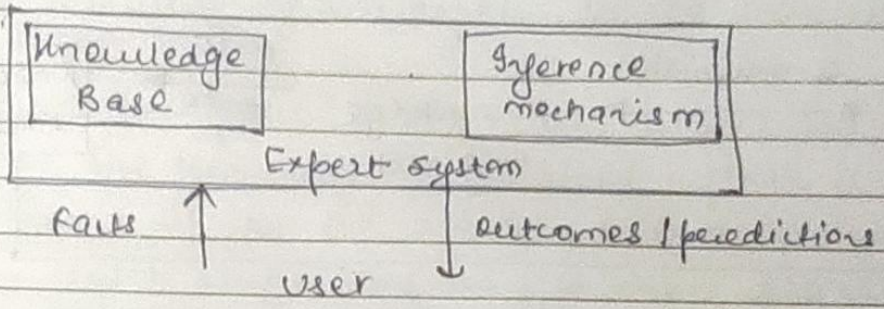
Unit-5

* Architecture of Expert System :-



→ The expert system needs to have a strong knowledge base along with an inference mechanism.

→ Types of knowledge that KB in expert system contains (i) Factual knowledge (ii) Heuristic knowledge

Knowledge that is obtained from human or some sort of text or other form.

knowledge based on prediction & experience that helps in good prediction.

→ The inference mechanism is responsible for getting the conclusion with the help of KB.

→ The ES processes the rules and gives the outcome.

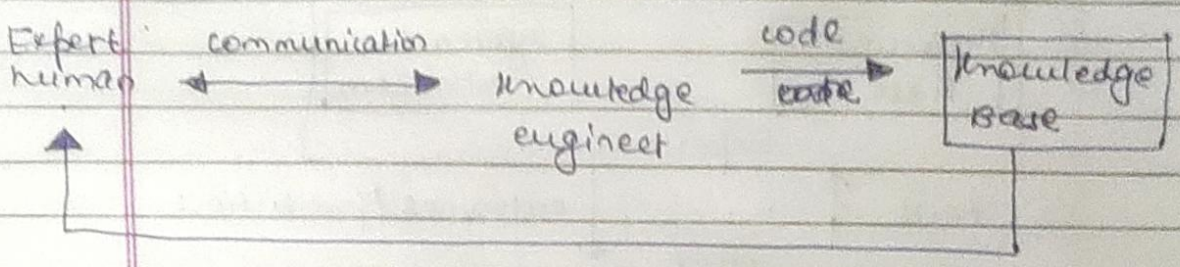
→ In ES we can relate 2-types of knowledge

(i) Problem knowledge specific to the type of problem

(ii) Domain knowledge how about solving a specific problem.

→ Domain knowledge forms subset of problem knowledge and both are influential in the formula of KB.

* How can expert system be designed ?

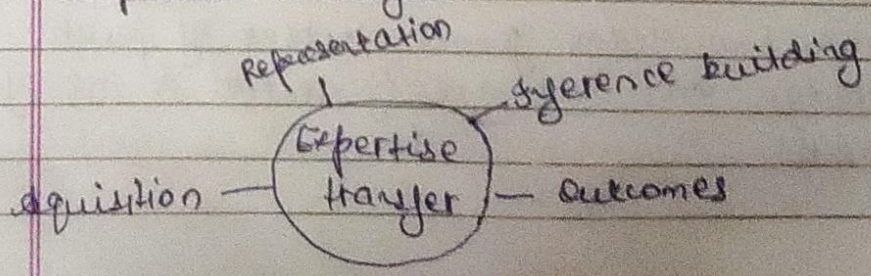


→ In first phase, the knowledge engineer sets up communication with human expert. This helps to get max^m knowledge for inferring.
→ In the next phase the knowledge engineer codes in the knowledge for KB explicitly.
→ The evaluation for such system is carried out by the expert.
whose inputs helps in building better system.

* Parameters in Building an Expert System :-

① Domain Exploration :-
→ concerned with the exploration of knowledge about the domain in which system is operational.
→ Here, objects & their relationships to the domain are identified along with the other relevant info. that is essential from the KB building perspective.

