

Types of knowledge

Declarative - includes facts & figure, rules

Procedural - steps, imperative knowledge, How to do something, rules, strategy, procedure, agendas.

Meta - knowledge about ^{other types of} knowledge

Heuristic - Specialized knowledge of some expert in a subject.

Structural - basic knowledge related to problem solving.
describes relationship b/w various subjects

Representation of knowledge

1) Logical Representation

rules \rightarrow logic \rightarrow procedure
 \rightarrow reasoning

Input \rightarrow Output

Types :- ① Prepositional Logic \rightarrow boolean \rightarrow 0 or 1
② Predicate Logic \rightarrow Declarative Sentence \rightarrow T or F

Compound proposition \rightarrow combining two or more propositions using logical characters or parenthesis

logical ~~set~~ characters:-

① Negation \neg

② Conjunction $P \wedge Q$

③ Disjunction $P \vee Q$

④ Implication \rightarrow

⑤ Biconditional \Leftrightarrow $P \leftrightarrow Q$

Conjunction \wedge

P	Q	$P \wedge Q$
True	T	T
F	T	F
T	F	F
F	F	F

Disjunction \vee

P	Q	$P \vee Q$
T	T	T
F	T	T
T	F	T
F	F	F

Implication \rightarrow

P	Q	$P \rightarrow Q \Rightarrow \neg A \vee B$
T	T	T
T	F	F
F	T	T
F	F	T

Bi-conditional \leftrightarrow

P	Q	$P \leftrightarrow Q$
T	T	T
T	F	F
F	T	F
F	F	T

PP → object

ACT → Action

PA → Attribute of objects

AA → Attribute of action

T → Time

LOC → location

Two way links b/w the actor (PP) & the action (ACT) is prepositional logic

In preposition logic, we represent statements which are facts but unfortunately PL is not sufficient to represent complex or natural language sentences i.e PL has very little expressive power.

e.g. "Some users are intelligent."
→ "Sachin likes cricket".

Predicate logic is also called FOL (First order Logic) or FOPL (First Order Predicate Logic).

First Order LOGIC is a powerful language that represents info about objects & also express relationship b/w these objects.

It is an extension of preposition logic.

It can be unary relation
e.g. → Sister of, brother of etc.

Syntax of FOL represents collection of symbols

Basic elements of FOL:

- i) Constant i.e. any value - John, Ram, Market
- ii) Variable - x, y, A, B, a, b
- iii) Predicate - brother, eat, father
- iv) Function - sqrt, left of
- v) Connectives - \wedge , \vee , \neg , \Rightarrow , \Leftarrow , \Leftrightarrow
- vi) Equality - $=$

① Atomic Sentence

→ Ram and Shyam are brothers = brothers (Ram, Shyam)

predicate (term₁, term₂ - "terms")

② Compound Sentence

→ Combining atomic sentence using connectives

→ can be divided into:

* Subject - main part of sentence

* predicate - a relation which binds two object together in one statement.

Quantifier in post order logic → tells scope

→ It is a language element which generates quantifier.
(Quantity of specimen)

→ These are the symbol that permit to determine the range & scope of the variable in a logical expression.

① Universal Quantifier → symbol of logical representation

forall that the statement within its range is true for everything.

→ always used with implication (\rightarrow)

If x is a variable \forall (for all) ~~\exists~~ x , for each x .

Eg \rightarrow ① All man drink coffee

$$= \forall x \text{ man}(x) \rightarrow \text{drinks}(x, \text{coffee})$$

② Every student is intelligent.

$$\forall x \text{ student}(x) \rightarrow \text{intelligent}(x)$$

③ Every Indian likes national anthem.

$$\forall x \text{ Indian}(x) \rightarrow \text{likes}(x, \text{national anthem})$$

④ Existential Quantifier (\exists)

\rightarrow statements within its core is true
for at least one instance of substance.

Note - Existential quantifier we always use AND
or conjunction (\wedge) $\rightarrow A \wedge B$

There exist a x , for some x , for at least one x

Eg \rightarrow i) ~~These~~ Some boys are intelligent

$$\exists x \text{ boy}(x) \wedge \text{intelligent}(x)$$

ii) Some girls are beautiful and intelligent (x)

$$\exists x \text{ girl}(x) \wedge \text{beautiful}(x) \wedge \text{intelligent}(x)$$

iii) Every man respects his parents

$$\forall x \text{ man}(x) \rightarrow \text{respect}(x, \text{parents})$$

iv) All birds fly.

$$\forall x \text{ bird}(x) \rightarrow \text{fly}(x)$$

v> Some boys play cricket

vi> Not all students like both maths & science

vii> Only one ~~student~~ student failed in math.

viii> Every likes icecream.

