UNIT 3 INTRODUCTION TO KNOWLEDGE DATABASES

Structure

3.0 INTRODUCTION

A knowledge base management system (KBMS) is a computer system that manages the knowledge in a given domain of interest and exhibits reasoning power to the level of a human expert in this domain. In typical Artificial Intelligence (AI) application, knowledge representation requires data structures with rich semantics that go beyond the simple structure of the relational model. AI is the part of computer science with designing inlelligent computer systems, that is systems that exhibit the characteristics we associate with intelligence in human behaviour. Furthermore, operations in a knowledge base are more complex than those in traditional database. When a rule is added the system must check for contradiction and redundancy. Such operations cannot be represented directly by relational operations and the complexity of checking increases rapidly as the size of knowledge base grows.

3.1 OBJECTIVES

After going through this unit you will be able to :

- define what is knowledge, hypothesis and belief
- explain what is a Knowledge base system
- differentiate between Knowledge base system and a database system
- list several knowledge representation schemes

3.2 DEFINITION AND IMPORTANCE OF KNOWLEDGE

Definition and Importance of Knowledge

Knowledge can be defined as the body of facts and principles accumulated by human-kind or the act, fact or state of knowing. While this definition may be true, it is far from complete. We know that knowledge is much more than this. It is having a familiarity with language, concepts, procedures, rules, ideas, abstractions, places, customs, facts and associations, coupled with an ability to use these notions effectively in modelling different aspects of the world. Without this ability, the facts and concepts are meaninglessand, therefore, worthless. The meaning of knowledge is closely related to the meaning of intelligence. Intelligence requires the possession of and access to knowledge. And a characteristic of intelligent people is that they possess much knowledge.

In biological organisms, knowledge is likely stored as complex structures of interconnected

neurons. The structures correspond to symbolic representation of the knowledge possessed by the organism, the facts, rules and so on. The average humane brain weighs about 3.3 pounds and contains an estimated number of 10^{12} neurons. The neurons and their interconnection capabilities provide about 10^{14} bits of potential storage capacity.

In computers, knowledge is also stored as symbolic structures, but in the form of collections of magnetic spots and voltage states. State-of-thc-art storage in computers is in the range of 10^{12} bits with capacities doubling about every three to four years. The gap between human and computer storage capacities is narrowing rapidly. Unfortunately, there is still a wide gap between representation schemes and efficiencies.

A common way to represent knowledge external to a computer or a humane is in the form of written language. For example, some facts and relations represented in printed English are

> Jancy is tall.
>
> Ram loves Sita.
>
> Som has learned to use recursion to manipulate Binary tree in several programming languages.

The first item of knowledge above expresses a simple fact, an attribute possessed by a pcrson. The second item expresses a complcx binary relation between two persons. The third item is the most complex, expressing relations between a person and more abstract programming concepts. To truly understand and make use of this knowledge, a person needs other world knowledge and the ability to reason with it.

Knowledge may be declarative or procedural. Procedural knowledge is compiled knowledge related to the performance of some task. For example, the steps used lo solve an algebraic equation are expressed as procedural knowledge. Declarative knowledge, on the other hand, is passive knowledge expressed as statements of facts about the world. Personnel data in a database is typical of declarative knowlcdge. Such data are explicit pieces of independent knowledge.

Frequently, wc will be interested in the use of heuristic knowledge, a special type of knowlcdgc used by humans lo solve complex problems. Heuristics are tlie knowledge used to make good judgments, or the strategies, tricks or "rules of thumb" used to simplify the solution of problems. Heuristics arc usually acquired with much experience. For examplc, in locating a fault in a TV set, an experienced technician will not start by making numerous voltage checks when it is clear that the sound is present but the picture is not, but instead will immediately reason that the high voltage flyback transformer or related component is the culprit. This type of reasoning may not always bc correct, but it frequently is, and then it leads to a quick solution.

