Q.2 a. Define Artificial Intelligence. What are the objectives of AI research? (8)

Answer:

Artificial Intelligence can be defined as:

- AI is the study of how to make computers do things at which, people are better at the moment.
- AI is that branch of computer science dealing with symbolic, non-algorithmic methods of problem solving.
- AI is that part of computer science concerned with designing intelligent computer systems that exhibit the characteristics used to associate with intelligence in human behaviour. (2Marks)

The main objectives of AI research are:

- *Understand human cognition,* i.e. how do human solve problems? Try to obtain deep knowledge of human memory, problem-solving abilities, learning, decision making, etc.
- *Cost-effective automation* replaces humans in intelligent tasks. Have programs which perform as well as humans currently doing the job?
- *Cost-effective intelligent amplification* builds systems to help humans think better, faster, deeper ... For example system to help GP diagnose disease.
- *Superhuman intelligence* builds programs to exceed human intelligence.
- *General problem solving* solves broad range of problems. Systems with breadth of mind.
- *Coherent discourse* communicates with people using natural language. Carries out an intelligent dialogue.
- *Autonomy* has intelligent systems acting on own initiative. Must react to the real world.
- *Learning* (induction) the system should be able to gather own data and how. Generalize, hypothesize, apply/learn heuristics; reason by analogy.
- *Store information* and know how to retrieve it. (6 Marks)

b. Briefly describe the following applications area of Artificial Intelligence:(2×4)

(i) Intelligent retrieval from Databases

(ii)Combinatorial and scheduling Problems

Answer:

(i) Intelligent retrieval from Databases

Database systems are large bodies of facts about some subjects, which are used to answer users' queries that subject. For example, consider a player database. For each player listed in the database, there could be entities representing his age, height, weight, number of caps, total earnings, etc. With these facts stored in an orderly form, one can obtain answers to queries like 'List all players who are over 6 feet tall." or "What is the average yearly salary of player X?", etc. The design of database systems is a very active field in the area of computer science even today. From the point of view of AI, the field becomes even more interesting if the answers require deductive reasoning with the facts in the database.

There are numerous problems that confront an individual trying to build such an intelligent retrieval system. First, the problem of understanding queries stated in a natural language like English. Second, assuming that the system has fully understood the query, how should it deduce the answer from the facts in the database? Third, for the given facts there are some queries which require common knowledge. For example, if player P is the captain of a side, then the system should be in a position to know that P leads the team

always on the field and he is part of the core group which has chosen the team. Such facts may not be explicitly stated during the development of the database. (4 Marks)

(ii) Combinatorial and scheduling Problems

One class of problems is concerned with specifying optimal scheduled. A classical example is the *Travelling Salesperson Problem*, where the problem is to find a minimum distance tour, starting at one of the several cities, visiting other cities only once and then returning back to the starting city. The problem generalizes to finding a minimum cost path over the edges of a graph containing n nodes such that each of the n nodes is visited not more than once. In such problems the domain of possible combinations or sequences from which to choose an answer is very large. Brute force attempts to generate solution very often leads to a *combinatorial explosion* of possibilities that exhaust the resources of even large computers. Such problems are called as NP-complete problems by the computational.

AI researchers have worked on methods for solving various types of combinatorial problems. The key to solving such problems is the knowledge about the problem domain. Methods developed to solve combinatorial problems have proven to be helpful in solving other less combinatorial severe problems. (4Marks)

Q.3 a. Let G and H be two formulas. Consider the truth values of the formulas generated over G and H are given as:

- **~G is *true* when G is *false*, and is *false* when G is *true*. G is called the *negation* of G.**
- **(G ∧ H) is *true* if G and H are both *true*; otherwise, (G ∧ H) is *false*. (G ∧ H) is called the *conjunction* of G and H.**
- **(G ∨ H) is *true* if at least one of G and H is *true*; otherwise, (G ∨ H) is *false*. (G ∨ H) is called the *disjunction* of G and H.**
- **(G → H) is *false* if G is *true* and H is *false*; otherwise, (G → H) is *true*. (G → H) is read as "If G, then H", or "G implies H".**
- **(G ↔ H) is *true* whenever G and H have the same truth values; otherwise, (G ↔ H) is *false*.**

Represent the above relations by means of truth-table. (4)

Answer:

The above given relation can be represented by means of truth table as follows:

G	H	~G	(G ∧ H)	(G ∨ H)	(G → H)	(G ↔ H)
T	T	F	T	T	T	T
T	F	F	F	T	F	F
F	T	T	F	T	T	F
F	F	T	F	F	T	T

(4 Marks)

b. List and describe the properties of well-formed formulas (WFFs)? (6)

