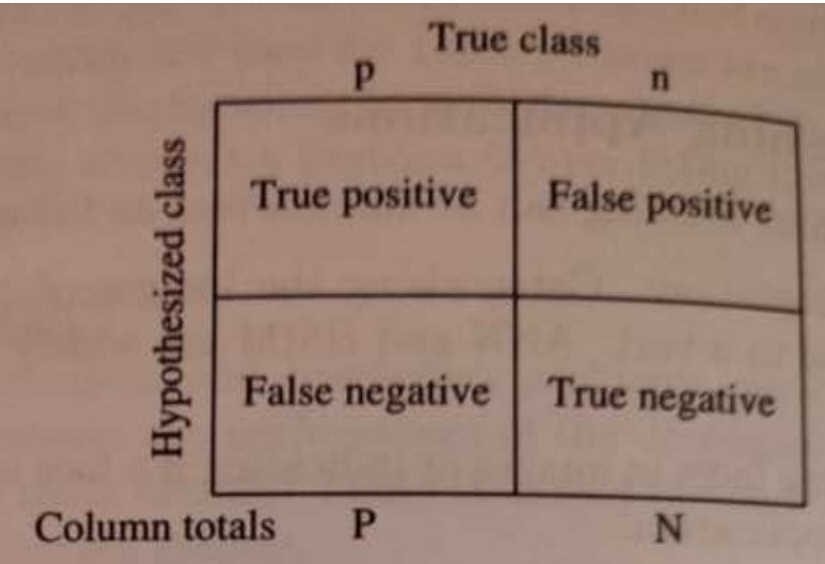


Figure 7.9 Confusion matrix.

The hit rate or true positive rate (TPR) of a classifier is estimated as

$$\text{TPR} \approx \frac{\text{Correctly classified positives}}{\text{Total positives}} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

True positive rate is also called sensitivity (Se).
The false positive rate (FPR) of the classifier is

$$\text{FPR} \approx \frac{\text{Incorrectly classified negatives}}{\text{Total negatives}} = \frac{\text{FP}}{\text{TN} + \text{FP}}$$

False positive rate is also called 1 – Specificity (Sp), i.e.,

$$\text{Specificity (Sp)} = \frac{\text{TN}}{\text{TN} + \text{FP}}$$

The accuracy can be estimated as

$$\text{Accuracy (Ac)} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}$$

7.3 INTELLIGENT AGENTS

- Machine learning processes produce numerous models in the form of software. These models are very useful for various applications. For example, a spam filter in the e-mail server acts as an intelligent software component.
- We call this system an intelligent agent. Basically, these agents are software components incorporated within an application.
- Anything which can perceive its environ ment through sensors and react upon the environment through actuators can be termed as an agent (Figure 7.11). A human agent is eyes, ears and any other organs. For sensors, hands, legs, mouth and other body parts for actuators.
- A robotic agent is any one of cam eras and infra-red range finders for sensors and various motors for actuators.
- Intelligent agents are described as an abstract functional system similar to a computer program. The scientist trio, Smith, Cypher and Spohrer, in the year 1994, described an agent as a per sistent software entity dedicated to a specific purpose.

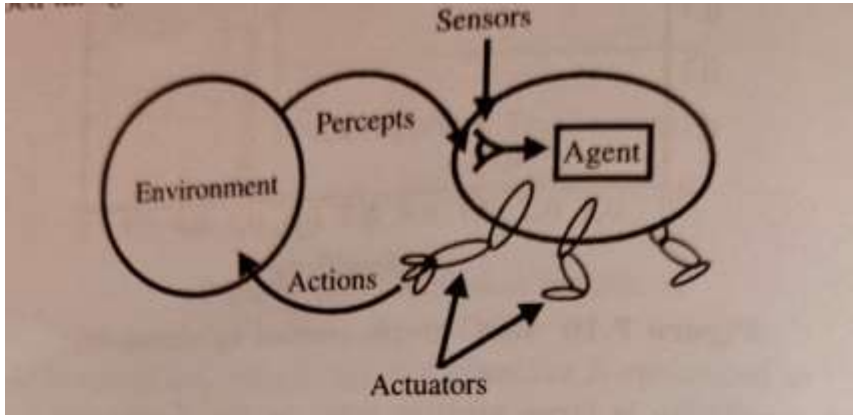


Figure 7.11 An agent.

- The duration of the agent’s life does not depend wholly on a user launching and quitting it. The autonomy of an agent does not need user’s input to function. Agents are di-vided into five classes based on their degree of perceived intelligence and capability (Figure 7.12).