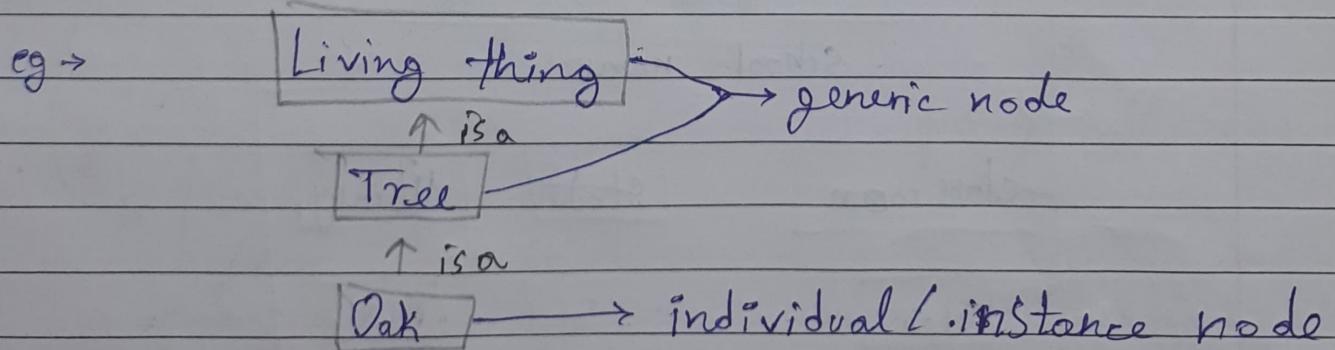
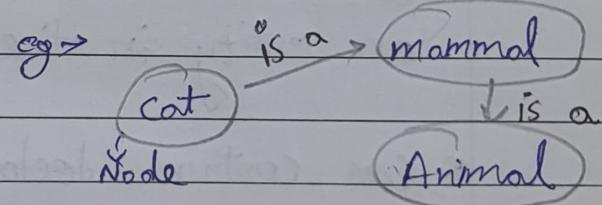
ix> There is no one who doesn't like icecream.

knowledge representation type 2 continuation..

- 2) Semantic Networks: [alternate to predicate logic]
- Graphical representation
 - Network consists of nodes (object) & arcs (relationship)

Representation consists of two relations:-

- i) is a relation
- ii) kind of relation



- predicate logic → $\forall x (\text{trees}(x) \rightarrow \text{living thing}(x))$
universal quantifier for all trees are living things
- $\forall x (\text{cat}(x) \rightarrow \text{mammal}(x))$

Advantage of semantic network:

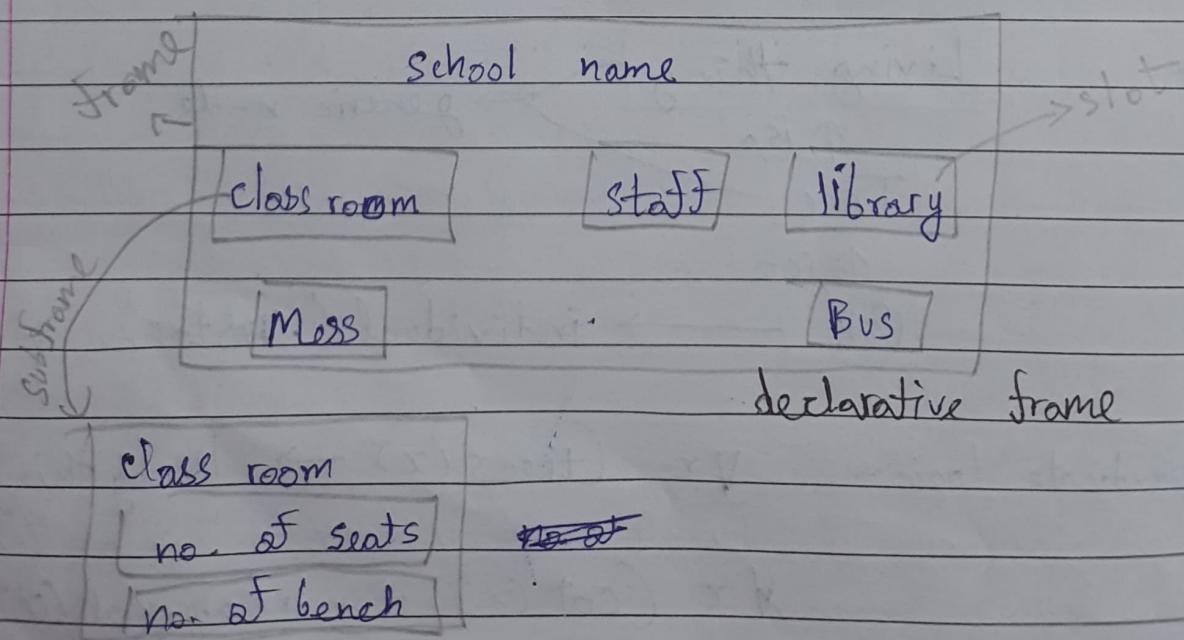
- * It represents knowledge graphically
- * It is simple & easy to understand.
- * It conveys meaning in transparent manner.
- * Easily translated to prologue language.