Answer:

Properties of WFFs

- **Interpretation:** When assignment of values is given to each predicate symbol in a wff, we say interpretation is given to the wff. If value of interpretation is *true*, then it is called the *model of the wff*. An interpretation of a formula in predicate logic consists of a non-

empty domain D, and assignment of values to each constant, function symbol, and predicate symbol in it as follows:

- To each constant / variable we assign an element in D.
- To each n-ary function symbol, we assign mapping from D" to D.
- To each n-ary predicate symbol, we assign mapping from D" to (T, F) e.g. consider the wff (∀x) (∃y) P(x, y)
  We can define the interpretation over the domain D = {1, 2} as follows:
  P(1, 1) = true, P(1,2) = false, P(2, 1) = false, P(2, 2) = true.

- **Validity:** A wff is said to be valid, if it is *true* under every interpretation, otherwise it is called invalid wff. Valid ground wffs are known as tautology.
- **Satisfiability:** A wff which is *true* for some interpretation is satisfiable and wff which is *false* under every interpreatation is said to be unsatisfiable (inconsistent).
  Valid wff is satiafiable, and inconsistent wffs are invalid.
- **Logical consequence:** In Mathematics as well as in daily life, we often have to decide whether one statement follows from the other statement. It leads to the concept of logical consequences. We can define it as follows:
  Given formulas W1, W2, ..., Wn and a formula G, which is a logical consequence of W1, W2, ..., Wn (or logically follows from W1, W2, ..., Wn) if and only if for any interpretation 1 in which W1 ∧ W2 ∧ .....∧ Wn is *true*, G is also *true.* W1, W2, ..., Wn are called *axioms* or *postulates* of G.
- **Equality:** If the truth values of the two wffs are same regardless of the interpretations, we say these wffs are equivalent. (6Marks)

c. Write some strategies that are helpful in building rule-based expert systems. (6)

Answer:

Some strategies that are helpful in building rule-based expert systems are given below:

- Once a rule fires, eliminate it so that it does not fire again (the right-hand side is already true).
- Schedule the rules that are more general earlier than those that are more specific (A && B && C → Q is more general than A && B && C && D → Q).
- Put more important rules higher in the ordering of rules, if known.
- Decompose the rule base into distinct sets when possible, or break into two or more sets that join together farther to the right (we start on the left with the first antecedents).
- Validation of the system can be done by giving the system and an expert the same set of test problems to solve, then comparing the results. (6Marks)

Q.4 a. With the help of a diagram, explain the procedure for knowledge acquisition. (6)

Answer:

3.4 PROCEDURE FOR KNOWLEDGE ACQUISITION

Initially the knowledge domain (Figure 3.2) must be clearly identified to ensure the development of a tightly focussed system. This is achieved by a careful examination

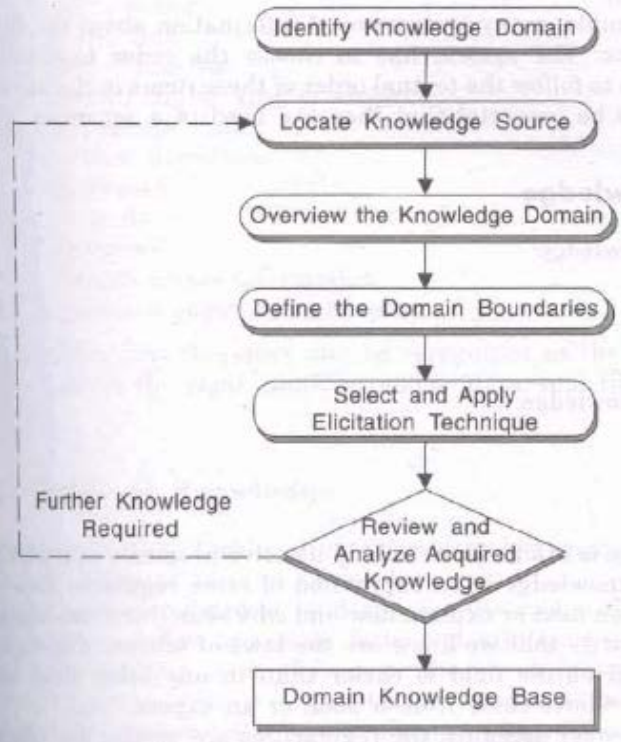


Figure 3.2 Procedure for knowledge acquisition.

of the proposed system goals. Sources of relevant domain knowledge could then be located for subsequent analysis and extraction. Prior to the definition of the domain boundaries it is advisable to overview the subject matter. This provides an insight and structure for subsequent work and highlights any potential problems. The definition of the domain boundaries ensures that during the elicitation process there are no major omissions or wasted effort gathering unwanted information.The selection of a suitable elicitation technique follows with the chosen method applied to abstract the required information. A variety of techniques are available.