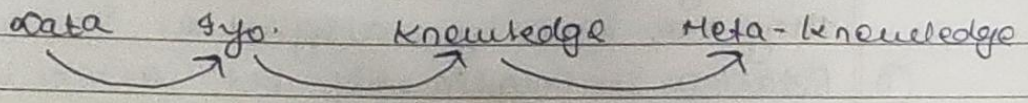
② Expertise Transfer :-



→ In ES, this expertise takes place from acquiring the knowledge from expert to the transfer of it to the user.
→ This takes place during data acquisition phase.
→ Once the knowledge is obtained, there is transfer to represent it for the system to understand.
→ At later stages the transfer takes place to build an inference and finally to have outcomes transferred back to the user.

expertise in form of knowledge.

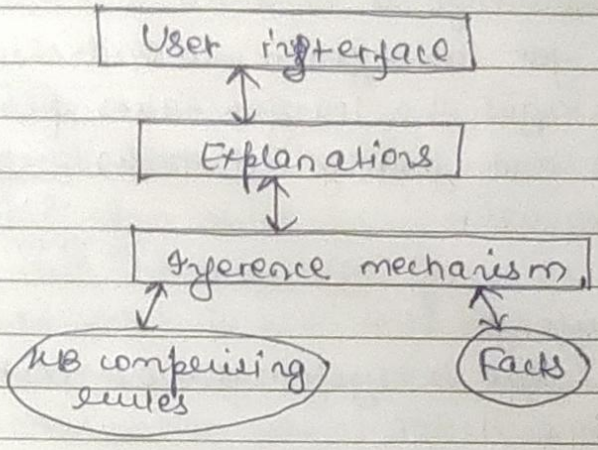
③ Meta Knowledge :-
→ Meta knowledge is systematic as well as domain independent knowledge.
→ This helps in building a system that is less dependent on a particular domain, helping in keepin a generalized view.



* Rule Based Expert System :-
→ Rule-Based Expert System consist of rules (if-then) set of facts and an interpreter.
→ Rule-Based ES are essentially designed so that they can have heuristic reasoning and proper explanations. They can seperate the knowledge from reasoning and facts.
→ In the figure, the inference mechanism, often referred to as inference engineer seeks the necessary info. and relationships from the KB to answer.

→ Two types of inference mechanism are used
(i) Forward chaining (ii) Backward chaining

→ Explanation make the user understand how a particular conclusion has arrived.



→ Rules fall under certain semantics, which tells us that the production rule have some meaning.
They are broadly classified as :-

① Relation :- Relationship bet? antecedent and consequent is specified.
e.g, water is boiling, then water is hot.

② Recommendation :- a rule that directs or gives some advice. For e.g, if the water in glass is hot, then advice is not to touch the glass.

③ Strategy :- Tells what needs do be checked, advice regarding action for betterment.
e.g, if road 1 is blocked then try road 2.

④ Directive :- refers to indication of some action needs to be carried out. For eg, if the light is green, then you can proceed

(5) Heuristic :- states guideline-based rule for e.g. if there is severe pain in chest & difficulty to breath then the diagnosis might be as heart attack.

→ Since the Rule-Based Expert System is based on rules, so there can be conflicts
→ conflict can arise with regard to select the rule or w.r.t to consequent mapping.
→ Handled by :-
(i) One way is to prioritise the rules.
(ii) Sequential way to trigger the rule. triggering the longest rule in case of conflict.

* Pros of Rule-Based ES :-
(i) Effective because of the way knowledge is represented which enables rum reasoning.
(ii) Can handle uncertain data as well.

* Cons :-
(i) It deals with huge amt of data so, managing and updating KB is complex.
(ii) Conflict problem due to huge data size.
(iii) Rigid behaviour of system which makes it useless outside the scope of the domain.

* FRAME-BASED EXPERT SYSTEMS :-
→ A frame is a data str with a typical knowledge about object or the concept.
→ Frame has its name & set of attributes.

→ For eg, a car frame can have type, colour as set of attributes / slots.

→ Each slot has unique value associated with it so, we can say frame gives a concise representation.

→ Hence, a frame-based representation is suited for object-oriented programming techniques.

→ In AI terms, Knowledge engineer calls objects as the frame.

→ we can have following things in the slot.
① Frame name
② Relationship with other frames
③ Values or ranges
④ Procedural info.

→ By relationship we mean hierarchy, to be specific inheritance is depicted.

→ Procedural info., the slot is attached to a procedure that is executed when any event is triggered such as change in value of slot.

→ Generally instance frame-refers to an object and class-frame refers to a group.

class : car
make
Type
colour

class : car instance : i10
make : Hyundai
Type : Small
colour : Red

class : car instance : Verna
make : Hyundai
Type : Sedan
colour : Blue

→ Aggregation, generalization and association concepts apply there as well.