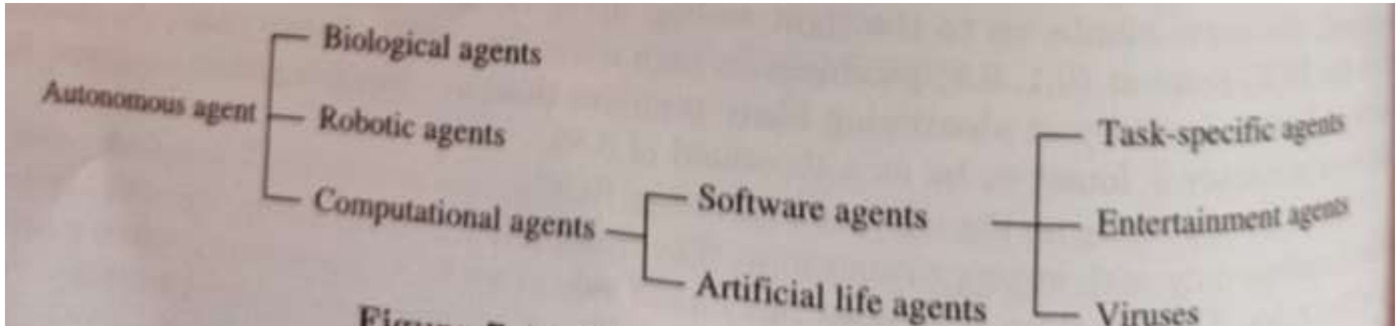


Figure 7.12 Classification of agents.

1. ***Simple reflex agents:***

- The decisions are taken based on the current perception and not on the history of perceptions. This can be basically termed as a condition action rule, that is, if <condition> then <action>.
- There is a lookup table of actions from which actions are taken for every possible state of environment. If the environment has 'n' variables. Each with 't' possible states, then the table size is 't.
- This will work only for a small number of possible states for the environment and works if the environment is fully observable. The simple reflex agent is shown in Figure 7.13.

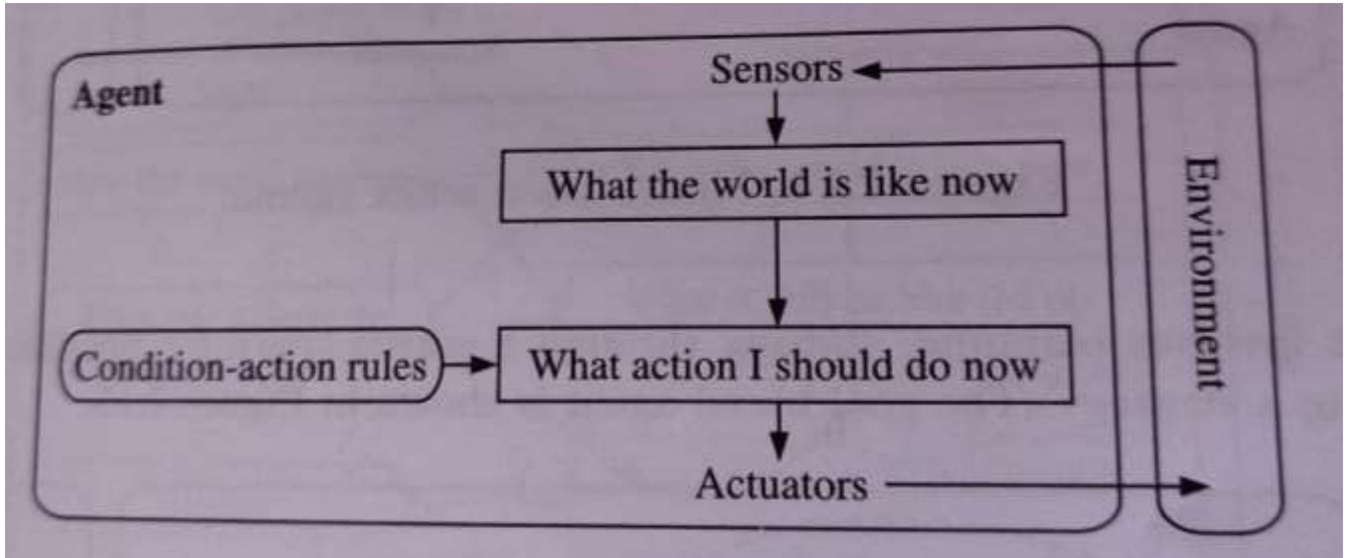


Figure 7.13 Simple reflex agents.

