IRS stands for Information Retrieval System.

Here

Information : Data, facts

Information is Communicated (or) received knowledge conveying a particular facts.

Retrieval :- To get and bring back

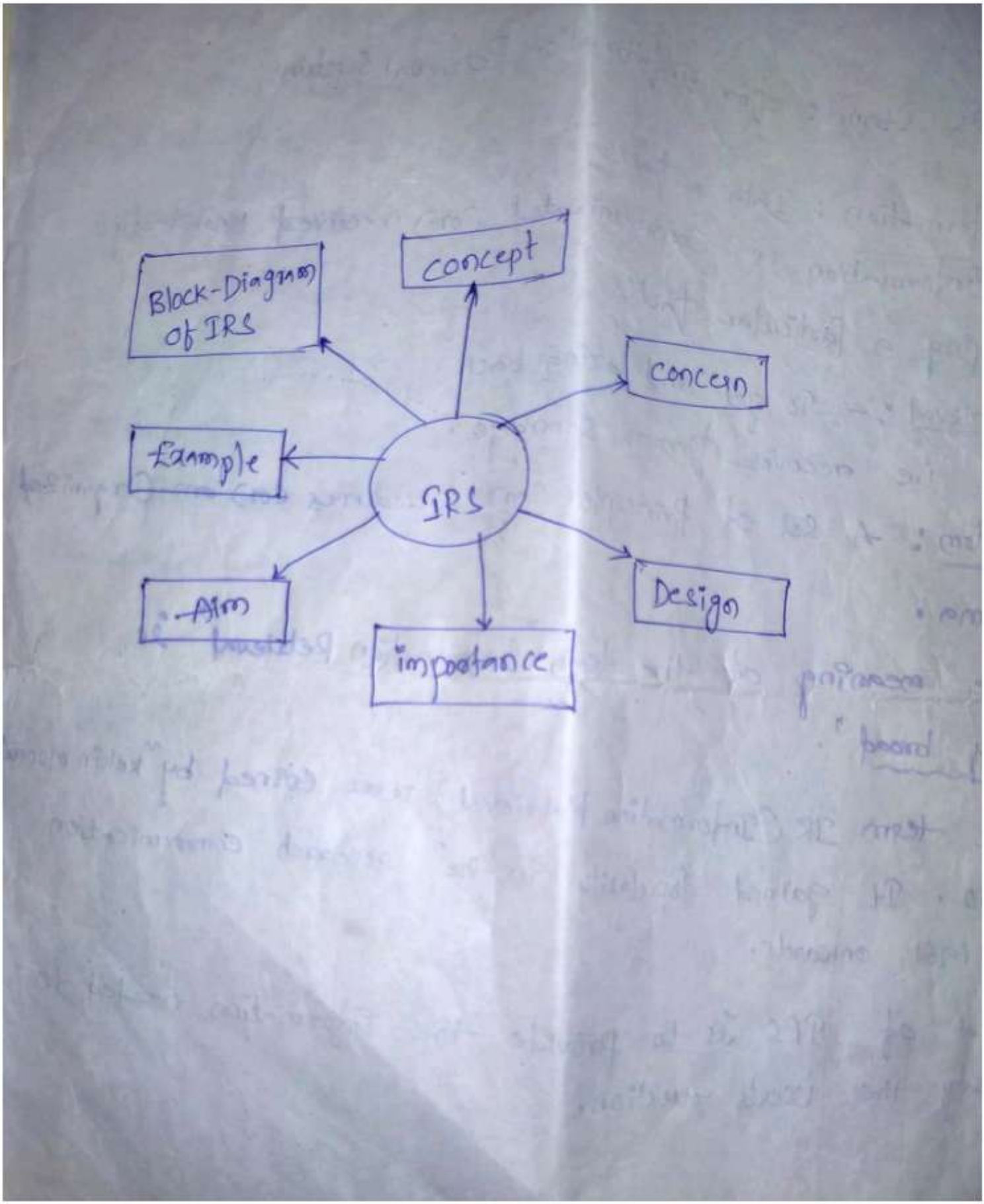
i.e recover from storage.

System : A set of Principles (or) procedures (or) an Organized schema.

⇒ The meaning of the term "Information Retrieval" is "very broad".

⇒ The term IR (Information Retrieval) was coined by "Kelvin Mooer" in 1950. It gained Popularity in the research communication from 1961 onwards.

⇒ Goal of IRS is to provide the information needed to satisfy the user's question.


Block-Diagram of IRS
concept
concean
Example
IRS
Aim
Design
importance

Introduction to Information Retrieval Systems:

⟹ The concept of Information Retrieval System (IRS) is "self-explanatory" from the "terminological Point of View" and refers to a system which retrieves information.

Here, "self-explanatory" means easily understood from the information already given and not needing further explanation.