Knowledge should not be confused with data. Some scientists emphasize this difference with the following example. A physician treating a patient uses both knowledge and data. The data is the patient's record, including patient history, measurements of vital signs, drugs given, response to drugs, and so on, whcrcas the knowledgc is what the physician has learned in medical school and in the years of internship, residency, specialization, and practice. Knowlcdge is what the physician now learns in journals. It consists of facts, prejudices, beliefs, and most importantly, heuristic knowlcdgc.

Thus, wc can say that knowledge includes and requires the use of data and information. But it is morc. It combines relationships, correlations, dcpendcncics, and the notion of gestalt with data and information.

Even with the abòve distinction, wc have been using knowledge in its broader sense up to this point. At times, however, it will be useful or cvcn necessary to distinguish bctwccn knowlcdgc and other concepts such as belief and hypotheses. For such cases wc make the following distinctions. Wc dcline belief as essentially any meaninglut and coherent expression that can be represented. Thus, a belief may be true or false. We define a hypothesis as a justified belief that is not known to bc true, Thus, a hypothesis is a belief which is backed up with some supporting evidence, but it may still be false. Finally, we define knowledge as true justified bclicf.

Two other knowledge tenns which wc shall occasionally use arc epistemology and metaknowledge. Epistemology is the study of the nature of knowledge, whereas metaknowledge is knowledge about knowlcdgc, that is, knowlcdgc about what we know.

In this section we have tried to give a broader definition of knowledge than that commonly found in dictionaries. Clearly, we have not offered a scientific definition, we are not able to measure knowledge. How then will we know when a system has enough knowledge to perform a specified task? Can we expect to build intelligent systems without having a more precise definition of either knowledgc or intclligcnce? In spite of o w ignorance about knowledge, the answcr is definitely yes.

Finally, our overall picture of knowlcdge cannot be complete without also knowing the meaning of closely related concepts such as understanding, learning, thinking, remembering, and reasoning. These concepts all dcpend on the use of knowlcdge. But then just what is learning, or reasoning, or understanding? Here too wc will find dictionary definitions lacking. And, as in the case of knowledge and intclligencc, we cannot give scientific definitions for any of these terms either.

The Importance of Knowledge

AI has given new meaning and importance to knowledge. Now, for the first time, it is possible to "package" specialized knowlcdge and sell it with a system that can use it to reason and draw conclusions. The potential of this important dcvclopmcnt is only now beginning to be realised. Imagine being able to purchase an untiring, reliable advisor that gives high level professional advice in specialised areas, such as manufacturing techniques, sound financial strategies, ways to improve one's health, top marketing sectors and strategies, optimal farming plans, and many other important matters. Wc are not far from the practical realisation of this, and those who create and market such systems will have more than just an economic advantage over the rest of the world.

3.3 WHAT IS A KNOWLEDGE BASE SYSTEM?

One of the important lessons learned in AI during the 1960s was that general purpose problem solvers which used a limited number of laws or axioms were too wwk to be effective in solving problems of any complexity. This rcalisation eventually led to the design of what is now known as Knowledge base system, systems that depend on a rich base of knowledge to perform difficult tasks,

Edward Fcigenbaum summarised this new thinking in a paper at the International Joint Conference on Artificial Intelligence (IJCAI) in 1977. He emphasised the fact that the real powcr of an expcrt system comes from the knowledge it possesses rather than the particular inference schemes and other formalisms it cmploys. This new view of AI systems marked the turning point in the development of more powerful problem solvers. It formed the basis for some of the new emerging expert systems being devclopcd during the 1970s including MYCIN, an expcrt system developed to diagnose infectious blood diseases. An expcrt system contains knowlcdge of experts in a particular domain along with an inferencing mechanism and an explanation sub-system. It is also called knowlcdgc base system.