2. ***Model-based reflex agents:***

- If the world is not fully observable, the agent must remember observations on the parts of the environment which it cannot currently see. This need to have an internal representation of the world or internal state. Since this representation is a model of the world, we call it a model-based agent. The model-based agent is shown in Figure 7.14

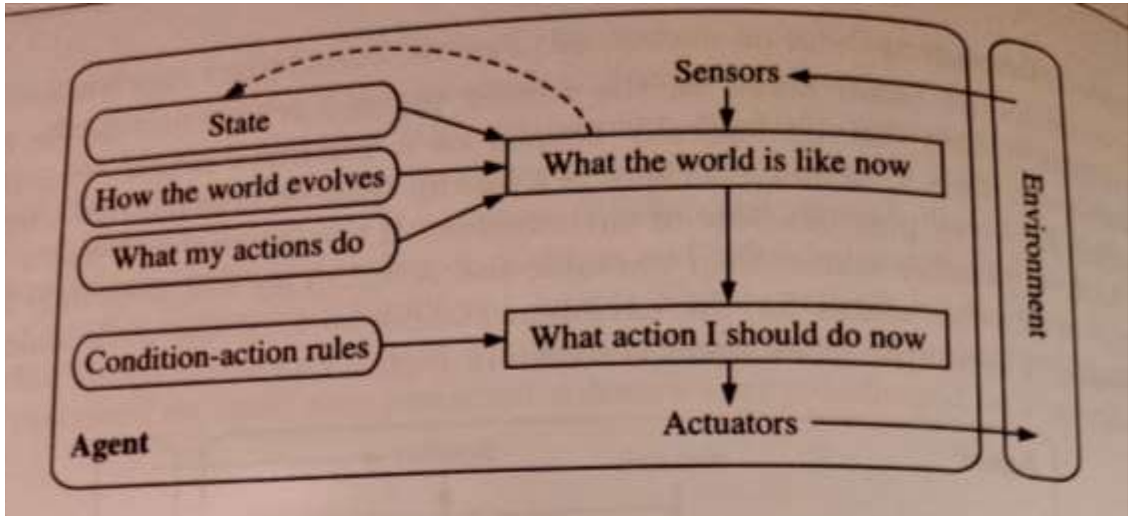


Figure 7.14 Model-based reflex agents.

- The most effective way to handle partial observability is to keep track of the part of the world it cannot see now.
- The knowledge about "how the world works" is called a model of the world. An agent that uses such a model is called a model-based agent.

1. ***Goal-based agents:***

- The agent has a purpose and the action to be taken depends on the current state and on what it tries to accomplish (the goal). In some cases, the goal is easy to achieve. In Others, it involves planning, shifting through a search space for possible solutions and developing a strategy. The goal-based agent is shown in Figure 7.15.

