L60: Forward chaining, Backward chaining | Example | Comparison | Artificial Intelligence Lectures

Easy Engineering Classes — GGSIPU, UPTU, Mumbai Univ., Pune Univ., GTU, Anna Univ., PTU and Others

Ques:- Differentiate between forward and Backward ~~chaining~~ Reasoning.

Forward Reasoning:- It moves forward from start to goal state. Also called as Data Driven Reasoning.

↳ Search Tree has Initial Configuration(s) at root of tree.

↳ Next Level of Tree is generated by finding all the rules whose left side matches the root node.

[IF-THEN RULES]

[Medical Diagonsis]

if S1 → C1

↳ If C1 → C2

↳ If C2 → D1

① Initially System is provided with one or more constraint.

② Rules are Searches for each Constraint.

③ Satisfying rules (R.H.S) becomes L.H.S

Backward Reasoning:- It moves backward from goal to initial state. Also called as Goal Driven Reasoning.

↳ Search Tree has Goal Configuration(s) at the root of the tree.

↳ Next Level of Tree is generated by finding all the rules whose right side matches root nodes.

D1 → S1

① Goal state and rules are selected where Goal state resides in THEN part.

② from If part of Selected rules Subgoals are made

7:14 / 7:18

(AI-11) Easy Engineering Classes – Free YouTube Lectures

Ques:- What are Semantic Networks? Explain with the help of an example.

↳ Graphical notation for representing knowledge in interconnected nodes pattern.

↳ Popular in AI and NLP ∵ it represents knowledge or support reasoning.

↳ Alternative of Predicate Logic

↳ Nodes represents objects → circle (Physical), Ellipse (Concept), Rectange (Situation)

↳ Arcs represents Rel^n b/w objects] links/arrows

↳ Link Labels] specify relationships

↳ Also known as Associative Nets] nodes are associated.

COMPONENTS:

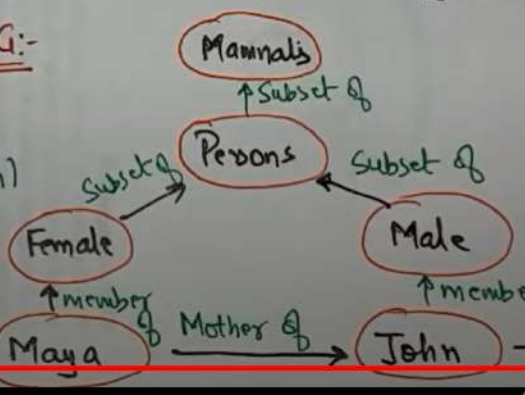
① Lexical Components: Nodes, Links, Labels.

② Structural: Link or Nodes] Directed.

③ Semantic:] Def^n related to links/nodes.
↳ Facts.

④ Procedural: → Constructor] creation of new links
→ Destructor] removal of link.

EG:-

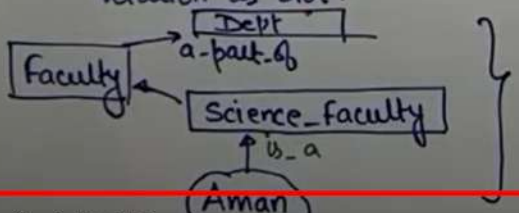


Representing Knowledge using FRAMES:-

FRAME is a Collection of attributes or (Slots) and associated Values that describe some real-world entity.

↳ uses datastructure (records) to represent the knowledge represented in Semantic N/w.

↳ Each frame represents the node in Semantic N/w as a class, or an instance and each relation as slot.



Attributes attached with each slot:-

1. Instance: Relates Slot with Class
2. Definition: slot defⁿ/ Value.
3. Default: Slot default Value.
4. Domain: Slot elements domain
5. Range: Specifies class of which elements.
6. Range Constrain: Logical expression] $6 < i < 10$
7. To-Compute: Value of Slot is to be computed.
8. Single-Valued: funcⁿ returns Single Value.
9. Inverse: Slot inverse used in reasoning.
10. Transfers -through: Inheritance.