Once knowledge is identified and acquired, the final task in the process is to ensure that it is sound, reliable and concentrated into a suitable format. The wealth of knowledge that contributes to the domain under investigation must be comprehensible and unambiguous.

Elicitation delivers single items of knowledge that need to be organized into a unified whole. How much organizing is needed depends very much on the way in which the knowledge base is implemented and used. Some implementations require items to be arranged in groups and sequences, where the arrangement displays relationships between items in the knowledge base. Their order could be used, for example, to determine the sequence in which various items are used—an important decision that the system has to make if several items could be validly used at a certain point of consultation.

To quote a concrete example, many systems need information about the client or patient: name, age, sex, etc. The system has to choose the order to obtain this information. A simple way is to follow the textual order of these items in the knowledge base. In other cases it may be essential that they are used in a set order.

b. Discuss Semantic Networks. Draw the semantic network intended to represent the following data:

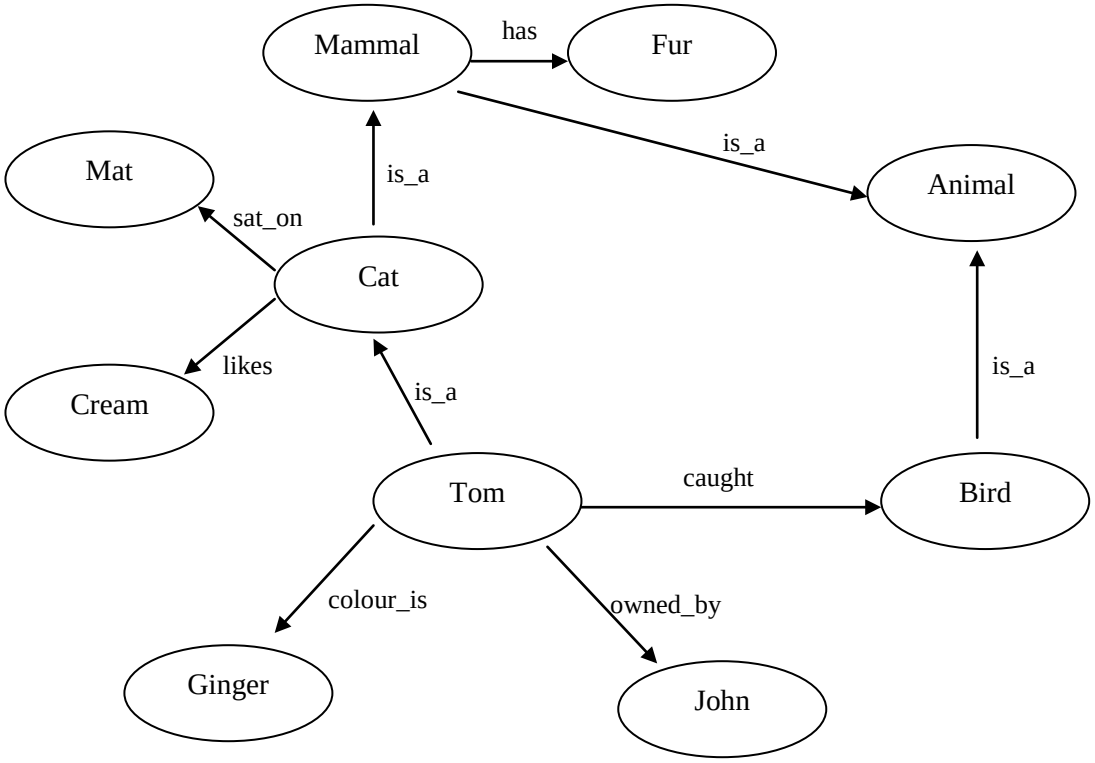
- **Tom is a cat.**
- **Tom caught a bird.**
- **Tom is owned by John.**
- **Tom is ginger in colour.**
- **Cats like cream.**
- **The cat sat on the mat.**
- **A cat is a mammal.**
- **A bird is an animal.**
- **All mammals are animals.**
- **Mammals have fur.** **(10)**

Answer:

Semantic networks are alternative to predicate logic as a form of knowledge representation. The idea is that we can store our knowledge in the form of a graph, with nodes representing objects in the world, and arcs representing relationships between those objects.