"Terminological" means Technical words and expressions used in a particular subject.

⟹ IRS is concerned with two basic aspects:

i) How to store information &

ii) How to retrieve information.

Design :-

⇒ An IRS is designed to "analyze process" and store sources of information and retrieve when ever required. (or)

⇒ An IRS is a set of "rules and procedures" for performing some (or) all of the following Operations:

i.e * Indexing (or) constructing the representation of documents.

* Search formulation / constructing the representation of information needs).

* Searching / Matching representations of documents against the representations of needs

* Index language / Generation of rules of representation.

Importance of IRS :-

→ Information Retrieval systems are Very important. Important to make sense of the data.

→ Imagine how hard it would be to find some information on the Internet without Google (or) other search engines.

Aim of IRS :-

→ It provide the "best possible" information from a database.

Example of IRS :-

→ The most Obvious example of an IRS is "Google" and the english language has been extended with the term "Google it" means search for something.

Block Diagram of IRS:-

It consists of 3 components. They are

* Input
* Processor
* Output

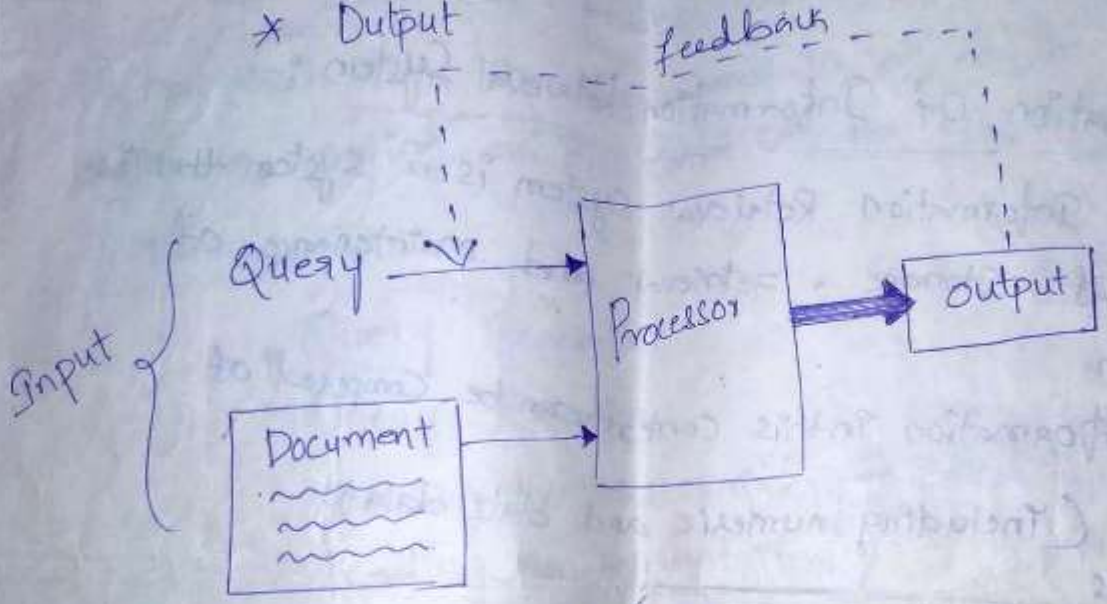


fig: IR Block Diagram

* **Input**:- It stores the only representation of the "Query" (or) "document".

Query: It represent the expression of the User -Information need.

Document: It represent the list of extracted words consider to be significant.

* **Processor**:- It involve in performing "actual retrieval function" and also executing the search strategy in response to a query.

* output :-
A set of extracted document is called output.

feedback :
It improving the subsequent run after sample retrieval.

(*) Defination of Information Retrieval System :-

⇒ An Information Retrieval System is a system that is capable of storage, retrieval and maintenance of information.

⇒ Information in this context can be composed of

- text (including numeric and date data).
- images
- audio
- Video and
- other multi-media objects.

→ The following terms discussed in IRS.

User :-
The term 'User' represents an end user of the information system who has minimum knowledge of computers and technical fields in general.

Item :-
The term "Item" is used to represent the "smallest complete unit" that is processed and manipulated by the system.

⊛ Objectives Of Information Retrieval Systems:

The general Objective of an IRS is minimizing the overhead of a user by locating needed information.

* Overhead;-

The Over-head can be expressed in terms of "time" i.e how much time the user spends in all of the steps leading to reading an item containing the needed information.

Example:-

* Query generation
* Query Execution
* Scanning the result of Query to select item to read,
* Reading non-relevant items.

The success of the Information System is very "Subjective", and it depends on two things

i) What information is needed.

ii) The willingness of a user to accept Overhead.

* Relevant Item:-

In information retrieval System the term "relevant" item is used to represent an item Containg the needed information.