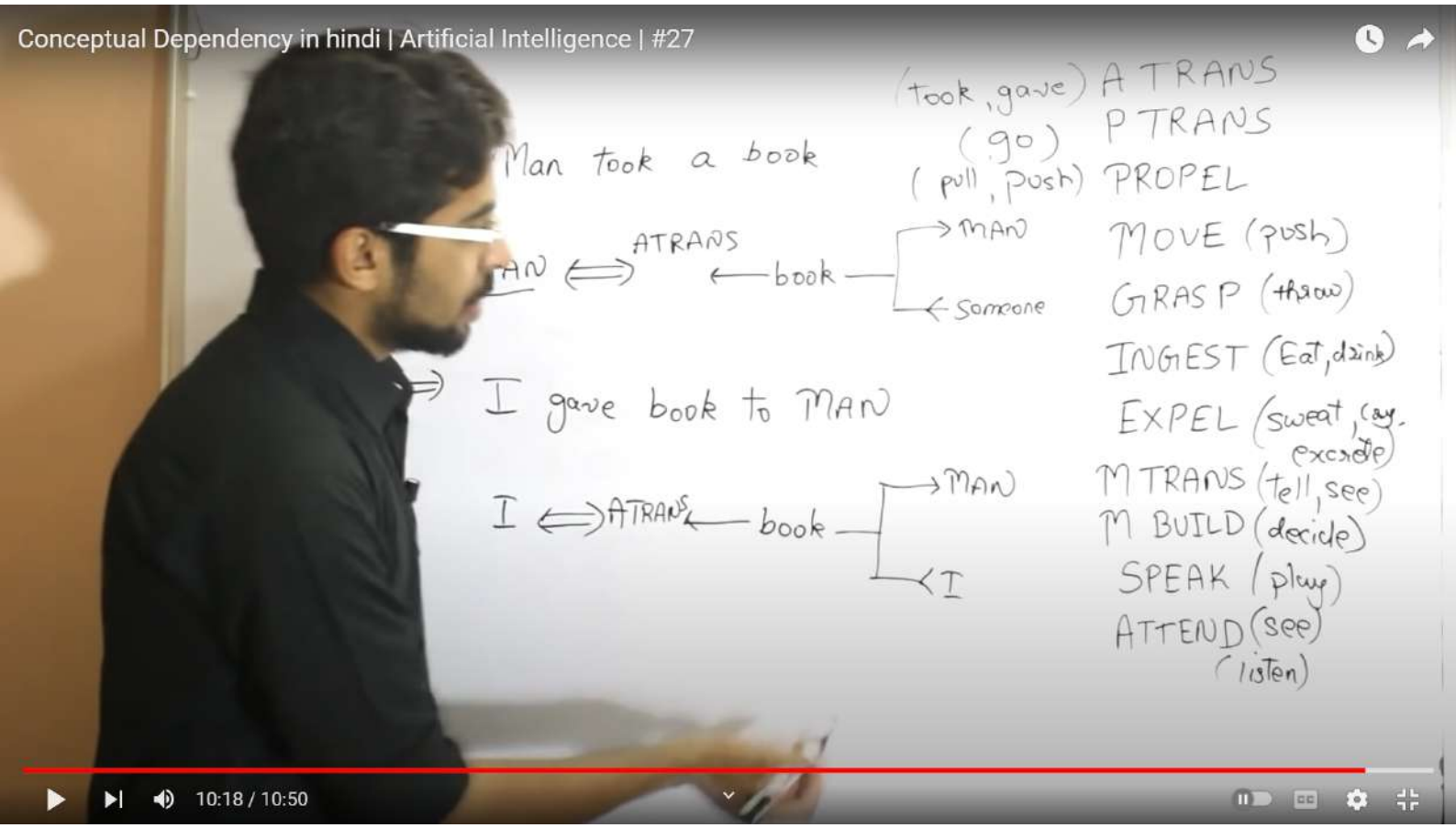
L71: Frames for knowledge Representation in Artificial Intelligence | Attributes, Reasoning Actions