Since this realisation, much of the work donc in AI has been rclatcd to so-called Knowledge base systcms, including work in vision, learning, general problcm solving and **natural language understanding.** This in turn has led to more emphasis being placed on research rclatcd to knowlcdgc representation, memory organisation, and the use and manipulation of knowledge.

Knowledge base systems get their power from the expcrt knowledge that has been coded into facts, rules, heuristics, and procedures. The knowledgc is stored in a knowledge base separate from the control and infercncing components. This makes it possible to add new knowledge or refine existing knowlcdge without recompiling the control and inferencing programs. This greatly simplifies the construction and maintenance of Knowledge base systems.

In the knowledge lies the power! This was the message learned a few farsighted researchers at Stanford University during the late 1960s and early 1970s.

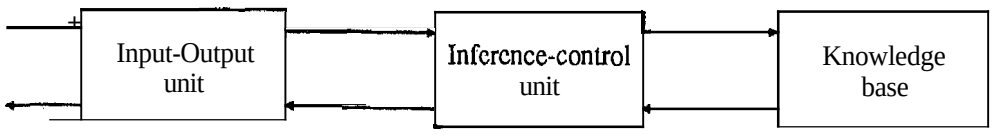

Figure 1 : Components of a Knowledge-based system.

The proof of their message was provided in the first Knowledge base expert systems which were shown to be more than toy problem solvers. Tticsc first systems were real world problem solvers, tackling such tasks as determining complex chemical structures given only the atomic constituents and mass spectra data from samples of the compounds and later performing medical diagnoses of infectious blood diseases.

Using the analogy of a DBMS, we can define a knowledge base management system (KBMS) as a computer system used to manage and manipulate shared knowledge. A knowledge base system's manipulation facility includes a reasoning facility, usually including aspects of one or more of the following forms of reasoning : deductive, inductive. or abductive. Deductive reasoning implies that a new fact can be inferred from a given set of facts or knowledge using known rules of inference. For instance, a given proposition can be found to be true or false in light of existing knowledge in the form of other propositions believed to be either true or false. Inductive reasoning is used to prove something by first proving a base fact and then the increment step; having proved these, we can prove a generalized fact. Abductive reasoning is used in generating a hypothesis to explain observations. Like deductive reasoning, it points to possible inferences from related concepts; however, unlike deductive reasoning, the number of inferences could be more than one. The likelihood of knowing which of these inferences corresponds to the current slate of the system can be gleaned from the explanations generated by the system. These explanations can facilitate choosing among these alternatives and arriving at the final conclusion.

In addition to the reasoning facility, a knowledge base system may incorporate an explanation facility so that.the user can verify whether the reasoning used by the system is consistent and complete. The reasoning facility also offers a form of tutoring to the uninitiated user. The so-called expert systems anti the associated expert system generation facilities are one form of knowledge base systems that have emerged from research labs and are being marketed commercially. Since a KBMS includes reasoning capacity, there is a clear benefit in incorporating this reasoning power in database application programs in languages such as COBOL and Pascal.

Most knowledge base systems are still in the research stage. The first generation of commercial KBMSs are just beginning to emerge and integration of a KBMS with a DBMS is a current research problem. However, some headway has been made in the integration of expert systems in day-to-day database applications.

3.4 DIFFERENCE BETWEEN A KNOWLEDGE BASE SYSTEM AND A DATABASE SYSTEM

There is no consensus on the difference between a knowledge base systcm and a database systcm, In a DBMS, the starting point is a data model to represent the data and the interrelationships between them; similarly, the starting point of a KBMS is a knowledge representation scheme. The requirements for any knowledge representation schema should provide some mechanism to organise knowledge in appropriate hierarchies or categories, thus allowing easy access to associated concepts. In addition, since knowledge can be expressed as rules and exceptions to rules, exception-handling features must be present in the knowlcdge stored in the system must be insulated from changes in usage in its physical or logical structure. This concept is similar to the data independence concept used in a DBMS. To data, little headway has been made in this aspect of a KBMS.