→ Apart from the knowledge part, Expert system needs to have some methods too.

∴ Methods and demons are added to frames

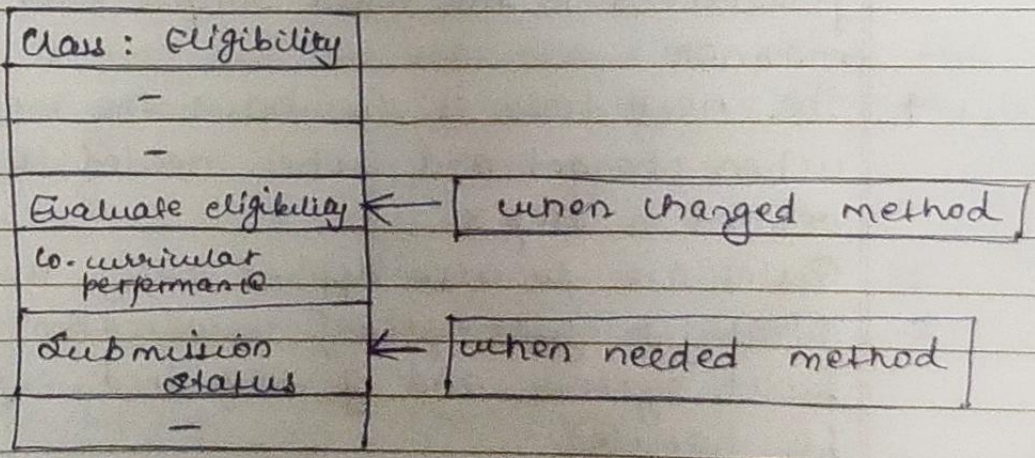
→ procedure that is executed when requested — used for more complex things

→ Two categories :-

(1) a "when changed method" is executed when value changes

(2) a "when needed method" is ~~executed~~ used to obtain the value for the attribute.

* Working of a Frame-Based Expert System :-



e.g. to check the eligibility of a student appearing for a exam. We have different slots as well as methods & demons for the frame

of check eligibility, request action and so on.

→ if when changed method is invoked when the expert system is asked for the evaluation status.

→ when needed method is invoked by the system to check the submission status of that particular student.

* Guidelines to build a Frame-Based Expert System :

1. The first step involved in designing any expert system is the scope and the specificity of the problem at hand. There has to be utmost clarity about these two factors.

2. Next step is identification of classes, instances and the attributes.

3. The display of the system has to be presented in the most simple and transparent manner.

4. The next task is to share the methods of when changed and when needed which is decided in step 3.

5. Rules are to also defined along with methods.

6. Finally a full-proof evaluation of the built system and if required, expansion should be handled.

* **Case Study :-**

→ Here we discuss - Perospector an expert system in geology domain.

* **Aim of the Expert System**

→ The aim of this expert system is to evaluate the mineral potential of geological region. It handles geological settings, structural controls and diff. types of rocks as well as minerals.

* **Target user :-**

→ Target user are the geologists who are at early stage of investigating the site to be a drilling site.

* **Key Points :-**

→ Perospector can handle uncertain data and incomplete data. It is also domain independent system.

→ Data is matched to the models to check the presence of ore so as to identify the site to be a drilling site.

* **KB :-**

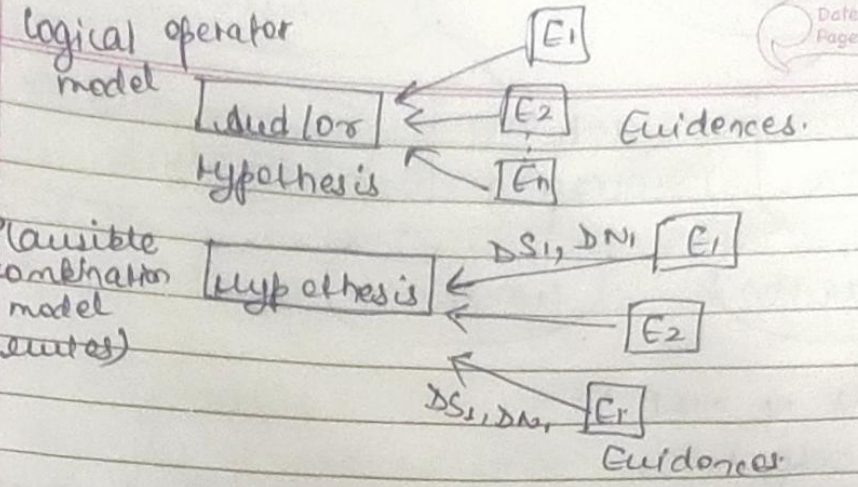
→ KB of perospector is divided into 2 parts

- General KB: Bg info. that can be used yet other appl^{n} as well
- Special KB: related to the domain

* Methodology :-
→ Perospecter makes use of production rule and semantic n/ws. also makes use of backward chaining strategy.