Graphs, by providing means of explicit representation of relations using arcs and nodes, have proved to be ideal vehicle for formalizing assocoationist theory of knowledge. A semantic network represents knowledge as a graph, with the nodes corresponding to facts or concepts and arcs to relations or associations between concepts. Some of the principles of semantic networks are as follows:

- Semantic nets describe relationships between things that are represented as nodes.
- The nodes are circle that have names.
- The relationships between nodes are represented by arcs that connect the circles.
- A semantic net can be used to generate structures and objects.
- A semantic net can be used to generate rules for a knowledge base.



Semantic network representing the given facts (10 Marks)

Q.5 a. What are the main features of Knowledge Representation Languages? (4)

Answer:

The main features of Knowledge Representation Languages are as follows:

1. **Object-orientedness:** All the information about a specific concept is stored with that concept, as opposed, to rule-based systems where information about one concept may be scattered throughout the rule base.
2. **Generalization/Specialization:** Long recognized as a key aspect of human cognition, KR languages provide a natural way to group concepts in hierarchies in which higher level concepts represent more general, shared attributes of the concepts below.
3. **Reasoning:** The ability to state in a formal way that the existence of some piece of knowledge implies the existence of some other, previously unknown piece of knowledge is important to KR. Each KR language provides a different approach to reasoning.
4. **Classification:** Given an abstract description of a concept, most KR languages provide the ability to determine if a concept fits that description, this is actually a common special form of reasoning. (1Marks each)

b. Explain the following hybrid representation systems: (2×4)
(i) COLAB
(ii) YAK

Answer:

i. COLAB

Compilation Laboratory (COLAB) is a hybrid knowledge system emphasizing the horizontal and vertical compilation of knowledge bases. It has been designed as a Compilation Laboratory aiming at a synergetic collaboration of different representation and reasoning formalisms. COLAB is a piece of research software developed at DFKI and the University of Kaiserslautern. It is comprised of subsystems dealing with different kinds of knowledge and that can also be used as stand-alone systems. The COLAB representation architecture splits into two main parts, an affirmative part, sometimes also called 'assertional', and a taxonomic part. (2Marks)

The affirmative part provides efficient reasoning with different kinds of relational or functional knowledge using tailored inference engines. For affirmative knowledge represented as constraint nets COLAB supplies constraint propagation techniques (CONTAX). Relational knowledge in the form of Horn rules is processed by forward (FORWARD) and backward (RELFUN) chaining. The backward component is also suited for expressing (non-deterministic) functional dependencies. Taxonomic knowledge is represented by intentional concept definitions, which are automatically arranged in a subsumtion hierarchy (TAXON). Dynamic cooperation of the subsystems is organized through access primitives providing an interface to the respective reasoning services. (2Marks)

ii. YAK

YAK is a hybrid KR system, and in its foundation is similar to CLASSIC and Loom. The core of the system is a traditional TBox/ABox hybrid representation language, enhanced possible in a "principled" fashion with other hybrid modules representing different kind of knowledge and reasoning. The system, fully implemented in CommonLISP is the main knaoledge representation module of the large NL architecture (ALFresco) natural

language system, a multimodel dialogue prototype for the exploration of Italian art history. (2Marks)

The expressivity of the YAK Tbox comes out from a study about the balancing of expressiveness, functional adequacy and formal prototypes of deductive procedures.

The classifier provided within the YAK system has a tractable, sound and complete algorithm. Moreover the classifier is speeded up by a caching mechanism and an intelligent name expression scheduling, borrowed from the tabular approach used in non-deterministic natural language parsers.

ABox is with contexts, viewpoints and belief representation, reasoning about sets and time, compositional query language with typed variables. (2Marks)

c. What are the basic ideas behind Bayesian Network? How it is implemented? (4)

Answer:

These are also called belief networks or probabilistic inference networks. The basic idea is as follows:

- Knowledge in the world is modular – most events are independent of most other events.
- Adopt a model that can use a more local representation to allow interactions between events that only affect each other.
- Some events may only be unidirectional, others may be bidirectional – make a distinction between these in model.
- Events may be causal and thus get chained together in a network.