Drawbacks of semantic network:

- * It takes more computation time at runtime as we need to traverse the complete tree to answer questions.
- * Semantic networks do not have standard definition for the link names.
- * These networks are not intelligent & depends on the creator of network (is not self-sufficient)

3) Frame Representation → like objects

- * Frame is a record like structure which consists of collection of attributes & its value to describe an entity in the world.
- * Can contain declarative knowledge or procedural knowledge.



* Slots can be filled by values, procedures, pointers to other frames

* Types of frames:

- i) Declarative frames → details / description / info about object
- ii) Procedural frame → How to perform an activity

Action frame

Types of slots in action procedural frame

→ Actor slot

→ Source slot → task slot

→ object slot

→ destination slot



eg →

Saving new file in computer

Actor

Object

user

~~object~~ file

Source

destination

comp

comp

Task 1

Task 2

Create a new
File

click on source

Task 3

choosing ~~location~~

procedural frame

Creating a file

Actor

User

Object

file

Source

Comp

destination

Comp

Task 1

create a
blank file

Task 2

write on
blank file

Advantage of Frames:

- * can add slots easily
- * makes program easier by grouping the data
- * Easy to understand & visualize
- * Comparably flexible & used by many AI applications

Disadvantage of Frames:

- * Inference mechanism can not be smoothly proceeded by frame representation.
- * It has a much generalized approach.

4) Production Rules

- * Consists of (condition, action) i.e if condⁿ then action
- * ~~IDE~~ has mainly three parts :
 - i> Set of production rules.
 - ii> Working memory
 - iii> Recognize act cycle

- Production rules are expressed in natural language
- Production rules are highly modular, we can easily remove, add & modify an individual rule.
- Doesn't perform learning as it doesn't store result for future use.
- During execution of program many rules are active hence rule based production system are inefficient.

5) Script

* Frame like structure

* also contains slots. In each slot there is info about slots, what can happen in particular situation, what event can occur.

eg → going to market, purchasing, paying bill

→ going to restaurant, ordering food & paying bill

Components of Script:-

i) Entry Conditions

ii) Results - condition after event has occurred.

iii) Slots representing objects involved in event.

iv) Roles & actions to be performed by user

v) Track → specific instance

vi) Scene → sequence of events

Types of relationship:-

i) is - are

ii) ~~is~~ kind of

BATRANS token

eg → Script → withdraw money

Script for going to bank & withdraw money

Role - user, cashier, employee

Entry condition = P have no money or less money

The bank is open

⇒ money

Results → P have no money.

Scene 1 :- Entry / Entering

- 1> P PTRANS (go) P into Bank
- 2> P ATTEN (watch) eyes into E
- 3> P MOVES (walk) P to E.

Scene 2 :- Filling form

- 1> P MTRANS signal to E
- 2> E ATRANS (pull) form to C
- 3> E ATRANS token to P

*

Scene 3 :- withdrawing money.

- 1> P ATTEND eye to counter.
- 2> P PTRANS token to C.
- 3> C ATRANS money to P.

- ① A-trans / ATRANS :- Transfer ^{of abstract} relationship (give)
- ② PTRANS :- Transfer physical location (go)
- ③ PROPEL :- Applying physical force to an object (push)
- ④ MOVE :- Move a body part by owner (kick)
- ⑤ MTRANS :- Transferring mental info (Tell)
- ⑥ ATTEN :- Focus sense organ to specific thing
(listen, watch)

Advantage:-

- > Ability to predict events
- > A single coherent interpretation ^{to build} from collection of observation.

Disadvantage:-

- > less general than frames

2) May not be suitable for all levels of knowledge

Conceptual dependency was originally developed to represent knowledge acquired from natural language.

To be independent of word used in original input i.e. independent of language in which it was initiated.

For any two sentence identical in meaning their should be one represented in meaning.

ATRANS :- Give/Take = Abstract Transfer

PTRANS = Go/come = Physical Transfer

MTRANS = Tell/ask = Mental Transfer

PROPEL = Push/Pull

MBUILD = Decide, inferring = Mentally Building new into old.

SPEAK = Say, talk, Murmuring = speaking by producing sound

ATTEND = listen, watch, touch, hearing

MOVE = Punch, Kick,

GRASP = clutch snare clutch/catching

INGEST = to eat, drink

EXPEL = cry, sweat

Script - Robbery [Track : Successful Snatch]

Props :

G = GUN

L = Loot money

B = Bag

C = Car

Roles :

R = Robber

M = Cashier

B = Bank Manager

P = Police men

Scene I: Getting a Gun

R PTRANS R into Gun shop
R MBUILD R choice of G
R MTRANS choice
R ATRANS Guys G

Scene II: Holding up Bank

R PTRANS R into Bank
R ATTEND eyes M, O and P
R MOVE R to M position
R GRASP G
R MOVE G ~~to~~ to point to M
R MTRANS "Give me money or else"
P MTRANS "Hands up" to R. ^{ton}
R PROPEL shoots G
P INGEST bullet from G
M ATRANS ~~L~~ to ~~G~~ R
~~R~~ ATRANS L puts in B
R PTRANS EXIT
S ATRANS RAISES Alarm

Scene III: ~~The~~ The Gateaway

R PTRANS C