A KBMS is developed to solve problem for a finite domain or portion of the real world. In developing such a systcm, the designer selects a significant objects and relationships among these objects. In addition to this domain-specific knowledge, general knowledge such as concepts of up, down, far, near, cold, hot, on top of, and besides must be incorporated in the KBMS. Another type of knowlcdge, which we call common sense, has yet to be successfully incorporated in the KBMS.

The DBMS and KBMS have similar architectures; both contain a component to model the information being managed by the system and have a subsystem to respond to queries. Both systems are used to model or represent a portion of the real world of interest to the application. A database system, in addition to storing facts in the form of data, has limited capability of establishing associations bctwecn these dam. These associations could be pre-established as in the case of the network and hierarchical models, or established using

common values of shared domains as in the relational model. A knowledge base system exhibits similar associative capability. However, this capability of establishing associations between data and thus a means of interpreting the information contained is at a much higher level in a knowledge base system, ideally at the level of a knowledgeable human agent.

One difference between the DBMS and KBMS that has been proposed is that the knowledge base system handles a rather small amount of knowledge, whereas a DBMS efficiently (as measured by response performance) handles large amounts of shared data. However, this distinction is fallacious since the amount of knowledge has no known boundaries and what this says is that existing knowledge base systems handle a very small amount of knowledge. This does not mean that at some future date we could not develop knowledge base systems to efficiently handle much larger amounts of shared knowledge.

In a knowledge base system, the emphasis is placed on a robust knowledge representation scheme and extensive reasoning capability. Robust signifies that the scheme is rich in expressive power and at the same time it is efficient. In a DBMS, emphasis is on efficient access and management of the data that model a portion of the real world. A knowledge base system is concerned with the meaning of infonnation, whereas a DBMS is interested in the information contained in the data, However, these distinctions are not absolute.

For our purposes, we can adopt the following informal definition of a KBMS. The important point in this definition is that we are concerned with what the system does rather than how it is done.

A knowledge base management system is a **computer system that manages the knowledge in a given domain or field of interest** and **exhibits reasoning power to the level of a human expert in this domain.**

AKBMS, in addition, provides the user with an integrated language, which serves the purpose of die traditional DML of the existing DBMS and has the power of a high-level application language. A database can be viewed as a very basic knowledge base system in so far as it manages facts. It has been recognised that there should bc an integration of the DBMS technology with the reasoning aspect in the development of shared knowledge bases. Database technology has already addressed the problems of improving system performance, concurrent access, distribution, and friendly interface; these features are equally pertinent in a KBMS. There will be a continuing need for current DBMSs and their functionalities co-existing with an integrated KBMS. However, the reasoning power of a **KBMS** can improve the ease of retrieval of pertinent information from a **DBMS.**

3.5 KNOWLEDGE REPRESENTATION SCHEMES

Knowledge is the most vital part of **Knowledge Base System** or **Expert System.** These systems contain large amounts of knowledge to achieve high performance. A suitable **Knowledge Representation** scheme is necessary to represent this vast amount of knowledge and to perform inferencing over the Knowledge Base (KB), A Knowledge Representation scheme means a set of syntactic and semantic conventions to describe various objects. The syntax provides a set of rules for combining symbols and arrangements of symbols to form expressions.

Knowledge Representation is a non-trivial problem, which continues to engage some of the best minds in this field even after the successful devclopinent of many a Knowledge Base System. Some of the important issues in Knowledge Representation arc the following:

i) **Expressive Adequacy**: What knowledge can be and cannot be represented in a particular Knowledge Representation scheme ?

ii) **Reasoning Efficiency :** How much effort is required to perform inferencing over the KB? There is generally a trade off between expressive adequacy and reasoning efficiency,

iii) **Incompleteness**: What can be left unsaid about a domain and how does one perform infcrencing over incomplete knowledge ?

iv) **Real World Knowledge** : How can we deal with attitudes such as beliefs, desires and intentions ?