* Representation :
→ The representation used is inference n/w.
→ Inference n/w is a collection of evidence and hypothesis.
→ 'E' represents evidence and 'H' represents hypothesis.
→ There exists linkage bet: evidences. The hypothesis has some degree that is pre-stored or ranged.
→ The value of E & H changes with the help of Baye's Theorem in course of execution.

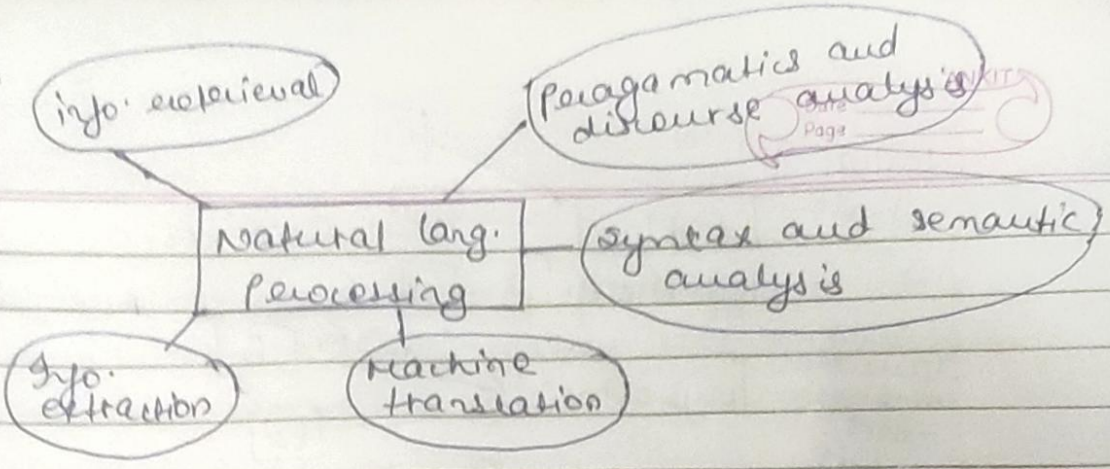
* Inference Mechanism :-
→ ∴ it has to handle uncertainity, probability has to come in picture.
It makes use of Bayesian, certainity factors and fuzzy sets.
→ The measure of sufficiency is that is used when the evidence E is known to be existing, whereas measure of necessity is used when E is not known to be existing.



DS: Degree of sufficiency
DN: Degree of necessity.

Sample models used in Perspector

* NLP [Natural Language Processing]
→ Natural language Processing refers to AI method of communicating with an intelligent systems using a natural language.
→ Processing of natural language is required when you want an intelligent system like robot to perform as per your instructions.
→ The field of NLP involves making computers to perform useful tasks with the natural languages human use.
→ The i/p and o/p of an NLP system can be -
  • Speech • written text



* Levels of NLP :-

1. Morphology :-

→ Analysis of individual words that consists of morphemes - the smallest gramatical unit.

→ Words with 'ing' 'ed' changes meaning of the word.

→ The analysis becomes necessary in the determination of tense as well.

2. Syntax :-

→ Syntax is concerned with the rules. It includes legal formulation of sentences to check the structure.

e.g, "Hari is good not to"; The sentence it is totally invalid here.

3. Semantic :-

→ Meaning check is carried out. The way in which meaning is conveyed.

For e.g, "The table is on the ceiling". This is syntatically correct but semantically wrong.

4. Discourse Integration :-
→ In communication or even in text formats, often the meaning of current sentence is dependent on the one that is prior to it.

5. Pragmatics :-
→ Analysis of the response given from the user with reference to what actually the lang. meant to convey is handled.
→ For eg. "Do you know how long it will take to complete the work"?
→ The expected answer is no. of hours rather than a yes or no.