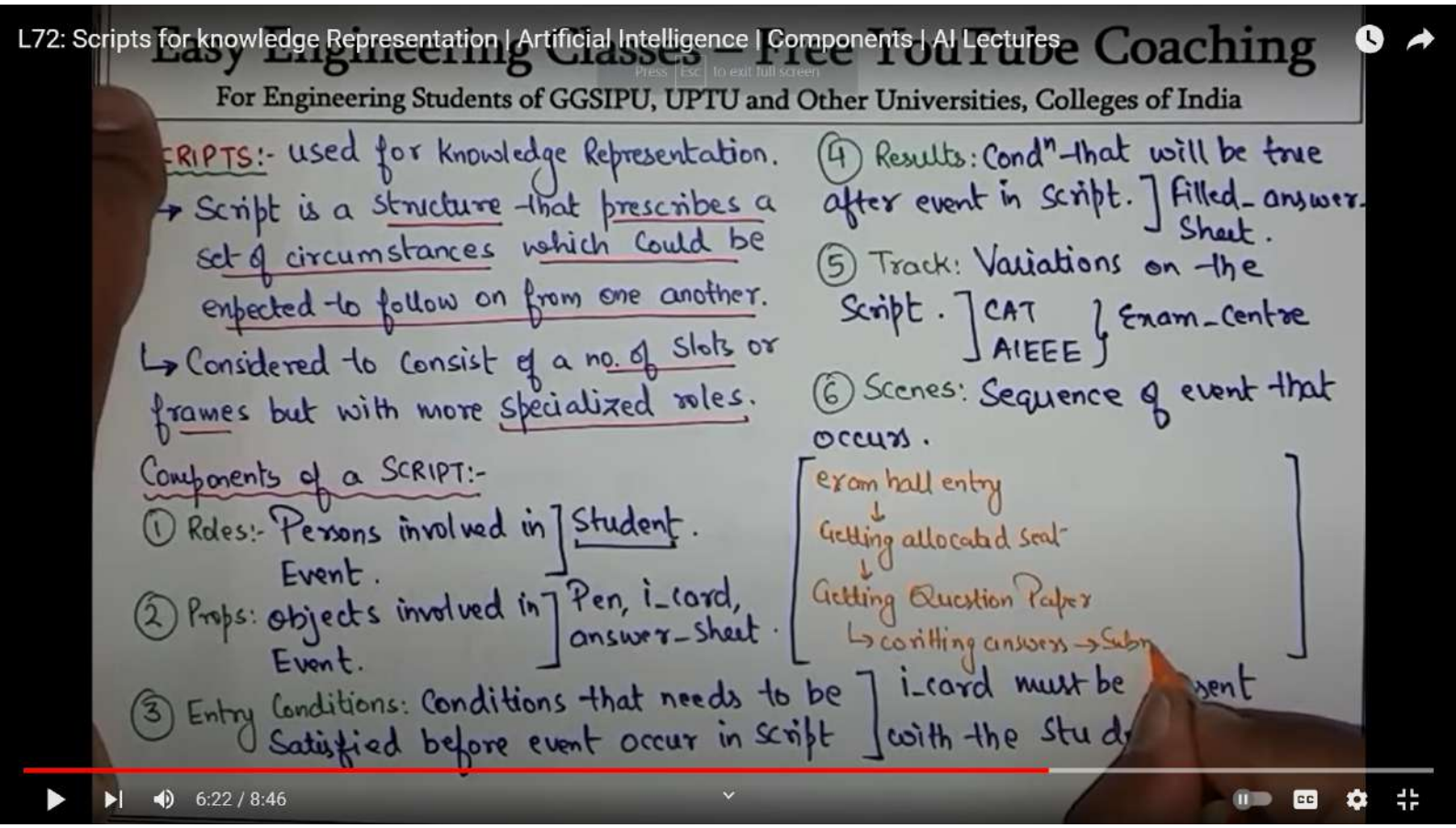
Easy Engineering Classes – Free YouTube Coaching

For Engineering Students of GGSIPU, UPTU and Other Universities, Colleges of India

Reasoning actions that can be performed frames:-

1. Relating the definition: isa, inverse links] Propagation of def^n to relate all info^n.
2. Inheritance: Inherited all values including default values of the slot.
3. Legality of Value: Checks the legalality of slot value.] Range constraint. ✓
4. Consistency check: Verifying slot value consistency before adding to the frame.] Domain ✓
5. Maintaining Consistency: when one slot is updated its inverse should also be updated.
6. Computation of a Value of Slot: to compute transfers through.


Conceptual Dependency in hindi | Artificial Intelligence | #27
Man took a book
MAN ⟺ ATRANS ← book → MAN
← someone
⟹ I gave book to MAN
I ⟺ ATRANS ← book → MAN
← I
(took, gave) ATRANS
(go) PTRANS
(pull, push) PROPEL
MOVE (push)
GRASP (throw)
INGEST (Eat, drink)
EXPEL (sweat, cry, excrete)
MTRANS (tell, see)
M BUILD (decide)
SPEAK (play)
ATTEND (see) (listen)
10:18 / 10:50