Major Knowledge Representation schemes are based on production rules, frames, semantic nets and logic. Facts and rules can be represented in these Knowledge Representation schemes. Inference Engines using forward chaining, backward chaining or a combination thereof are used alongwith these Knowledge Representation schemes to build actual Expert System. We will briefly describe these Knowledge Representation schemes and inferencing engines.

3.5.1 Rule Based Representation

A rule based system is also called production rule system. Essentially, it has three parts, working memory, rule memory or production memory and interpreter. Working memory contains facts about the domain. These are in the form of triples of objects, attribute and value. These facts are modified during the process of execution. Some new facts may be added as conclusions.

Production memory contains IF-THEN rules. IF part contains a set of conditions connected by AND. Each condition can have different other conditions connected by AND or OR. Each condition can give either true or false as its value. THEN part has a set of conclusions or actions. Conclusions may change values of some entity or may create new facts.

Arule can be fired when all the conditions in it are true. If any of the conditions is not true or unknown, the rule cannot be fired. If it is unknown, the system will try to determine its value. Once a rule has fired, all its conclusions and actions are executed.

For firing a rule, the system looks into its database. If a rule has some of its conditions satisfied, it is a candidate for further exploration. There may be more than one such rule. This conflict is resolved by some strategy like choosing the rule which contains the maximum number of satisfied conditions, or there may be metarules which may be domain dependent to move the reasoning in a particular direction.

Rules may be used in both forward and backward reasoning. When it is used in forward mode, the system starts with a given set of initial data and infers as much information as possible by application of various rules. Again new data.are used to infer further. At any point system may ask the user to supply more information, if goal state has not been reached and no more rules can be applied. System keeps on checking for goal state at each firing of rules. Once goal state has been detected reasoning comes to an end. In backward reasoning mode, reasoning starts with the god and rules arc selected if they have the goal in their right hand side (RHS). To achieve the goal, left hand side (LHS) conditions have to be true. These conditions become new sub-goals. Now the system tries to achieve these sub-goals before trying the main goal. At some point it may not be possible to establish goal by application of rules. In this situation the system asks the user to supply the information.

It may bc noted that these rules are not IF-THEN programming constructs available in most of the procedural Programming languages. These are different in the sense that they are not executed sequentially. Their execution depends on the state of the database which determines which are the candidates rules. Another difference is that IF-part is a complex pattern and not just a Boolean expression.

Rules have been used in many classical systems like MYCIN, RI/XCON etc. Even today it is the most frequent used Knowledge Representation scheme. The reason is that most of the time, experts find it easier to give knowledge in the form of rules. Further, rules can be easily used for explanations.

One problem with rules is that when they grow very large in number it becomes difficult to maintain them because KB is unstructured. Some techniques like context in MYCIN solve the problem to some extent.

3.5.2 Frame Based Representation

The concept of frame is quite simple. When we encounter a new situation, we do not analyse it from scratch. Instead we have a large number of structures or (records) in memory representing our experiences. We try to match the current situation with these structures and then the most appropriate one is chosen. Further details may be added to this chosen structure so that it can exactly describe the situation. A computer representation of this common knowledge is called a frame.

It is convenient to create a knowledge base about situations by breaking it into modular

chunks, called frames. Individual frames may be regarded as a record or structure.

Each frame contains slots that identify the type of situations or specify the parameters of a particular situation.

A frame describes a class of objects such as ROOM or BUILDING. It consists of various slots which describe one aspect of the object. A slot may have certain condition which should be met by the filler. A slot may also have default value which is used when the slot value is not available or cannot be obtained by any other way. If added procedure describes what is to be done if slots get a value. Such information is called facet of slot.