* User's perspective :-

From a User's perspective "relevant" and "needed" are synonymous

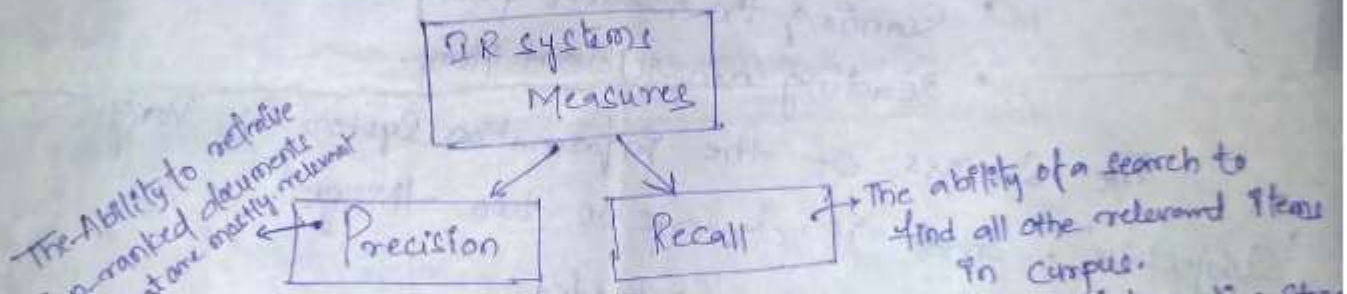
* System Perspective :-

From a System Perspective, Information could be relevant to search statement even though it is not needed / relevant to User.

Eg: User already knew the information.

→ The Objectives of IRS is commonly associated with two major measures:

i) Precision
ii) Recall.



∴ "corpus" means collection of information stored in the system.

Precision:

⟹ When a User decides to search for a topic, the total database is logically divided into 4 segments, as shown below.

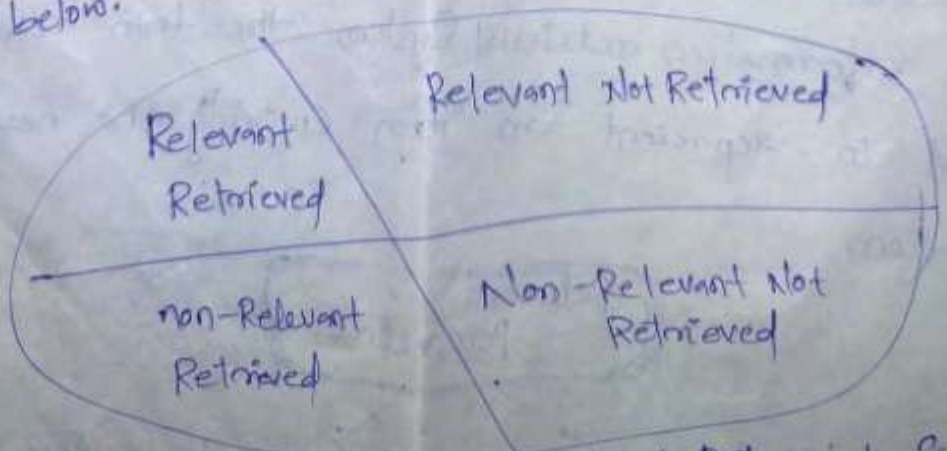


fig: Effects of search on total Document Space.

Let us see, what is "Relevant item" and "non-relevant item".

Relevant items :-

It is used to represent an item containing the needed information. that helps the Searcher/User to answering the question.

Non-relevant item :-

Non-relevant items are those items that do not provide any useful information directly.

⇒ The Precision and Recall formally defined as :

* $$\text{Precision} = \frac{\text{Number\_Retrieved\_Relevant}}{\text{Number\_Total\_Retrieved}}$$

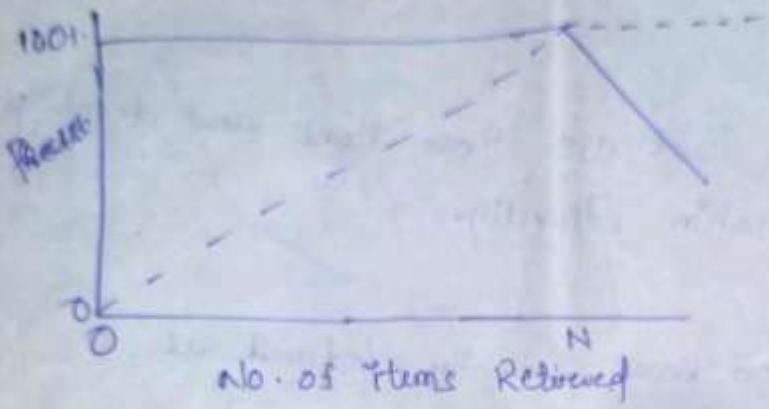
* $$\text{Recall} = \frac{\text{Number\_Retrieved\_Relevant}}{\text{Number\_Possible\_Relevant}}$$