Implementation

- A Bayesian network is a directed acyclic graph.
- A graph where the directives are links indicating dependencies that exist between nodes.
- Nodes represent propositions about events or events themselves.
- Conditional probabilities quantify the strength of dependencies. (4 Marks)

Q.6 a. Explain the basic procedure for solving a problem in AI? (5)

Answer:

The basic procedure for solving a problem in AI can be written as:

procedure PRODUCTION

1. DATA ← initial database
2. while (DATA does not satisfy the termination condition) do
   (a) select some rule, R, in the set of rules that can be applied to DATA
3. DATA ← result of applying R to DATA

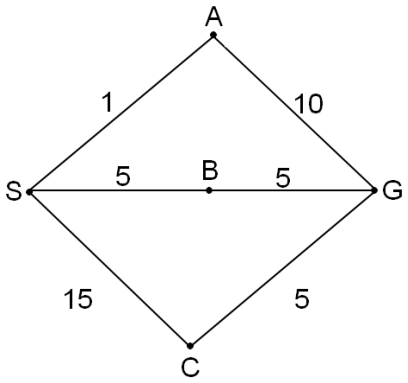
end PRODUCTION

In the above procedure, the fundamental control problem is selecting the rule R to apply to the database. An important characteristic in this regard is the amount of "knowledge" of information at hand that these computations use. Such a search procedure uses the knowledge at hand as an informed/heuristic search procedure against an uninformed/blind search procedure where the selection is made arbitrarily.

The overall computational efficiency of a search depends on wherein the uninformed/informed spectrum the control strategy falls. This efficiency is a measure of two types of costs: rules application costs and control costs. A completely uninformed control strategy incurs only a small control computation. However, such a strategy results in high rule application costs as these

require large number of rule applications to find a solution. To inform a control strategy about the problem domains of interest usually involves high cost computations. These strategies, however, result in minimum rule application costs since they guide the search systems directly to a solution. Part of the art of designing efficient control strategies in AI systems is to decide to balance these two costs. (5Marks)

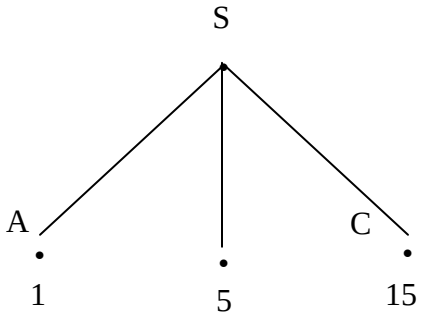
b. Using Uniform Cost Search, find the shortest route from S to G of the following graph. Explain each step. (6)



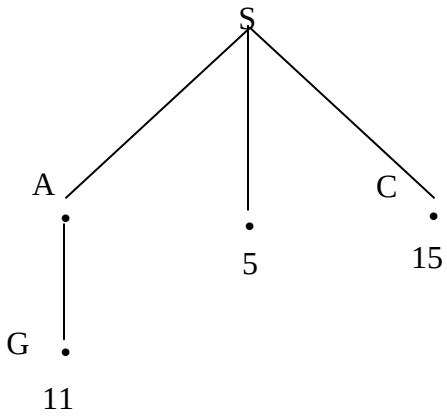
Answer:

We want to find the shortest route from S to G, that is S is the initial state and G is the goal state. We can see that SBG is the shortest route but if we let breadth-first search loose on the problem it will find the path SAG, assuming that A is the first node to be expanded at level 1. But this is how uniform cost search tackles the problem.

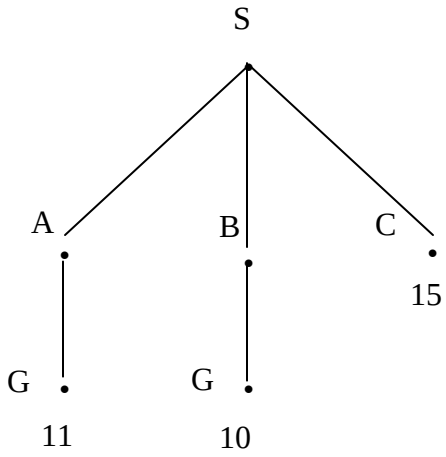
We start with the initial state and expand it. This leads to the following tree:



The path cost of A is the cheapest, so it is expanded next, giving the following tree:



We now have a goal state but the algorithm does not recognize it yet as there is still a node with a cheaper path cost. In fact, what the algorithm does is order the queue by the path cost so the node (3Marks)
will cost 11will be behind node B in the queue. Node B (being the cheapest) is now expanded, giving the following tree.