6. Prosody :-
→ Handles rhythm. This is the most difficult analysis that plays an imp. role in poetry or shlokas.

7. Phonology :-
→ Involves analysis of diff. kinds of sounds that are combined. Concerned with speech recognition.

* SYNTACTIC AND SEMANTIC ANALYSIS
→ Morphology deals with word formation from small parts.
→ Morphological analysis is simply separating or tokenising the words. Removal of additional words like 's', 'ed' and assigning them to the syntactic category is carried out in morphological analysis.

* Syntax Analysis :-

→ Syntax analysis is concerned with checking the syntactic structure and its component.

→ Once, the words of a sentence are categorized into diff. parts of speech, the syntax analysis generates a parse tree

→ A parse tree is a representation for the sentences into a structure.

→ Grammar is used to produce a parse tree. In NLP, grammar is description of lang. It comprises rules.

→ There are diff. types of grammar. Lets discuss CFG (Context Free Grammar)

S → SB VP OB

SB → PN

VP → ADV V|V

OB → S1

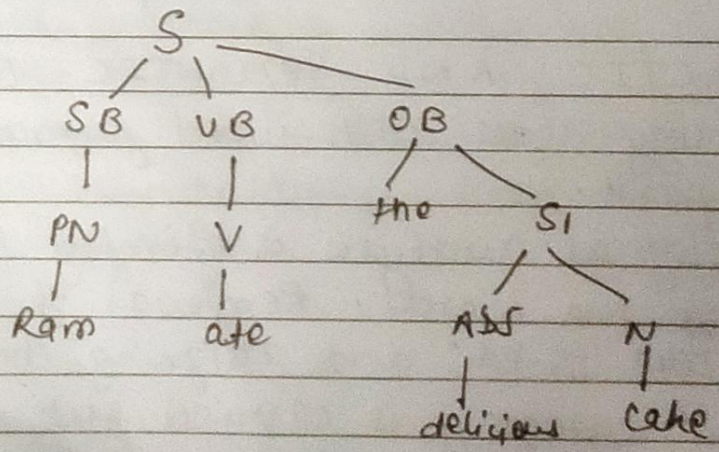
S1 → ADJ|N

PN → Ram

ADJ → interesting | delicious

N → book | cake

V → read | ate



Top down Parsing begins with the start symbol in the rules and keeps on applying the rule till the terminal become the symbol of the sentence.

Bottom-up-Parsing the rules are applied backward by replacing the words in the sentence with the appropriate rules till the start symbol is reached.

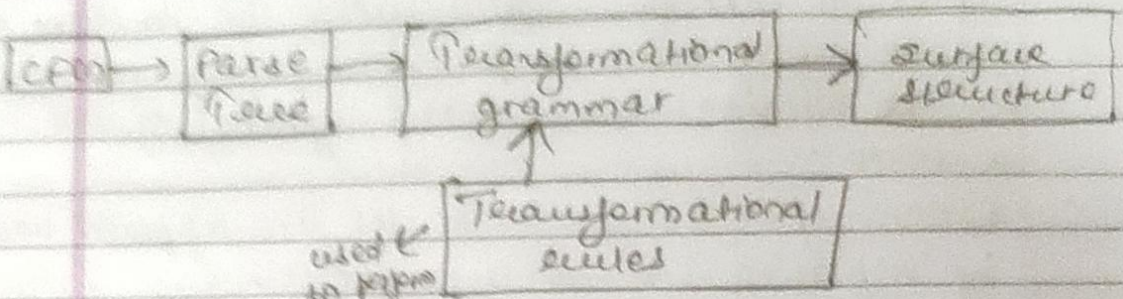
→ Though top-down parser can build trees without the time factor making a constraint, so it is possible that the trees are produced without having a look at what the next symbol or token in the sentence is to be parsed. So many wrong rules are selected.

→ The branching factor is also a point of concern in both the approaches.

* Transformational Grammar :-

→ This grammar is an extension to the CFG. So, the parse tree generated by the CFG is further defined with addⁿ of tags and re-constructing.

→ The tree generated by CFG is called deep structure and the one with transformational grammar is called surface structure. The reason behind is that it consists of additional tags that enable to understand the context-sensitive entities.



* Augmented Transition Network (ATN)

→ To check the correctness of sentence gramatically

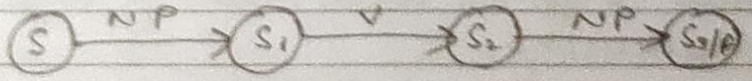
→ RTN (Recursive Transition n/w) is core concept for ATN.