Where,

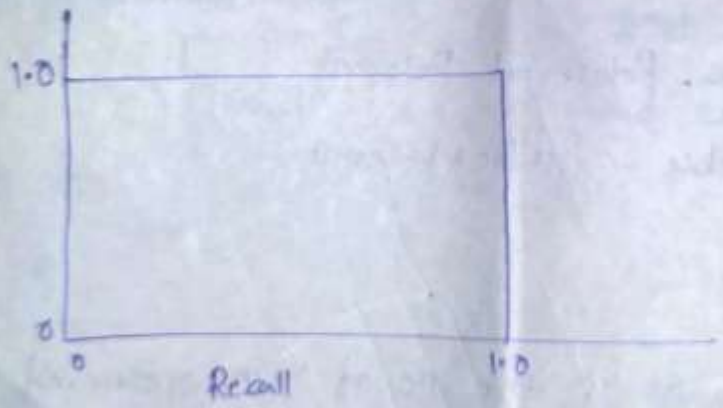
* Number_Retrieved_Relevant is the no. of items retrieved that are relevant to the 'User's needed search.
* Number_Total_Retrieved is the total no. of items retrieved from the "Query".
* "Number_Possible_Relevant" is the no. of relevant items in the "db" "database".

⇒ In a search of "100%", (or) in 100% search "85%" is Precision (i.e Precision = 85%) and the meaning of Recall is "15%" of the User effort is Overhead on deviating the non relevant items.

Let us see the 'Precision' and 'Recall' graphs:



(a) Ideal Precision and Recall



(b) Ideal precision/Recall Graph.

⇒ from 'a' diagrams, the basic properties of precision (solid line) and Recall (dashed line) can be observed.

-Here, 'Precision starts off' at 100% and maintains that value as long as relevant Items are retrieved.

'Recall starts off' close to zero and increase as long as relevant Items are retrieved. (until all posible relevant Items have been retrieved)

⇒ Once all "N" relevant items have been retrieved, the only items being retrieved are "non-relevant".

⇒ Precision is directly affected by non-retrieval of non-relevant items and drops to the number close to zero.

⇒ Recall is not effected by retrieval of non-relevant items and remains at 100%.

fig: c. Achievable precision/Recall graph.

⇒ fig (b) & fig (c) shows the optimal and currently achievable relationships b/w Precision and Recall.

⇒ To understand the implications on fig (c),

⇒ Assume that there are 100 relevant items in the database and from the graph at Precision of .3 (i.e 30%) there is an associated recall of .5 (i.e 50%).

This means there would be 50 relevant items in the hit file from the recall value.

⇒ A Precision of 30% means user would likely review 167 items to find the 50 relevant items.

(x) Functional Overview

A total Information Store and Retrieval System is composed of 4 major functional processes:

1. Item Normalization
2. Selective Dissemination of Information (i.e "mail").
3. Archival Document Database Search
4. Index database Search along with the Automatic file Build Process that Support index files.