An example is presented below:

{CHAIR

IS-A	:	FURNITURE
COLOUR	:	BROWN
MADE-OF	:	WOOD
LEG	:	4
ARMS	:	default: 0
PRICE	:	100

Reasoning with the knowledge stored in a frame requires choosing an appropriate frame for the given situation. Some of the ways information may be inferred are the following:

a) If certain information is missing from current situation, it can be inferred. For example, if we have established that the given object is a room, we can infer that room has got a door.

b) Slots in a frame describe components of situation. If we want to build a situation then information associated with the slots can be used to build components of the situation.

c) If there is any additional feature in the object which can be discovered using a typical frame, it may require special attention. For example, a man with a tail is not a normal man.

3.5.3 Semantic Nets

Semantic net representation was developed for natural language understanding. Semantic net was originally designed to represent the meaning of English words. It has been used in many expert systems loo. It is used for representation of declarative knowledge. In semantic nets, the knowledge is represented as a set of nodes and links. A node represents an object or concept and a link represents relationship between two objects (nodes).

Moreover, any node may be linked to any number of other nodcs, so giving rise to a formation of network of facts. An example is shown in the following figure.

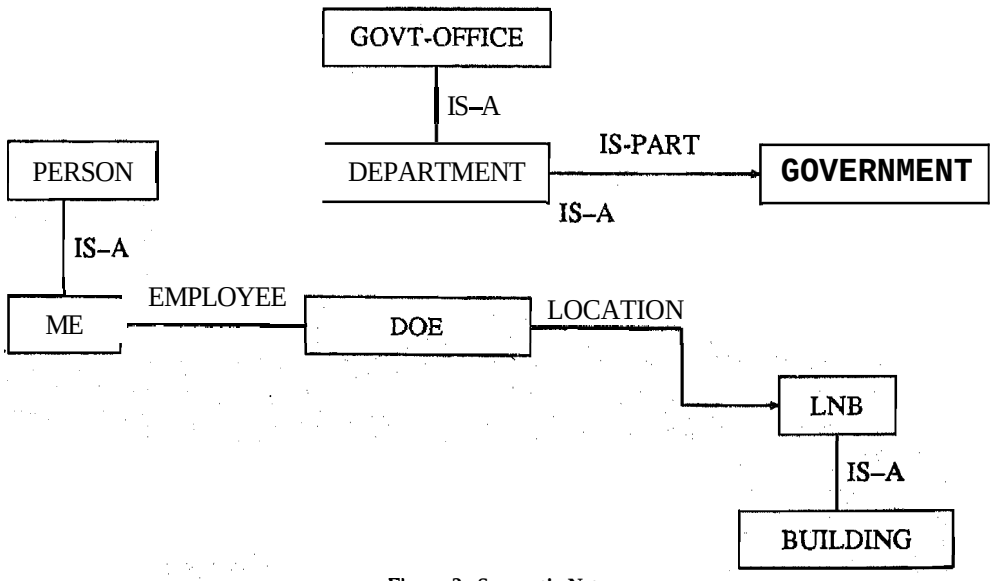

Figure 2 : Semantic Net

A semantic net as shown in figure 2, cannot be represented like this in computer. Every pair and its link are stored separately. For example, IS—A (DOE, Department) in **PROLOG** represents

DEPARTMENT

Figure 3 : One-wry link representations

The link as shown in the figure is a one-way link. If we want an answer to "who is my employer ?" the system will have to check all the links coming to node ME. This is not computationally efficient. Hence revcrsc links are also stored. In this case we add

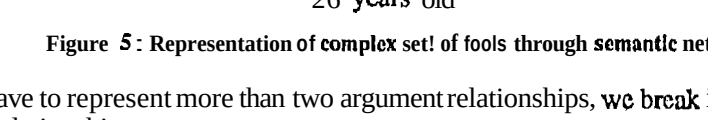

Figure 4 : Representation of a reverse link

In LISP the basic semantic network unit may bc programmed as an atom/property list combinalion. The unit given in department-DOE semantic networks would bc composed of "DOE" as the atom, "IS–A" as a propcrty and "Department" as the value of that property. The value "Dcpartrnent" is of course, an atom in its own right and this may have a property list associated it as well. "Is-a" relationship indicates that one concept is an attribute of the other. Other links (relationship) of particular use for describing object concepts are "has", indicating that one concept is a part of the other. Using such relations, it is possible lo represent complex set of facts through scmanlic network. The following figure illustrates one possible representation of facts about an cmployce "AKSHAY". These include—