A goal state (G) will now be at the front of the queue. Therefore the search will end and the path SBG will be returned. (3Marks)

c. Write the procedure for BRANCH-AND-BOUND with UNDERESTIMATES. (5)

Answer:

Procedure BRANCH-AND-BOUNT with UNDERESTIMATES

1. Form a queue of partial paths. Let the initial queue consist of zero-length, zero-step path from the root node to nowhere.
2. Until the queue is empty or the goal has been reached, determine if the first path in the queue reaches the goal node
   (a) If the first path reaches the goal, node, do nothing.
   (b) If the first path does not reach the goal node
      (i) Remove the first path from the queue.
      (ii) Form new paths from the removed path by extending one step.
      (iii) Add the new path to the queue.
      (iv) Sort the queue by the sum of cost accumulated so far and a lower-bound estimate of the cost remaining, with least-cost paths in front.
3. If the goal has been found, announce success; otherwise announce failure.

End BRANCH-AND-BOUND with UNDERESTIMATES (5Marks)

Q.7 a. Differentiate between expert systems and conventional programs. (4)

Answer:

8.2.1 How do Expert Systems Differ from Conventional Programs?

The most basic difference is that expert systems manipulate knowledge while conventional programs manipulate data (Table 8.2).

Table 8.2

Data processing	*Knowledge processing*
Representation and use of static data	Representation and use of data + control = knowledge
Algorithms	Heuristic
Repetitive process	Inferential process
For control and large data kept separately	Large control and few data kept together

Therefore major difference between database and knowledge base is as follows (Table 8.3):

Table 8.3

Database	*Knowledge base*
Collection of data representing facts	Information at a higher level of abstraction
Operates on a single object	Operates on a class of objects rather than a single object
Information needs to be explicitly stated	Uses inferencing power
Represented by relational or hierarchical or network model	Representation is by logic or rules or frames or scripts or semantic nets
Maintained for operational purpose	Used for data analysis and planning

Figure 8.2 shows a generic expert system architecture.

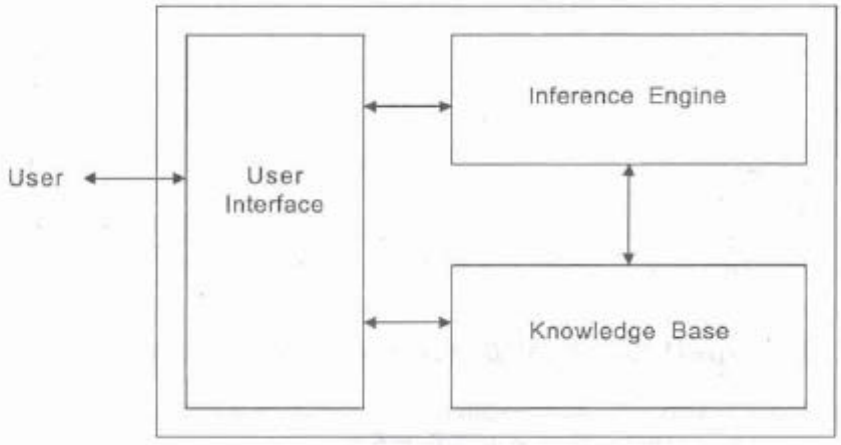


Figure 8.2 Structure of expert system.

b. Based on the characteristics, how the learning algorithms can be categorized? (8)

Answer:

9.7 LEARNING ALGORITHMS

Learning can be viewed as settling down of weights in a systematic manner. There are several learning laws in use, and new laws are being proposed to suit a given application and architecture. Based on the characteristics these learning algorithms expected to possess, they can be categorized as follows:

Supervised or unsupervised. In supervised learning the weight adjustment is determined based on the deviation of the desired output from the actual output. Supervised learning may be used for structural learning or for temporal learning. Structural learning is concerned with capturing in the weights the relationship between the given input-output pattern pairs. Temporal learning is concerned with capturing in the weights the relationship between patterns in a sequence of patterns.

Unsupervised learning discovers failure in a given set of patterns, and organizes the pattern accordingly. There is no externally specified desired output in this case. Unsupervised learning uses mostly local information to update the weights. The local information consists of signal or activation values of the units at the either end of the connection for which the weight update is being made.

Offline or online. In an offline learning all the given patterns are used together to determine the weights. On the other hand, in an online learning the information in each new pattern is incorporated into the network by incrementally adjusting the weights. Thus an online learning allows the neural network to update the information continuously. However, an offline learning provides solutions better than an online learning since the information is extracted using all the training samples in the case of offline training.