Item Normalization:-

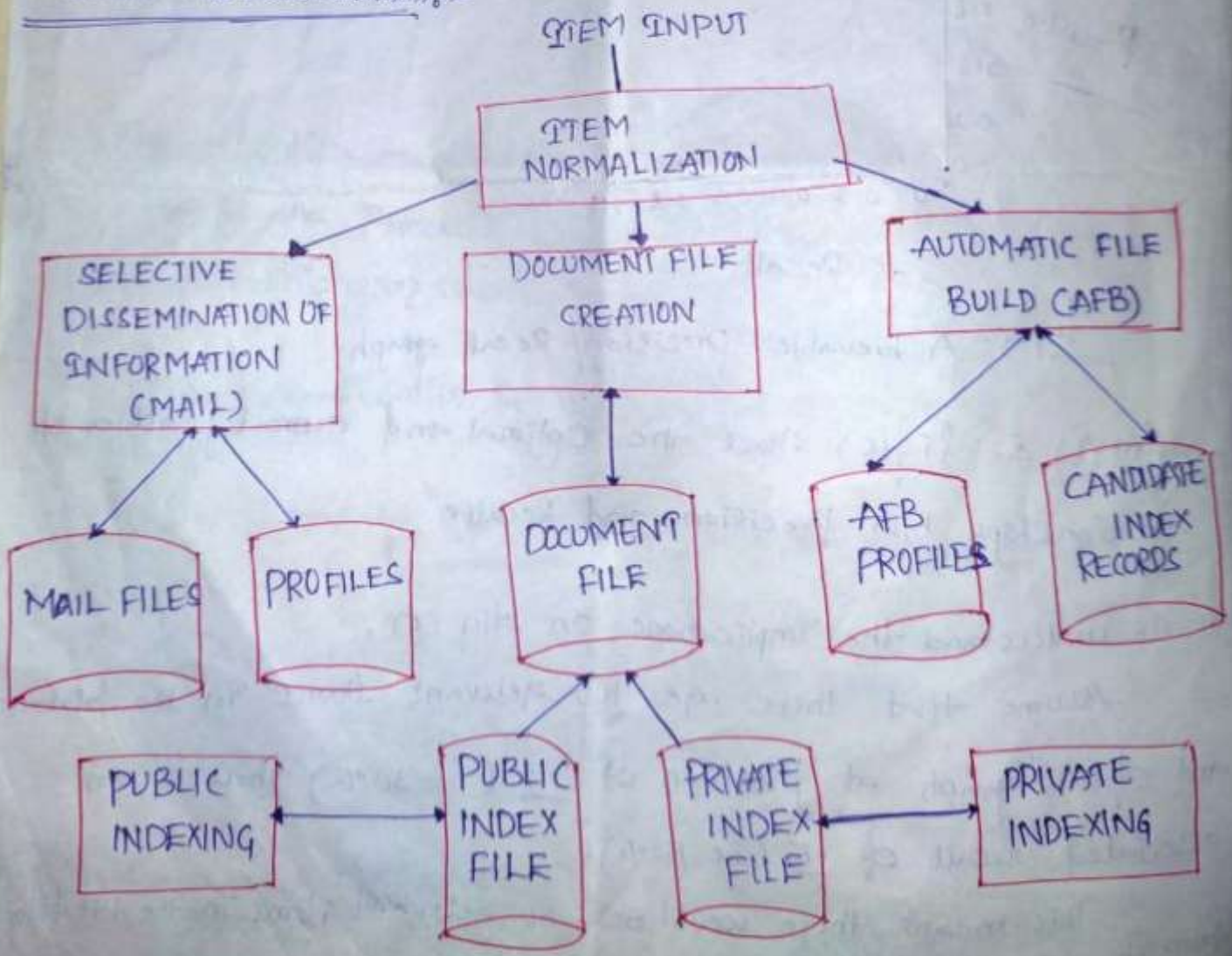


fig: Total Information Retrieval System.

① ITEM NORMALIZATION :-

The first step in any integrated system is to normalize the incoming items to a standard format.

Item / Item input :-

The term "item" is used to represent the smallest complete unit that is processed and manipulated by the system.

The definition of item varies how a specific source creates information.

For Example :-

→ A complete document, such as a book, a news paper or magazine could be an item. Sometimes the "chapter" and "articles" inside the magazine also consider as an item.

Normalization :-

It is the process of bringing (or) retrieving something to a normal condition (or) state.

⇒ Item Normalization provides "logical level restructuring of the item".

The following "Operations" are performed during the item - Normalization:

* "Identification" of Processing tokens (eg: Words).
* "characterization" of the tokens.
* "Stemming" of the tokens. (eg: removing word endings).

Now, let us see the text Normalization Process:

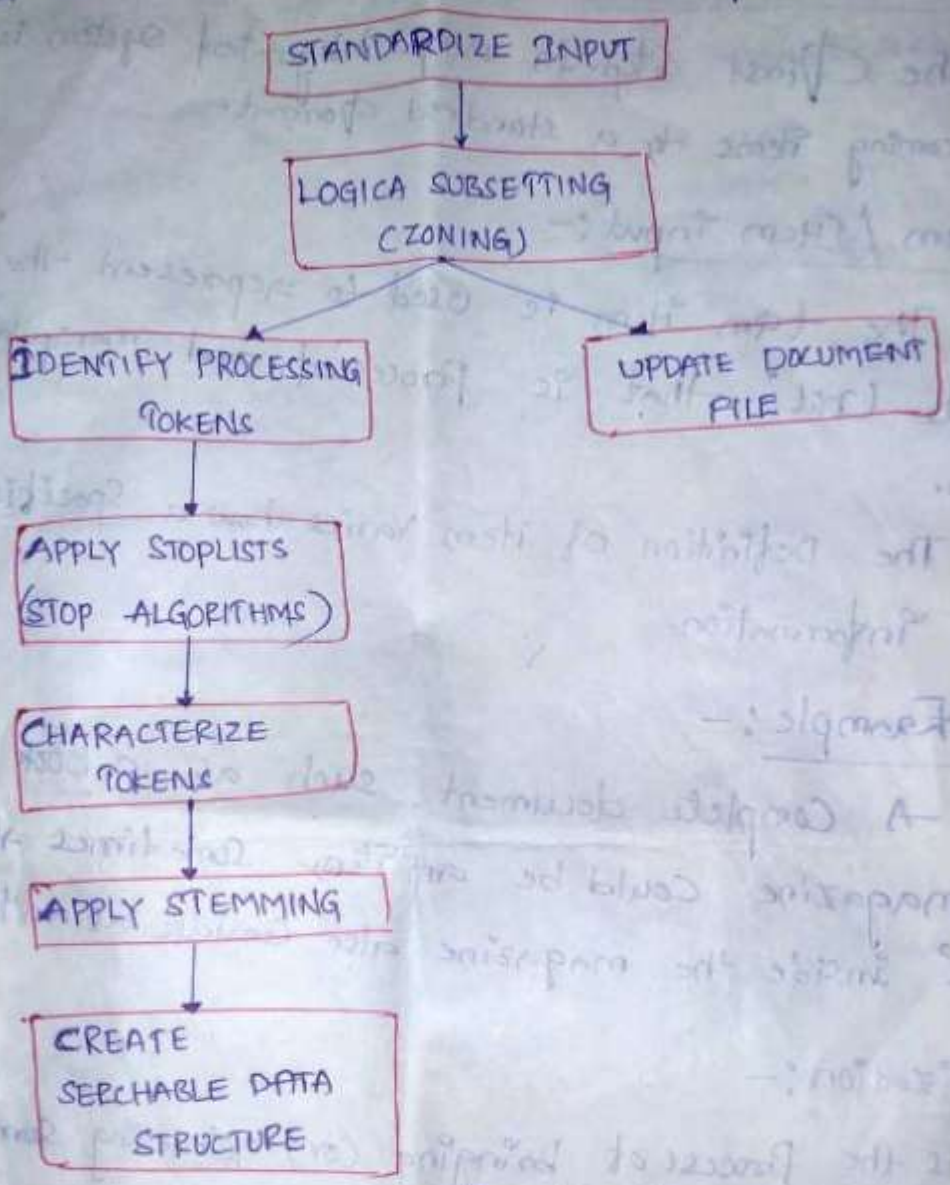


fig: The Text Normalization Process.

Standardize Input:-

⟹ Standardizing the input takes the different formats of input data and performs the translation to and the result of translation is acceptable by the system.

⟹ A system may have a single format for all the items (or) allow multiple formats.

Identify

UTF-8 → Unicode Transformation format-8 (15)

For Example :-

Standardization could be translation of foreign languages into "Unicode". Here Unicode is an international standard based upon 16-bit (i.e 2 bytes) that will able to represent all languages.

Ex: UTF-8

→ Every language has a different internal binary ending for the characters in the language. encoding

→ "One Standard encoding" that covers English, French, Spanish etc is ISO-Latin.

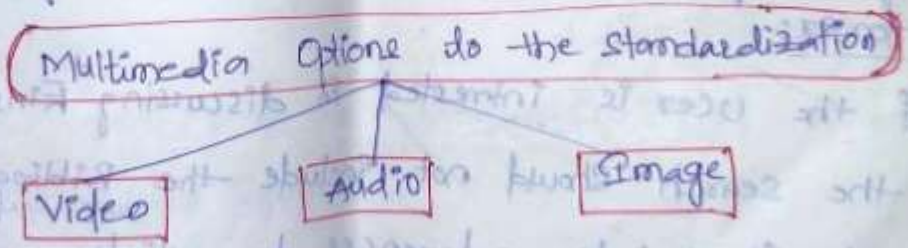
→ The another encoding for other language groups such as Russian, Japanese, Arabic etc is KOI-7, KOI-8.

the KOI-7 → kod Obmena Informatsiey, 7 bit which mean code for information Exchange 7 bit.

⇒ Multimedia is an extra dimension to the Normalization Process.

In order to normalize the textual input, the multimedia input also need to be standardized.

There are lot of options to standardized Normalization. i.e Input may be Video (or) Audio (or) Image.



Video :-

→ If the input is "Video" then following "digital standards" are applied.

→ The digital standards are either MPEG-2, MPEG-1, AVI (or) Real Media.

MPEG → motion Picture Expert Group

MPEG standards are the most universal standards for higher quality video where Real Media is the most common standard for lower quality video being used on Internet.

Audio :-

Audio standards are typically WAV/(or) WAVE Real media (Real Audio).

* WAV → Waveform Audio file format.

* Real audio is a property Proprietary Audio formed developed by Real Networks in 1995.

Image :- Images vary from JPEG to BMP - Bitmap image file

JPEG → Joint picture expert Group

Logical Subsetting (Zoning) :-

→ The second step in the text normalization process is to parse the item into logical sub-divisions called zoning.

→ A typical item is sub-divided into zones and the zones which may overlap and can be hierarchical, such as Title, Author, Abstract, main Text, Conclusion and References.

→ In zoning search is restricted to "specific zones" sometimes.

Ex:-

If the user is interested in discussing "Einstein" Articles then the search should not include the Bibliography, instead of that it include references to articles written by "Einstein".

