Q10.(a) Describe the role of various issues in knowledge representation with examples.

MDU BCA Nov. 2016

OR

Explain various issues in knowledge representation with examples.

MDU BCA April 2016

Ans. The key to problem solving is a large amount of knowledge and some mechanism for manipulating that knowledge. The basic goal of knowledge representation is to help derive conclusions from the knowledge. When many knowledge based techniques are used into a single problem many issues may arise. Some of the issues that arise while using knowledge representation technique are the following:-

1. Important attributes
2. Relationships among attributes
3. Choosing Granularity
4. Set of objects
5. Finding Right Structure

1. Important attributes:-

There are two attributes that are of very general significance. These are "instance" and "is a". These attributes are significant because they support property inheritance.

2. Relationships among attributes:-

The attributes that are used to describe objects are entities that we represent. They have certain properties independent of the specific knowledge they encode. The relationship between the attributes of an object may hold the following four properties:

(a) Inverses

(b) An 'is a' hierarchy of attribute

(c) Techniques for Reasoning about values

(d) Single Valued Attributes

(a) Inverses:-

Entities are related to each other in a number of different ways. These relationships are described as attributes. This may begin at the object being described and terminate at the object representing the value of the specified attribute. But the focus can be on the object representing the value. Although different but there is still a relationship between the two entities.

(b) An 'is a' hierarchy of attribute:-

It is generalization vs specialization. For example, the height of a person is a specialization of general attribute physical size. These generalization-specialization relationships are important for attributes because they support inheritance.

(c) Techniques for Reasoning about values:-

Sometimes values attributes are specified explicitly when a knowledge base is created. The reasoning system must reason about values which have not been given explicitly. Several kinds of information play a role in this reasoning. For example, the value of height must be a number measured in a unit of length. Another example, the age of a person can not be more than the age of either of his parents etc.

(d) Single Valued Attributes:-

A specific but very useful kind of attribute is one that is guaranteed to take a unique value. For example, President of India. Another example is that a player can at a time be a member of only one team.

The knowledge representation system takes several different approaches to provide support for single-valued attributes.

3. Choosing Granularity:-

Granularity means what level should the knowledge be represented and what are the primitives.

- Should there be a small number or should there be a large number of low-level primitives or High-level facts.
- High level facts may not be adequate for inference while Low-level primitive may require a lot of storage.

Granularity of representation is an important aspect in knowledge representation. It deals with depth of detail. The Granularity decides the size of knowledge base. As more and more information is stored, the size of knowledge base grows. The more detail will mean more accurate information but it also increases the size of knowledge base. If fewer details are stored the decisions requiring more depth can not be generated.

4. Set of objects:-

There are certain properties of objects that are true as a member of a set but not as individual. It is important to represent sets of objects if a property is true for all or most elements of a set then it is more efficient to associate it with the set rather than to associate it explicitly with every element of the set. This can be done in the following manner:

- ***A logical representation through the use of universal quantifier and***
- ***A hierarchical structure where nodes represent sets and inheritance propagate set level assertion down to individual.***

5. Finding Right Structure:-

For describing a particular situation right structure is selected. This requires selecting an initial structure and then revising or reviewing the choice. For this, it is important to know the following:-

- ***How to make an initial selection of the most appropriate structure?***

- *How to fill in details from the current situations?*
- *How to find a better structure in case the one chosen initially is not appropriate?*
- *What to do if none of the available structure is appropriate?*
- *When to create a new structure?*

Q10. (b) Define predicate logic. How is it useful and used? Explain with an example.

MDU BCA May 2016

Ans. Predicate logic or first order predicate logic (FOPL) is one of the oldest and most important knowledge representation schemes used in Artificial Intelligence.

Predicate logic allows describing the objects involved and their relationships. Predicate logic is a powerful tool that can express and give reasoning and it is build on the top of propositional logic, making the use of ideas.

Every complete sentence contains two parts – a subject and a predicate.

Subject: The subject is what (or whom) the sentence is about.

Predicate: It tells something about the subject.

For example, in the sentence, "Suman is an author", subject is "Suman" and predicate is "author".

Sentences involving the predicates that describe the property of objects are denoted by P(x) where

P – the predicate,
x – is a variable denoting any object

Characteristics of predicate logic:

Following are the characteristics of predicate logic:

1. It allows logical inferencing.
2. Program designing is its application area.

3. It is more accurate knowledge representation (KR) of facts of real world.
4. It is better theoretical foundation.

Symbols used:

Following are the symbols used in predicate logic:

1. **Predicate symbols:** These are used to represent a relation in a domain. For example, to represent a sentence like "Suman writes books", the predicate symbol Write is used. The simple formula will be:
   Write (Suman, Books)

2. **Function symbols:** They denote functions in the domain of discourse. For example, Ravi's mother is married to Ravi's father, the atomic formula would be:
   MARRIED[mother (Ravi), father (Ravi)] where mother and father are function symbols.