Discrete or continuous. Whether the weight values are updated in discrete steps or in continuous time determines the type of that algorithm.

All these factors influence not only the convergence of weights, but also the ability of the network to learn from training samples.

c. What are the features of Biological Neural Networks? (4)

Answer:

Some attractive features of the biological neural network that makes it superior to even the most sophisticated AI computer system for various tasks are as follows:

- **Robustness and fault tolerance:** The decay of nerve cells does not seem to affect the performance significantly.
- **Flexibility:** The network automatically adjusts to a new environment without using any pre-programmed instructions.
- **Ability to deal with a variety of data situations:** The network can deal with information that is fuzzy, probabilistic, noisy and inconsistent.

Collective computation: The network performs routinely many operations in parallel and also given task in distributed manner (4Marks)

Q.8 a. What are the limitations of Neural computing? (4)

Answer:

There are some limitations to neural computing. The key limitation is the neural network's inability to explain the model it has built in a useful way. Analysts often want to know why the model is behaving as it is. Neural networks get better answers but they have a hard time explaining how they got there. This is the reason *neural connection* provides so many tools for exploring output and so many operational choices on the tools which build the model. By

experimenting with different parameters, and fully exploring the results in text and graphics, neural connection users can gain more understanding of the model's behaviour and more confidence in the results.

There are a few other limitations that should be understood. First, it is difficult to extract rules from neural networks. This is sometimes important to the people who have to explain their answer to others and to those who have been involved with artificial intelligence, particularly expert systems which are rule based.

In most analytical methods, you cannot just throw data at a neural network and get a good answer, but you have to spend time understanding the problem or the outcome you are trying to predict. And, you must be sure that the data used to train the system are appropriate (i.e. reflects factors involved) and are measured in a way that reflects the behaviour of the factors. If the data are not representative of the problem, neural computing will not produce good results. This is a classic situation where 'garbage in' will certainly produce 'garbage out'.

Finally, it can take time to train a model from a very complex data set. Neural techniques are computer intensive and will be slow on low end PCs or machines without math coprocessors. It is important to remember though that the overall time to result can still be faster than other data analysis approaches, even when the system takes longer to train. Processing speed alone is not the only factor in performance and neural networks do not require the time programming and debugging or testing assumptions that other analytical approaches do. (each step 1Marks)

b. Write the description and their application for the following network architectures?

(i) ADALINE (Adaline Network)

(ii)HOPFIELD (Hopfield Model)

(iii)BAM (Bidirectional Associative Memory)

(iv)CPN (Counter-Propagation Network) (3×4)

Answer:

i. ADALINE (Adaline Network)

Description: The Adaline is essentially a single-layer back-propagation network. It is trained on a pattern recognition task, where the aim is to classify a bitmap representation of the digits 0 – 9 into the corresponding classes. Due to the limited capabilities of the Adaline, the network only recognizes the exact training patterns. When the application is ported into the multilayer back-propagation network, a remarkable degree of fault-tolerence can be achieved.

Application

- Pattern Recognition
- Classification of Digits 0—9 (3Marks)

ii. HOPFIELD (Hopfield Model)

Description: The Hopfield model is used as an autoassociative memory to store and recall a set of bitmap images. Images are stored by calculating a corresponding weight matrix. Thereafter, starting from an arbitrary configuration, the memory will settle on exactly that stored image, which is nearest to the starting configuration in terms of Hamming distance. Thus given an incomplete or corrupted version of a stored image, the network is able to recall the corresponding original image.

Application

- Autoassociative Memory
- Associative Recall of Images (3Marks)

iii. BAM (Bidirectional Associative Memory)

Description: The bidirectional associative memory can be viewed as a generalization of the Hopfield model, to allow for a heteroassociative memory to be implemented. In this case, the association is between names and corresponding phone numbers. After coding the set of exemplars, the network, when presented with a name, is able to recall the corresponding phone number and vice versa. The memory even shows a limited degree of fault-tolerance in case of corrupted input patterns.

Application

- Heteroassociative memory
- Association of Names and Phone Numbers (3Marks)

iv. CPN (Counter-Propagation Network)

Description: The counterpropagation network is a competitive network, designed to function as a self-programming look-up table with the additional ability to interpolate between entries. The application is to determine the angular rotation of a rocket-shaped object, images of which are presented to the network as a bitmap pattern. The performance of the network is a little limited due to the low resolution of the bitmap.