→ RTNs are based on finite state automata, but are equivalent to push down automata.

→ An RTN is a directed graph, which consists of states and arcs that are labelled.

→ It takes input sentence and tells whether the string is acceptable or not.

→ The arc represent the labelled category or can even refer to other n/ws.



→ An ATN differs from RTN. So ATN it is essentially an RTN that has set of test rules which needs to be satisfied prior to the arc traversal.

→ While doing so, it makes use of register to store the states.

NP: Noun Phrase N: Noun
V: verb
PN: Proper noun
Det: Article

* Semantic Analysis :-
→ It attempts to understand the meaning of natural lang.
→ Different parts of semantic analysis :-
1. Lexical Processing :-
→ lexical semantic analysis involves understanding the meaning of each word of the text individually.

2. Compositional Semantics analysis :-
→ Sentence 1 :- Student love geeks for geeks
Sentence 2 :- Geeks for geeks loves students.
Although both these sentences 1 and 2 use the same set of root words {student, love, geeks for geeks} they convey entirely different meanings.
→ ∴ In compositional semantics analysis, we try to understand how combinations of individual words form the meaning of the text.

* Tasks involved in Semantic Analysis :-
1. Word Sense Disambiguation
2. Relationship Extraction

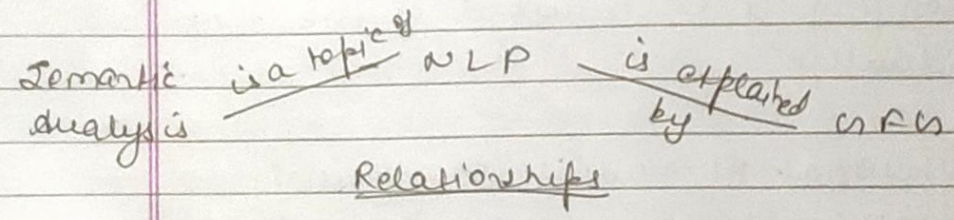
1. Word Sense Disambiguation :-
→ Involves interpreting the meaning of a word based upon the context of its occurrence in a text.
→ e.g., 'Bark' mean sound made by dog or outermost layer of a tree.

2. Relationship Extraction :-

→ Involves identifying various entities present in the sentence and then extracting the relationship bet^n those entities.

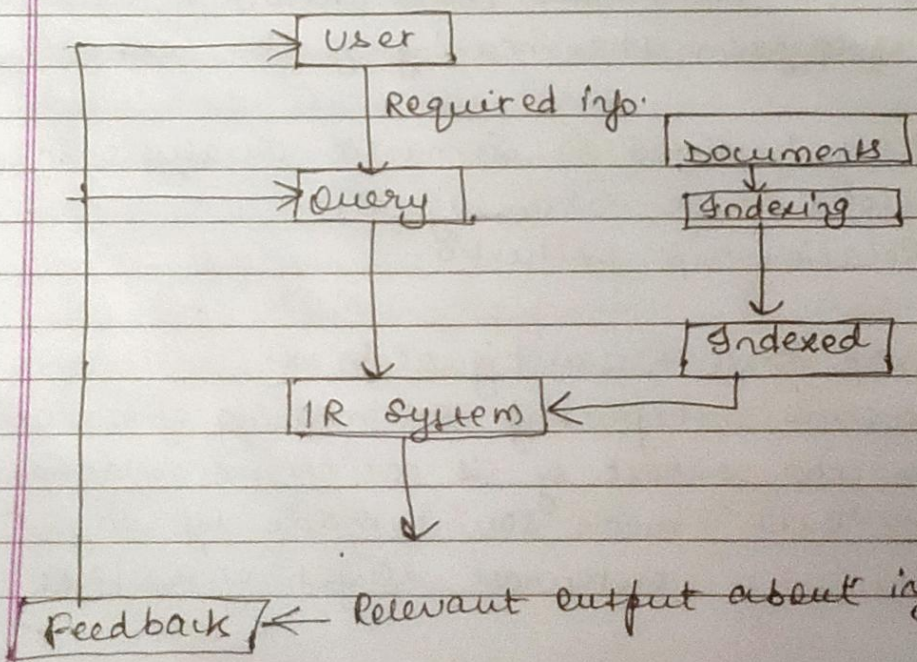
e.g,

Semantic analysis [entity] is a topic of NLP [entity] which is explained by NLP [entity]