"Akshay is a bank manager"
"Akshay works in the State Bank of India locate din IGNOU Campus"
"Akshay is 26 ycars old"
"Akshay has blue eyes"

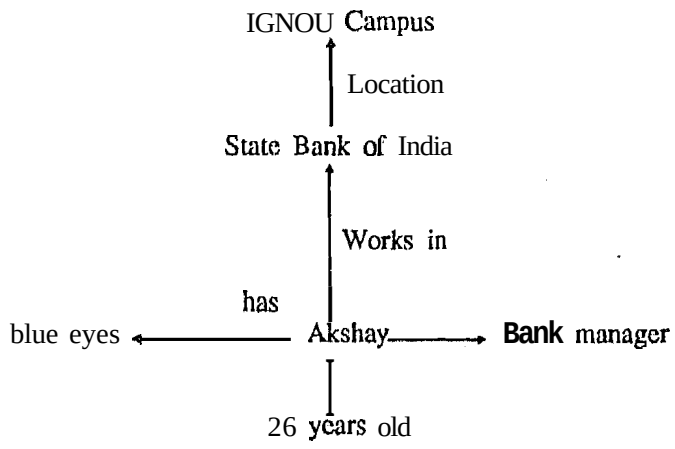

Figure 5 : Representation of complex set! of fools through semantlc nets

When we have to represent more than two argument relationships, we break it down into arguments relationships.

SCORE (INDIA AUSTRALIA (250 150))
can be written as
participant (match-1 INDIA)
participant (match-1 AUSTRALIA)
score (match-1 (250 150))

As with an Knowledge Representation scheme, the problem solving power comes from the ability of the program to manipulate knowledge to solve a problem. Intersection search is used to find the relationship between two objects. In this case activation sphere grows from the two modes and intersects each other at some time. The corresponding paths give the correct relationship. More techniques have been developed to perform more directed search:

3.5.4 Knowledge Representation Using Logic

Traditionally logic has been studied by philosophers and mathematicians in different countries in order to describe and understand the world around us. Today computer scientists are using this tool to teach a computer about the world. Here we discuss propositional and predicate logic.

We can easily represent real-world facts as logical propositions in propositional logic. In propositional logic we deal with propositions like

It is raining. (RAINING)

It is sunny. (SUNNY)

It is windy. (WINDY)

If it is raining then it is not sunny.

(RAINING ⟶ ~SUNNY)

Given the fact that "It is raining" we can deduce that it is not sunny.

But the representational power of propositional logic is quite limited. For example, suppose we have to represent

Vivek is a man.

Anurag is a man.

We may represent them in computer as VIVEKMAN and ANURAGMAN. But from these, we don't get any information about similarity between Vivek and Anurag. A better way of representation is

MAN (VIVEK)

MAN (ANURAG)

Consider the sentence 'All men are mortal'. This requires quantification like

The form of logic with these and certain other extra features is called predicate logic. The basic elements are described here.

Here capital letter P, Q etc. stand for predicates. Aprcdicate is of the form predicate-name (arg1..argn)

It can have values true or false. Apredicate represents among the arguments.

AND ∧ (P∧ is true when both are true)

OR v (P v Q is true when atleast one is true)

NOT ~ (~P is true when P is false)

Implies (P → Q is true unless P is true and Q is false) means for all the values of X, P holds.

P(X) means there exists at least one value of X for which P holds.

means only some values are true.