Application

- Vision Determination of the Angle of Rotation (3Marks)

Q.9 a. Explain the different AI approaches for product selection and recommendation that are useful in B2C e-commerce. (6)

Answer:

AI is used in advising the users on the items they want to examine or buy through Internet. This advice is helpful in navigating a large range of product descriptions. There are a number of different types of product selection and recommendation approaches, such as automated collaborative filtering (AFC) approaches, knowledge based (KB) approaches, and hybrid approaches, etc.

- **ACF approaches:** These approaches aggregate data regarding previous consumer's preferences or purchase patterns, and provide recommendations to prospective buyers based on similarity in overall patterns. For example, GroupLens system recommends news articles based on similarities between users' reading behaviour. This approach has been used in many other areas such as consumers' products and web pages etc. and has become a standard marketing technique in e-commerce. Some ACF systems can even provide the reason and data behind recommendation. A main drawback with ACF is that it is not effective until a large number of users enter their profiles, and a sufficient number of rating items are available in the database.
- **KB approaches:** These approaches are based on the product knowledge base. Some of them are as follows:
  - *Case-based reasoning (CBR) approaches:* CBR system basically accept user preferences, retrieve similar product offered from case base and offer it to the user. The user may modify his preferences if there is no exact match. The above steps are repeated until the user selects a product or quits. The most common technique in e-commerce applications of CBR is *nearest neighbour retrieval.*
  - *Content-based recommendor approaches:* These approaches depend on machine learning-based classification. For example, the news filtering system NewsDude suggests news stories that a user might like to read. These systems primarily use supervised machine learning to induce a classifier and the classifier is used to

discriminate between items likely to be of interest to the user and items that are likely to be uninteresting.

- **Hybrid approaches:** Hybrid approaches are primarily a combination of ACF and KB approaches. In some systems, the ACF is used in the poet-processing stage and the systems are predominantly knowledge based. Other systems check whether there are a sufficient number of feedbacks from previous users. This number is used as a threshold value to decide on what approach is to be used. If the value is less than a proper KB approach is used, otherwise ACF approaches are used. The threshold value can be determined interactively based on the products and business. Some systems attempted to utilize CBR approaches for ACF. (6Marks)

b. What are the different types of clinical task to which expert systems can be applied? (6)

Answer:

There are many different types of clinical task to which expert systems can be applied:

- **Generating alerts and reminders:** In so-called real-time situations, an expert system attached to a monitor can warn of changes in a patient's condition. In less acute circumstances, it might scan laboratory test results or drug orders and send reminders or warnings through an e-mail system. (1Marks)
- **Diagnostic assistance:** When a patient's case is complex, rare or the person making the diagnosis is simply inexperienced, an expert system can help come up with likely diagnosis based on patient data. (1Marks)
- **Therapy critiquing and planning:** Systems can either look for inconsistencies, errors and omissions in an existing treatment plan, or can be used to formulate a treatment based upon a patient's specific condition and accepted treatment guidelines. (1Marks)
- **Agents for information retrieval:** Software agents can be sent to search for and retrieve information, for example on the Internet that is considered relevant to a particular problem. The agent contains knowledge about its user's preferences and needs, and may also need to have medical knowledge to be able to assess the importance and utility of what it finds. (2Marks)
- **Image recognition and interpretation:** Many medical images can now be automatically interpreted, from plane X-rays through to more complex images like angiograms, CT and MRI scans. This is of the value in mass-screenings, for example, when the system can flag potentially abnormal images for detailed human attention. (1Marks)

c. Explain TravelPlan architecture that integrates different types of agents. (4)

Answer:

TravelPlan is a MAS approach that integrates different types of agents. It could be summarized as:

- **UserAgent:** This agent pays attention to the user's queries and shows the solution. It analyzes the problem and obtains an abstract representation. Subsequently it requests to a PlannerAgent for a solution to that problem. The UserAgent has different skills like communication with PlannerAgents and users, or learning the users profiles necessary to customize the system answer.
- **PlannerAgent:** The main goal of PlannerAgent is reasoning about UserAgents and other PlannerAgents problems, and find out a set of possible solutions. PlannerAgents have different skills like communication (with different agents in the system), planning (its

main reasoning module) and learning (using CBR techniques to index and to categorize each stored plan).

- **WebBot:** This agent fills in the details (requested by PlannerAgent) obtaining the required information from Internet. Different partial solutions given by the web agents are combined by the PlannerAgents to obtain a detailed solution (or solutions) to the UserAgent queries. (4 marks)

Text Book

Introduction to Artificial Intelligence, Rajendra Akerkar, PHI,2005