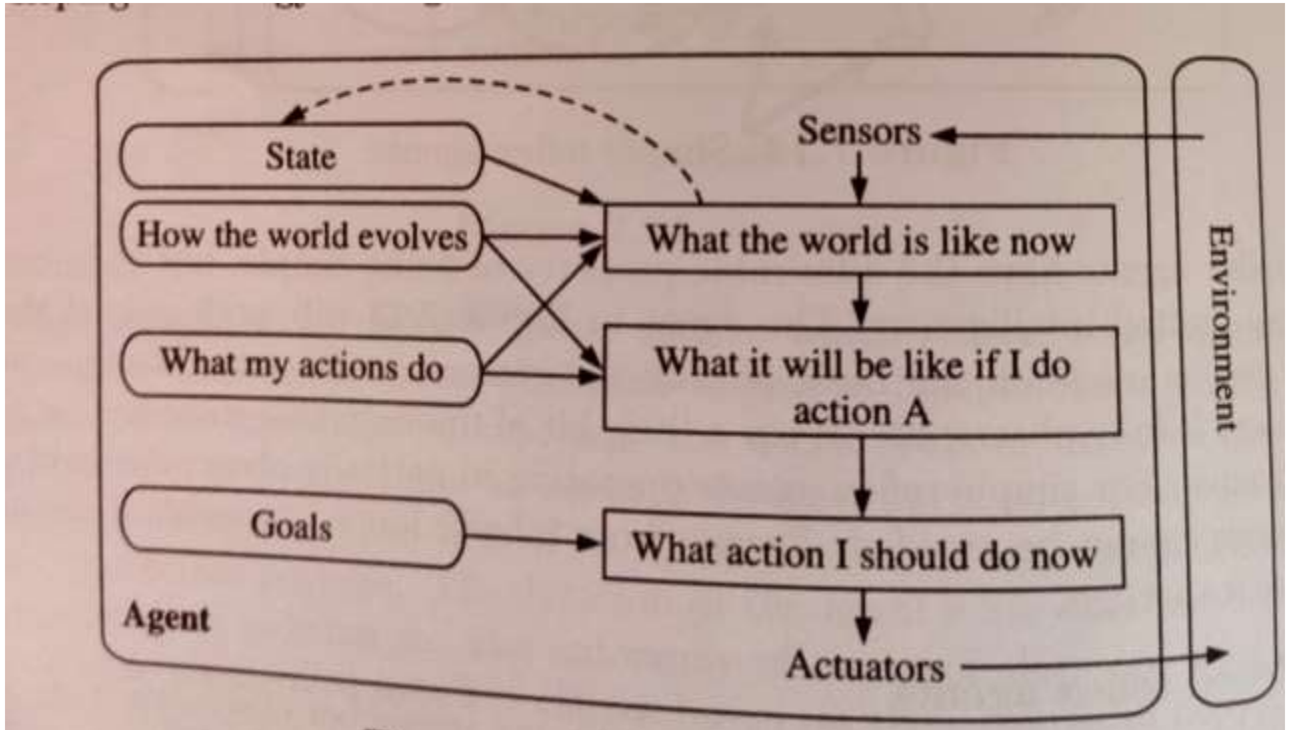


Figure 7.15 Goal-based agents.

- The agent sometimes needs some sort of goal information that describes situations that are desirable.
- The agent program can combine this goal with information about the results of possible actions.

4. *Utility-based agents:*

- The agent is aware of a utility function that estimates how close the current state is to the agent's goal. If there is more than one way to reach the goal, then we need to choose the best way of doing it.
- A utility function maps a state (or a sequence of states) onto a real number, which describes the associated degree of happiness.
- A complete Specification of the utility function allows rational decisions in two kinds of cases where goals are inadequate.
- which the likelihood of success can be weighed up

against the importance of the goals.

The utility-based agent is shown in Figure 7.16.

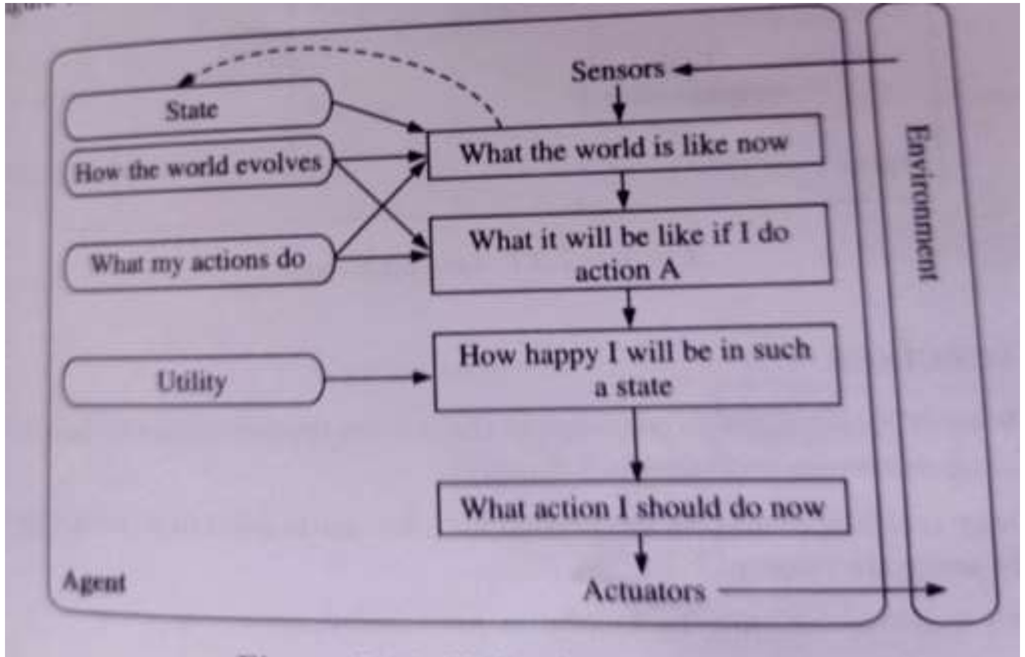


Figure 7.16 Utility-based agents.