L72: Scripts for knowledge Representation | Artificial Intelligence | Components | AI Lectures
Easy Engineering Classes - Free YouTube Coaching
For Engineering Students of GGSIPU, UPTU and Other Universities, Colleges of India
SCRIPTS:- used for knowledge Representation.
→ Script is a structure that prescribes a set of circumstances which could be expected to follow on from one another.
↳ Considered to consist of a no. of slots or frames but with more specialized roles.
Components of a SCRIPT:-
(1) Roles:- Persons involved in Event.] Student.
(2) Props: objects involved in Event.] Pen, i_card, answer_sheet.
(3) Entry Conditions: Conditions that needs to be Satisfied before event occur in script] i_card must be with the stu
(4) Results: Cond" that will be true after event in script.] Filled_answer_Sheet.
(5) Track: Variations on the Script.] CAT, AIEEE } Exam_Centre
(6) Scenes: Sequence of event that occurs.
exam hall entry
↓
Getting allocated seat
↓
Getting Question Paper
↳ writting answers → Subm
6:22 / 8:46

Propositional Calculus:-

↳ It is a system that deals with the method used for manipulation of the symbols according to some rules.

X: It is Hot
Y: It is Humid
Z: It is raining

{ All, Some → (PL) x }

ALPHABET SET:

↳ i) Set of Variables or Propositional Symbols P, Q, R
ii) Logical Constants → True (T), False (F)
iii) Two Parentheses "(" and ")" (X, Y)
iv) Set of logical operators

IMP:- ↳ i)

coord	symbol.	example
not	¬	¬X
ii) and	∧	X∧Y
iii) or	∨	X∨Y
iv) implies	→	(X→Y) if X then Y
v) if and only if	↔	(X↔Y)

(1) if it is humid ('Y') then it is hot. (X)
↳ (Y → X)

(2) if it is hot (X) and humid (Y) then it is not raining (Z).
(X∧Y) → ¬Z

v) Set of equivalence relations or laws: (P, Q, R) are Variables.

- Commutative Laws: $P \wedge Q \cong Q \wedge P$, $P \vee Q \cong Q \vee P$
- Associative Laws: $(P \wedge Q) \wedge R \cong P \wedge (Q \wedge R)$, $(P \vee Q) \vee R \cong P \vee (Q \vee R)$
- Double Negation: $\sim(\sim P) \cong P$
- imp De-Morgan's Law: $\neg(P \vee Q) = \neg P \wedge \neg Q$, $\neg(P \wedge Q) \cong \neg P \vee \neg Q$
- Absorption Law: $P \wedge (P \vee Q) \cong P$, $P \vee (P \wedge Q) \cong P$
- Law of Contradiction: $P \wedge \neg P \equiv$ False $\begin{matrix} P=1 \\ \neg P=0 \end{matrix}$ ⓪ $\begin{matrix} P=0 \\ \neg P=1 \end{matrix}$ ⓪
- Law of excluded middle: $P \vee \neg P =$ True $\begin{matrix} P=1 \\ \neg P=0 \end{matrix}$ ① $\begin{matrix} P=0 \\ \neg P=1 \end{matrix}$ ①
- Law of impotency: $P \wedge P \cong P$ $\left[\begin{matrix} P=1 \to 1 \\ P=0 \to 0 \end{matrix}\right]$

RULES OF INFERENCE:

→ ① MODUS PONENS: If 'P' and 'P→Q' is given to be true, then we can infer that 'Q' is true.

P: It is a holiday ✓ (T)

Q: The school is closed] → we can infer that it is true.

P→Q: If it is a holiday, then School is closed ✓ (T)

② MODUS TOLLENS: If '~Q' and '~P→~Q' are given to be true, then we can infer that ~P is true.

~Q = School is not closed.

(~P→~Q) → if it is not a holiday, then School is not closed.

→ ~P] it is not a holiday (True).

Representing Simple Facts in FOPL:

→ models world in terms of objects.

Real-world facts can be represented as logical propositions written as well-formed formulas in propositional logic.

Symbols:-
1. objects
2. Properties
3. Relation.

} Representation.

Symbols are formed of the following:-

1. Set of all uppercase english Alphabets.
2. Set of digits from 0 to 9.
3. Underscore character.

[All dogs are brown] Can't be written in propositional logic

↳ first-order predicate logic.

Defining a Sentence :-

Every atomic sentence is a Sentence.

↳ is defined as predicate constant of arity 'n', followed by $t_1, t_2, \ldots t_n$ terms enclosed in parentheses and separated by commas.

1. if 'S' is Sentence then $\neg S$ is sentence.
2. if S1, S2 are sentences] $S1 \wedge S2$ (Conjuction)
3. " "] $S1 \vee S2$ (disjunction)
4. " "] $S1 \rightarrow S2$ (implication)
5. " "] $S1 \equiv S2$ (equivalence)
6. if x is Var. and s is Sentence the $\forall x$ s is a sentence.
7. " " $\rightarrow \exists x$ s is a Sentence.