The predicatc logic has the following two properties:

a) Completeness: If P is a theorem of predicate logic then it can be derived only by the inferences rules available in predicate logic;

b) Soundness: There is no P such that both P and NOT P are theorems.

The dccidability property of propositional logic does not carry over into predicatc logic. The following inferences rules are some of the important rules available in predicatc logic:

Modus Ponens : IS P → Q and P is true then Q is true

Modus Tollens	:	If P → Q **and** Q is false **then** P is false
Chaining	:	If P∨Q **and (NOT P)** ∨Q then Q is **true**
Reduce to	:	**P** and NOT P reduces to ( ).

Most of the AI theorem provers for clausal form use resolution as the only way of inferencing. This subsumes h e above five rules of inference. Resolution proves a theorem by refutation. First a normal form of clauses is obtained and then negation of the theorem is added to it. If it leads to a contradiction then the theorem is proved. **A** discussion **of** detailed algorithm is beyond the scope of **this** handout

Let us now explore the use **of** predicate logic as a way of representing knowledge by looking at a specific example.

i) Anil is a Manager

ii) Anil is a disciplined

iii) All staff are either **loyal to** Anil or hate him

iv) Everyone is loyal **to someone**

The facts described by **these** sentences. can be represented in Predicate logic as follows:

i) **Anil** is a Manager
Manager (Anil)

ii) Anil is a disciplined
disciplined (Anil)

iii) **All** staff are either **loyal to Anil** or hate him.

iv) Everyone is loyal to someone

vj Predicate logic is useful for representing simple English sentences into a logical statement. But, it creates ambiguity for complicated sentences.

Check Your Progress

1. Define and describe the difference between knowledge, belief, hypothesis and data

.....

.....

.....

2. What is the difference between declarative and procedural knowledge ?

.....

.....

.....

3. What are the techniques available for the knowledge representation ?

.....

.....

.....

.....

4. How the knowledge representation is done through Semantic Network ?

.....

.....

.....

.....

3.6 SUMMARY

In this unit, we defined a knowledge base system as a computer system used for the management and manipulation of shared knowledge. We compared a knowledge base system with a DBMS and pointed out similarities and differences. We also considered the different schemes used to represent knowledge. The semantic network, first order logic, rule based system, frames and procedural representation.

3.7 MODEL ANSWERS

Check Your Progress

1. Knowledge can be defined as the body of facts and principles accumulated by human-kind or the act, fact, or state of knowing. While this definition may be true, it is far from complete. We know that knowledge is much more than this. It is having a familiarity with language, concepts, procedures, rules, ideas, abstractions, places, customs, facts, and associations, coupled with an ability to use these notions effectively in modelling different aspects of the world. Without this ability, the facts and concepts are meaningless and therefore worthless. The meaning of knowledge is closely related to the meaning of intelligence. Intelligence requires the possession of and access of knowledge. And a characteristic of intelligent people is that they possess much knowledge,

   Thus, we can say that knowledge includes and requires the use of data and information. But it is more. It combines relationships, correlations, dependencies, and the notion of gestalt with data and information.

2. Knowledge may be declarative or procedural. Procedural knowledge is compiled knowledge related to the performance of some task. For example, the steps used to solve an algebraic: equation are expressed as procedural knowledge. Declarative knowledge, on the other hand, is passive knowledge expresses as statements of facts about the world, Personnel data in a database is typical of declarative knowledge. Such data are explicit pieces of independent knowledge.

3. The following are knowledge representation techniques:

   i) Rule Based Representation

   ii) Frame Based Representation

   iii) Semantic Nets

   iv) Knowledge Representation Using Logics

4. In semantic nets, the knowledge is represented as a set of nodes and links. A node represents an object or concept and a link represents relationship between two objects (nodes).

3.8 FURTHER READING

1. An Introduction to Database Systems by Bipin C. Desai, Galgotia Publications Pvt. Ltd.