5. ***Learning agents:***

- These agents are capable of acquiring new competence through observations and actions. The basic components are learning element (modifies the performance element), performance element (selects actions), feedback element (critic) and exploration element (problem generator).
- Learning agents operate initially in an unknown environment and become more competent than their initial knowledge. A typical learning agent is shown in Figure 7.17.
- They do not require to be hand programmed. The major parts of a learning agent are as follows:

(a) What we have considered previously for an agent is the performance element. It Will take in precepts and decide on actions to the agent.

(b) A feedback from the critic is taken by the learning element. How will the agent determine the performance element and modify this in future? This is also take into consideration.

(c) How well the agent is performing with respect to a fixed performance standard? This is critic question regarding the learning element.

(d) A new and informative experience will lead to a problem generator. This will be responsible for suggesting actions in informative experience.

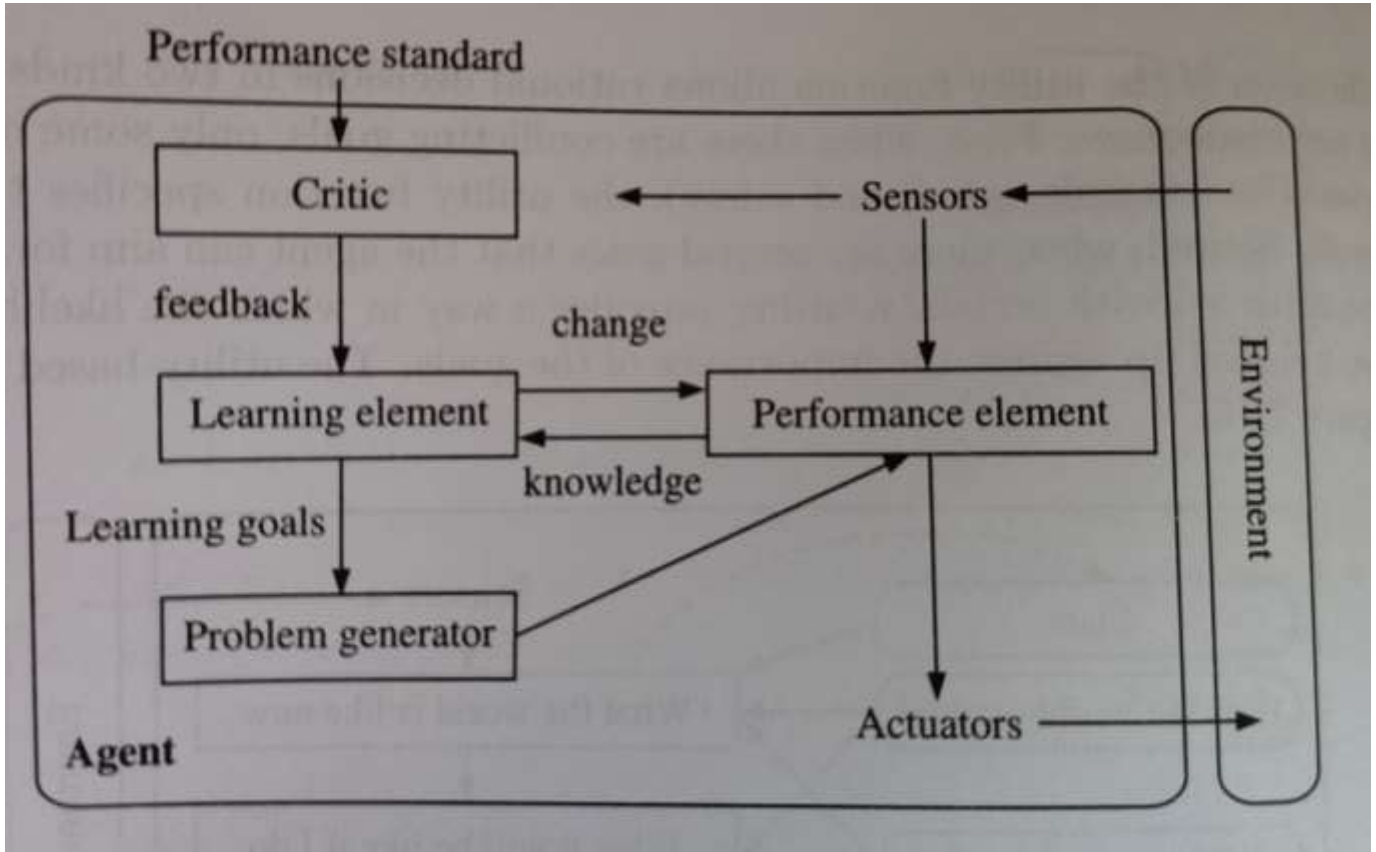


Figure 7.17 Learning agents.

THANK YOU