→ Once search is completed user is efficiently reviewing the needed information.

→ Zoning is performing the two tasks.

i) Identify processing tokens.

ii) Update Document file.

Identify processing tokens:-

Once standardization and zoning has been completed, let us see two terms "Identify" & "Processing token"

→ Information (i.e words) that are used in the search process need to be "identified" in the item.

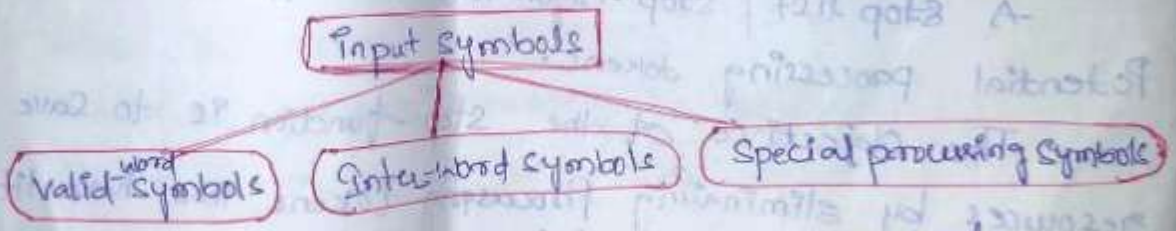
→ The term "processing token" is used because a "word" is not the most efficient unit on the search structure.

⇒ The first step in the identification of a processing token is determining the "word".

⇒ System determine the words by dividing the input symbols into three classes:

i. Valid Word Symbols
ii. Inter-Word Symbols
iii. Special processing symbols.

i.e



i. Valid Word Symbols:-

A Word is defined as a (contiguous) set of Word symbols bounded by Inter-Word symbols.

(sharing a common border)

ii. Inter-Word Symbols:-

In many systems, Inter-word symbols are non-searchable and should be carefully selected.

Examples: Semicolons, blanks, an apostrophe '

iii. Special processing Symbols :-

There are some symbols, that may required in special - Processing.

Example : Hyphen (-)

ie A Hyphen can be used in many ways,

* If it is used at left to the Writer means it gives the Judgment of the Writer Ex: Beinstein - E4.

* If it is used at the end of a line, means indicates the continuation of a word.

* If it used in other place means it links the independent words.

Apply stop lists (Stop Algorithms) :-

Next step,

A stop list / stop Algorithm is applied to the list of Potential processing tokens.

The objective of the stop function is to save system resources by eliminating processing tokens that have little value to the system.

The best technique to eliminate the majority of these words via stop algorithm is first list out them individually.

Examples of stop algorithms are:

→ Stop all the numbers greater than "999999" (this was selected to allow dates to be searchable).

* Stop any processing token that has numbers and characters is determined.

Characterize Tokens :-

Next Step in finalizing the processing tokens is "Identification of any specific word characteristics".

The characteristic is used in the systems to assist "disambiguation of a particular word".

For example :-

for a word "plane",

- the system understand that it could mean "level constant" as an "adjective".
- * "aircraft (or) facet" as a "noun".
- * "The act of ~~sem~~ smoothing or evening" as "a verb".

Apply Stemming :-

Once the potential processing token has been identified and characterized, most of the systems apply stemming algorithms to normalize the token.

The decision to perform the stemming is tradeoff (or like b/w) between "precision" of search and "recall".

for Example :-

The System must keep singular, plural, past tense, Possessive etc.

Create Searchable data structure :-

Once the processing tokens have been finalized, based upon the stemming algorithm, they are used as Updates to the searchable data structure.

→ The searchable data structure is the internal representation (i.e not visible to the user) of items that the user query searches.

② SELECTIVE DISSEMINATION OF INFORMATION:

SDI is a concept that was introduced in information science by Hans peter Luhn in 1958.

SDI is also called as "mail".

The selective dissemination of information (mail) process provide the capability to dynamically compare newly received items in the information system against standing statements of user.

The Mail process is composed of 3 things:

i) Search process
ii) User statements of interest (profiles) &
iii) User Mail files.

i) Search Process:-

After receiving item, it is matched against every user profile.

ii) Profiles :-

A Profile contains search statements along with list of user mail files.

→ Profiles define all the areas in which a user is intrested.

iii) User Mail files :-

→ When a search statement is satisfied, the item is placed in the "mail files" associated with profiles.