3. **Variable symbols:** Lowercase unsubscripted or subscripted letters like x, y, z, u, v etc. they can assume different values over a given domain.
   For example, write (x, y).

4. **Constants:** A constant symbol is used to represent objects or entities in a domain. These objects or entities may be physical objects, people, concepts or anything that is to be named. For example, in the above formula Suman and Books are constant symbols.

5. **Quantifiers:** A quantifier is a symbol that permits one to declare or identify the range or scope of the variable in a logical expression.

There are two basic quantifiers used in logic. They are Universal quantifier ($\forall$) and Existential quantifier ($\exists$).

- **Universal quantifiers:** It means something is true for all possible values of a variable. Using this, the sentence "All parrots are green" can be expressed as:

  $$\forall x\,[\text{parrot}(x) \Rightarrow \text{colour}(x, \text{green})]$$

- **Existential quantifier:** It means the formula consisting of an existential quantifier has a value true for at least one assignment of x to an entity in the domain. For example, "There is a person who drives the car" could be represented as:

  $$\exists\, x\, \text{drives}(x, \text{car})$$

6. **Logical operators/connectives:** It uses five connectives, i.e., $\sim$ (not), $\wedge$ (and), $\vee$ (or), $\rightarrow$ (implication) and $\leftrightarrow$ (equivalence).

Example:

Consider the following facts:

(a) Ravi is married to Shweta.

(b) Marcus was a man.

(c) The last meeting of the club was in Mohan's house.

(d) Everyone is loyal to someone.

(e) If it rains then sky will be cloudy.

(f) If you will not work hard, you will fail.

(g) All dogs are mammals.

Representation of the above facts in predictable logic is:

(a) Married (Ravi, Shweta)

(b) Man (Marcus)

(c) Meeting (last, Mohan)

(d) $\forall x \exists y$ loyalto (x, y)

(e) raining $\rightarrow$ cloudy (sky)

(f) $\neg$ work hard $\rightarrow$ fail

(g) $\forall$ x dog (x) $\rightarrow$ Mammal (x)

Ans. Propositional Logic:-

Propositional Logic is a simple language that is useful for showing key ideas and definitions. User defines a set of Propositional symbols, like P and Q. User defines the semantics of each of these symbols.

For example,

P means "It is hot"

Q means "It is humid"

R means "It is raining"

The alphabet of propositional logic contains the following symbols:-

1. The letters of the English alphabet; that is, A, B, C, ..., Z and each of these letters with an index (e.g A_3).
2. The logical values True and False.
3. These special symbols:

   ¬ (NOT)

   ∧ (AND)

   ∨ (OR)

   ⇒ (IF – THEN)

   ⇔ (IF AND ONLY IF)

   ( ) (GROUPING)

The symbol $\neg$ is called a unary connective and the symbols $\wedge$, $\vee$, $\Rightarrow$ and $\Leftrightarrow$ are called binary connectives.

The rules for creating propositions are as follows:

1. All letters, all indexed letters and the logical values True and False are propositions. They are called atomic propositions.

2. If P and Q are propositions, then so are $\neg P$, $P \wedge Q$, $P \vee Q$, $P \Rightarrow Q$, $P \Leftrightarrow Q$ and (P). They are called compound propositions.

The compound proposition $\neg P$ is called the negation of P, $P \wedge Q$ is called the conjunction of P and Q, and $P \vee Q$ is called the disjunction of P and Q.

Example: Suppose the propositions P and Q stand for these statement about the world:

P : It is raining outside.

Q : The street is wet.

Then the following compound propositions stand for these statements about the world:

$\neg P$: It is not raining outside.

$P \wedge Q$: It is raining outside and the street is wet.

$P \vee Q$: It is raining outside or the street is wet.

$P \Rightarrow Q$: It is raining outside, then the street is wet.

$P \Leftrightarrow Q$: It is raining outside if and only if the street is wet.

Q11(a) What is computable function? How is it useful and used? Explain with an example.

MDU BCA Nov. 2016

Ans. Computable functions are used to represent simple facts like greater than or less than. For example,

gt(12. 3)	lt(2, 12)
gt(5, 2)	lt(3, 7)
greater than	**less than**

It is useful to have computable functions as well as computable predicates. This helps to evaluate the truth of gt(7+3, 2).

To do this, the value of the plus function is computed first giving the arguments 7 and 3 and then these arguments 10 and 2 are sent to the gt function.