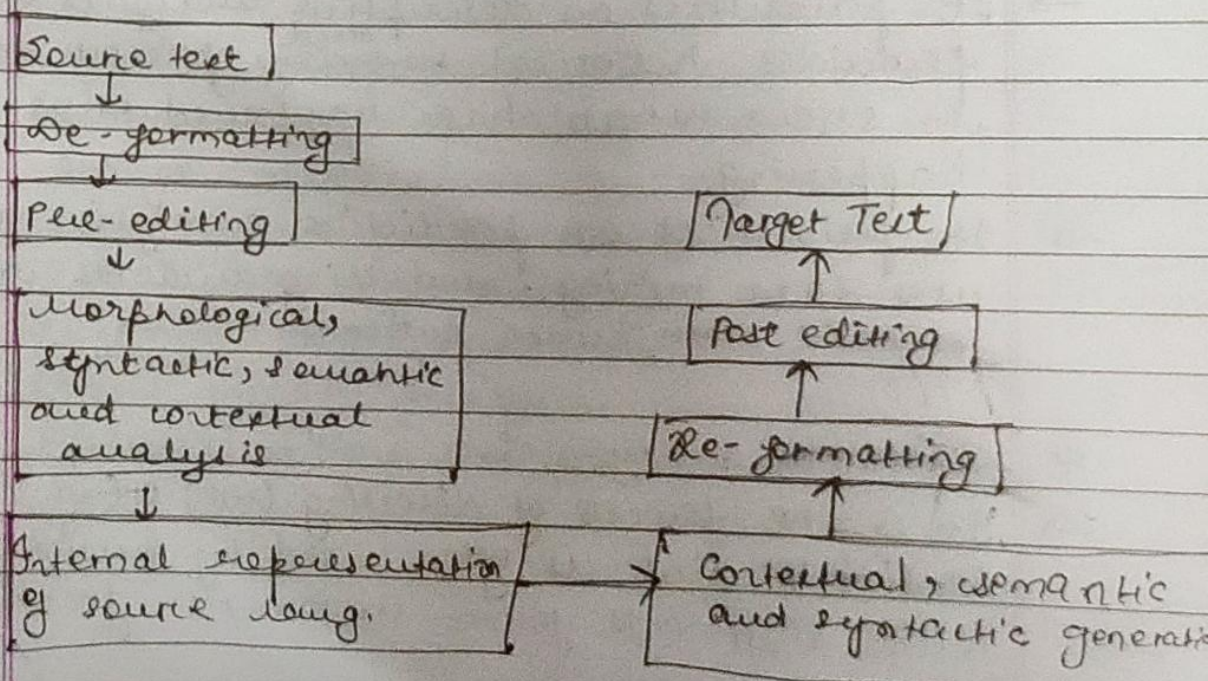
* Information Retrieval :-

→ IR defined as s/w program that deals with the organization, storage, retrieval and evaluation of info. from document particularly textual info.



→ It is clear from the diag. that a user who needs info. will have to formulate a request in form of query in NL.
The IR system will respond by retrieving the relevant output, in form of documents, about required info.

* ~~Models in IR~~ :-
* Machine Translation :-
→ MT is the task of automatically converting one natural lang. into another preserving the meaning of i/p text and producing fluent text in the o/p lang.
→ It refers to computerized systems responsible for producing translations with or without human assistance.
→ It excludes computer based translation tools that support translators by providing access to online dictionaries, remote terminology databanks etc.



* NLP appl^{n} :-
1. lang. Translator
2. Social Media Monitoring
3. Chatbots
4. Survey Analysis
5. Targeted Advertising
6. Hiring and Recruitment
7. Voice Assistants
8. Grammar checkers

* Business Intelligence and analytics :-
→ Business Intelligence is the process of collecting, storing and analyzing data from business operations.
→ BA refers to the practice of using company's data to anticipate trends and outcomes.
→ It includes data mining, statistical analysis and predictive modelling that helps make more informed decisions.
→ It prioritizes on descriptive analytics which provides historical and present data to show what has happened or is happening.
→ It prioritizes on predictive analysis, which uses data mining, modelling and ML to determine future outcomes.

* Sentiment Analysis :-
→ It is the process of detecting (+ve) or (-ve) sentiments and is often used by business to detect sentiments from customer's feedback.

→ eg: Sentiment analysis automatically analyzes 1,000+ reviews about your product will help you discover whether customers are happy or not.