Consider the following set of facts:

1. Chaminda was a witch.
   witch (Chaminda)
2. Chaminda was a Ravansenapati.
   Ravansenapati (Chaminda)
3. Chaminda was born in 20 A.D.
   born (Chaminda, 20)
4. All witches are mortal.
   $\forall x : \text{witch}(x) \rightarrow \text{mortal}(x)$
5. All Ravansenapatis died when the war started in 39 A.D.
   $\text{Started}(\text{war}, 39) \wedge \forall x : [\text{Ravansenapati}(x) \rightarrow \text{died}(x, 39)]$

6. No mortal lives longer than 120 years.

$$\forall x : \forall t_1 : \forall t_2 : \text{mortal}(x) \wedge \text{born}(x, t_1) \wedge \text{gt}(t_2 - t_1, 120) \rightarrow \text{dead}(x, t_2)$$

7. It is now 2005.

$$\text{now} = 2005$$

8. Alive means not dead.

$$\forall x : \forall t : [\text{alive}(x, t) \rightarrow \neg\text{dead}(x, t) \wedge [\neg\text{dead}(x, t) \rightarrow \text{alive}(x, t)]]$$

9. If someone dies, then he is dead at all the later times.

$$\forall x : \forall t_1 : \forall t_2 : \text{died}(x, t_1) \wedge \text{gt}(t_2, t_1) \rightarrow \text{dead}(x, t_2)$$

Q11.(b) Explain "is a" and "instance" relationships.

Ans. Two attributes "is a" and "instance" play an important role in many aspects of knowledge representation. The reason for this is that they support property inheritance.

Knowledge can be represented as classes, objects, attributes and Super class and sub class relationships. In property inheritance, elements of specific classes inherit attributes and values from general classes in which they are included. For example, if we know that birds can fly and that canaries are birds, then we can infer that canaries too can fly.

Attribute "is a" is used to represent the relationship "Class inclusion" (super class, sub class relationship)

For example: is a (mega_star, rich)

Attribute "instance" is used to represent the relationship "Class membership" (element of the class).

For example instance (prince, mega_star)

Example: Represent the following facts:

1. Shreekant was a man.
2. Shreekant is Hindu.
3. All Hindus are Indians.
4. Shivaji is ruler.
5. All Indians were loyal to Shivaji.

Representing these facts in predicate calculus:

1. man (Shreekant)

2. Hindu (Shreekant)
3. $\forall x : \text{Hindu}(x) \rightarrow \text{Indian}(x)$
4. ruler (Shivaji)
5. $\forall x : \text{Indian}(x) \rightarrow \text{loyal to}(x, \text{Shivaji})$

Using "instance" attribute:-

1. instance (Shreekant, man)
2. instance (Shreekant, Hindu)
3. $\forall x$: instance (x, Hindu) $\rightarrow$ instance (x, Indian)
4. instance (Shivaji, ruler)
5. $\forall x$: instance (x, Indian) $\rightarrow$ loyal to (x, Shivaji)

Using "is a" attribute:-

1. instance (Shreekant, man)
2. instance (Shreekant, Hindu)
3. is a (Hindu, Indian)
4. instance (Shivaji, ruler)
5. $\forall x$: instance (x, Indian) $\rightarrow$ loyal to (x, Shivaji)

Q11.(c) Differentiate between propositional logic and FOPL.

Ans. Following points makes clear the difference between propositional logic and FOPL:

Propositional Logic	FOPL (First Order Predicate Logic)
1. Propositional logic assumes that there are facts (propositions) which are either True or False in a given world.	1. FOPL assumes that the world consists of objects with certain relations among them that are True or False in the given world.
2. Propositional logic is sound, complete and decidable.	2. FOPL is sound and complete but not decidable.
3. Propositional logic deals with simple propositions such as "I like cheese".	3. FOPL allows to reason about complex statements such as "All people who eat cheese like cats", using the quantifiers $\forall$ and $\exists$ ("for all" and "there exists").
4. Propositional logic is the simple foundation and fine for some AI problems.	4. FOPL is much more expressive and more commonly used in AI.

Q11.(d) Consider the following statements:

(i) Ram likes all kinds of food.

(ii) Apples are food.

(iii) Chicken is food.

(iv) Anything anyone eats and isn't killed by is food.

(v) Bill eats peanuts and is still alive.

(vi) Rita eats everything Bill eats.

Represent these in predicate logic.

Ans. (i) $\forall x$ food (x) $\rightarrow$ likes (Ram, x)

(ii) food (Apples)

(iii) food (Chicken)

(iv) $\forall x \forall y$ eats (y, x) $\wedge \neg$ is_killed (y) $\rightarrow$ food (x)

(v) eats (Bill, peanuts) $\wedge$ alive (Bill)

(vi) $\forall x$ eats (Bill, x) $\rightarrow$ eats (Rita, x)