→ Sentiment analysis models focus on polarity (+ve, -ve, neutral) but also on feelings (angry, happy, sad), urgency (urgent, not urgent) and even emotions (interested, not interested)

* DEEP LEARNING :-

→ Sub-field of ML concerned with algorithms inspired by the structure and function of the brain called artificial neural n/ws.

→ It has n/ws capable of learning unsupervised data from data that is unstructured or unlabeled.

→ It is a AI function that mimics the working of human brain in processing data for used in detecting objects, recognizing speech etc.

* Appl^{n} of Deep learning :-

1. Automatic speech recognition
2. Image recognition
3. Natural lang. Processing
4. Visual Art Processing
5. Recommendation systems.
6. Medial image analysis
7. Financial Fraud detection
8.

* Types of algo. used in Deep learning :-

1. Convolutional Neural N/W (CNNs)
2. Long Short Term Memory N/W (LSTMs)
3. Recurrent Neural N/W (RNNs)
4. Generative Adversarial N/W (GANs)
5. Radial Basis Function N/W (RBFNs)

* Cloud computing :-

→ Cloud computing is a virtualization-based technology that provides computing as service without the need to purchase the resources.

→ It is a technology that uses remote servers on the internet to store, manage and access data online rather than local drives. The data can be files, images, audio, video etc.

* Features :-

(1) Developing new appl$ and services
(2) Storage, backup and recovery of data
(3) Delivery of S/W on demand
(4) Analysis of data
(5) Streaming audio and videos.
(6) High Scalability
(7) Device & location independent
(8) low cost

* Intelligent agents :- is a program that can make decisions or perform a service based on its environment, user input and experience.

ANKIT
Date
Page

* There are 3- models of cloud :-
1. Infrastructure as service (IaaS)
2. Platform " " (PaaS)
3. Software " " (SaaS)

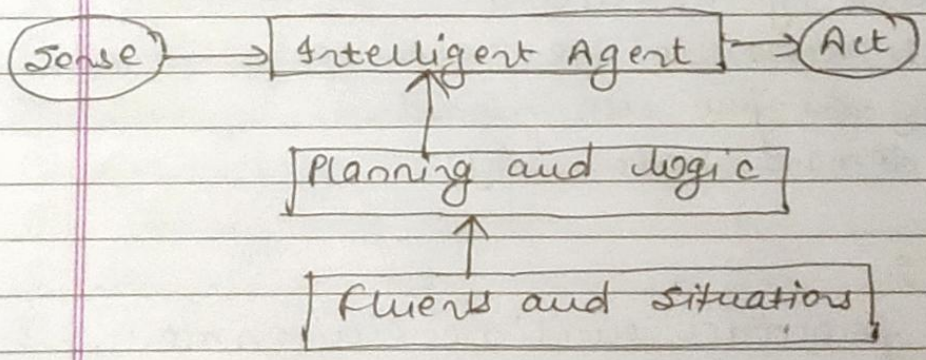
1. IaaS :-
→ IaaS allows customers to outsource their IT infrastructure such as servers, networking, storage, virtual and other resources.
→ Customers access these resources using a pay-as-p use model.
→ Provides following services :-
(i) Compute
(ii) Storage
(iii) Network
(iv) Load balancers.
→ Ex.
(1) Shared infrastructure
(2) Web access to the resources
(3) Pay-as-per use model.
(4) On-demand scalability.

2. PaaS :-
→ PaaS provides runtime environment.
→ It allows programmers to easily create, test, run and deploy web appl?.
→ You can purchase these appl? from cloud service provider on pay-per-use basis and access them using internet connection.

→ e.g. Google app engine, Azure.
→ PaaS providers provide programming lang, appl² framework, databases & other tools.

③ SaaS :-
→ Also called on-demand S/W.
→ These services are available to end-users over the internet so, the end-user do not need to install any S/W on their device to access these services.
→ Services provided by SaaS providers :-
① Business services
② Document management
③ Social n/ws
④ Mail services.

* Planning and logic in Intelligent agents :-



→ the agents involved is not just the perception and action but planning as well
→ To, have this there is a need of logic-based programming to support the decisions of the agents.

→ A typical "agent environment" is the one where agent senses the environment and acts on it.

→ The building blocks of these agents comprises fluents and situations this requires history of earlier action involved.

→ Situation : store group of actions.
Fluents include the objects with actions and situation