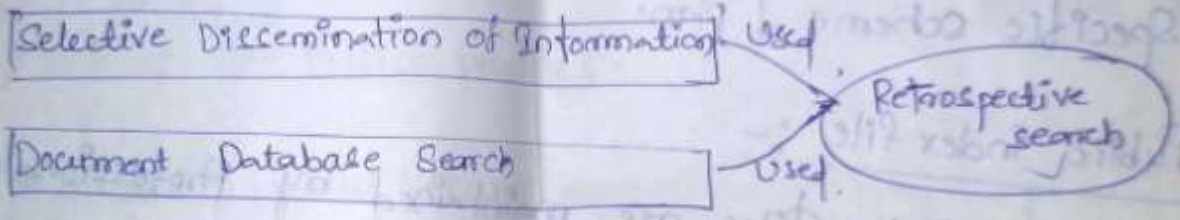
→ Items in mail files are typically viewed in the time of receipt (receiving) order and they are automatically deleted after a specified time period (eg- after a month etc)

③ DOCUMENT DATA BASE SEARCH :-

→ The document database search process is composed of
- search process
- User entered Queries
- the document database.

→ The document database, which contains all the items been received, processed & stored by the system.

→ The Selective Dissemination of Information and document database search both are used retrospective search.
i.e



→ The search can be performed in 2 ways
i.e 1. Perspective Search
2. Retrospective Search.

Perspective Search :

Searching in future and it is time valid.

RetroSpective Search :-

Searching in Past and it is time invalid.

⇒ Document database is very large, it consist of 100's of millions of items.

... classified in 2 types

Once the Document Database Search is completed then it Creates the document file.

The Document file is classified into 2 types.

i) Public Index file (public indexing)

ii) Private Index file. (Private indexing).

Indexing :-

It is Originally called as "cataloging".

→ Indexing refers to Organization of data according to specific scheme / plan.

Public Index file :-

→ Public Index files are maintained by Professional library services.

→ these files have access list that allow any one to search (or) retrieve data.

→ Public Index files are used to perform the public indexing.

Private Index file :-

→ Private Index files typically have very limited access.

→ Every user can have one (or) more no. of private Index files that leading to a very large no. of files.

→ Private Indexifiles are used to perform the private Indexing.

④ Automatic File Build (AFB):-

The automatic file build is also called as "Information Extraction".

AFB is classified in 2 types / performed through

i) AFB profiles

ii) Candidate Index Record.

AFB profiles:-

The index term extraction process are stored in Automatic file Build profiles.

Candidate Index Record:-

- When an item is processed it results in creation of Candidate Index Records.

⑧ Relationship to DataBase Mangement Systems :-

There are 2 Major categories of systems available to process items :

1. IRS
2. DBMS

i.e

Relationship to DBMS

⇒ A "confusion" can be arise, when the "software systems" supporting each of these Applications get confused with the data they are Manipulating.

① * An Information Retrieval System is "software" that has the features and functions required to manipulate "information" items

VS (Versus)

* A DBMS that is Optimized to handle "structured" data.

② * Information is "fuzzy text".

The term fuzzy is used to imply the results from the minimal standards. (or) Controls on the creators of the text items.

i.e the minimal standards in terms of Author is.

"The author is trying to present concepts, ideas and abstractions along with supporting facts."

VS (Versus)

* The Structured data is well defined data (facts) typically represented by "tables".

i.e table is associated with "attributes" and attribute contains semantic description.

For example:

→ There is no confusion to what values enter in a specific database record.

⇒ We have different attributes like "employee Name" and "employee Salary".

(3) * On the Other hand, if two different people generate "an abstract" for the Same item, they can be different.

i.e One abstract may generally discuss the most important topic in an item.

Another abstract may Specifys the details of many topics.

(VS) Versus

* With Structed data a User enters a specific request and the result retured with the desired information.

The results are frequently tabulated and presented in a report format for ease of use.

(4) * Search of "information" items in IRS & DBMS.

* A search of information items has a highpropably of not finding all the items a User is looking for.

So that in search User enter the additional items of interest, this process is called "iterative search". doing Something again&ag

→ ~~The~~ An information retrival System gives the capabilities to the User in finding the relevant items Such as "relevance feedback".

→ The Result from the/an Information System search are presented in "Relevance ranked Order".

* The Confusion comes, when DBMS software is used to store "Information".

This is easy to implement, but the system lacks the "ranking and relevance" feed back features that are critical to an Information System.

→ It is also Possible to have structured data used in an Information System, (such as TOPIC).

finally,

from a Practical Standpoint, The Integration of DBMS's and Information Retrival System is Very Important.

(*) Digital Libraries and Data Warehouses:-

Two Other Systems frequently described in the context of Information retrieval are "Digital Libraries and Data Warehouses" (Data Marts).

There is Significant Overlap between these two Systems

Digital Libraries:-

They are also called as "Online Library" (or) "Internet Library" (or) Digital Repositories.

A digital library is a library in which collections are stored in 'digital format'.

An electronic (or) Digital Library is a type of Information retrieval System (IRS).

It provides the Architecture to

* Model
* Map
* Integrate &
* transform scattered information in digital documents.

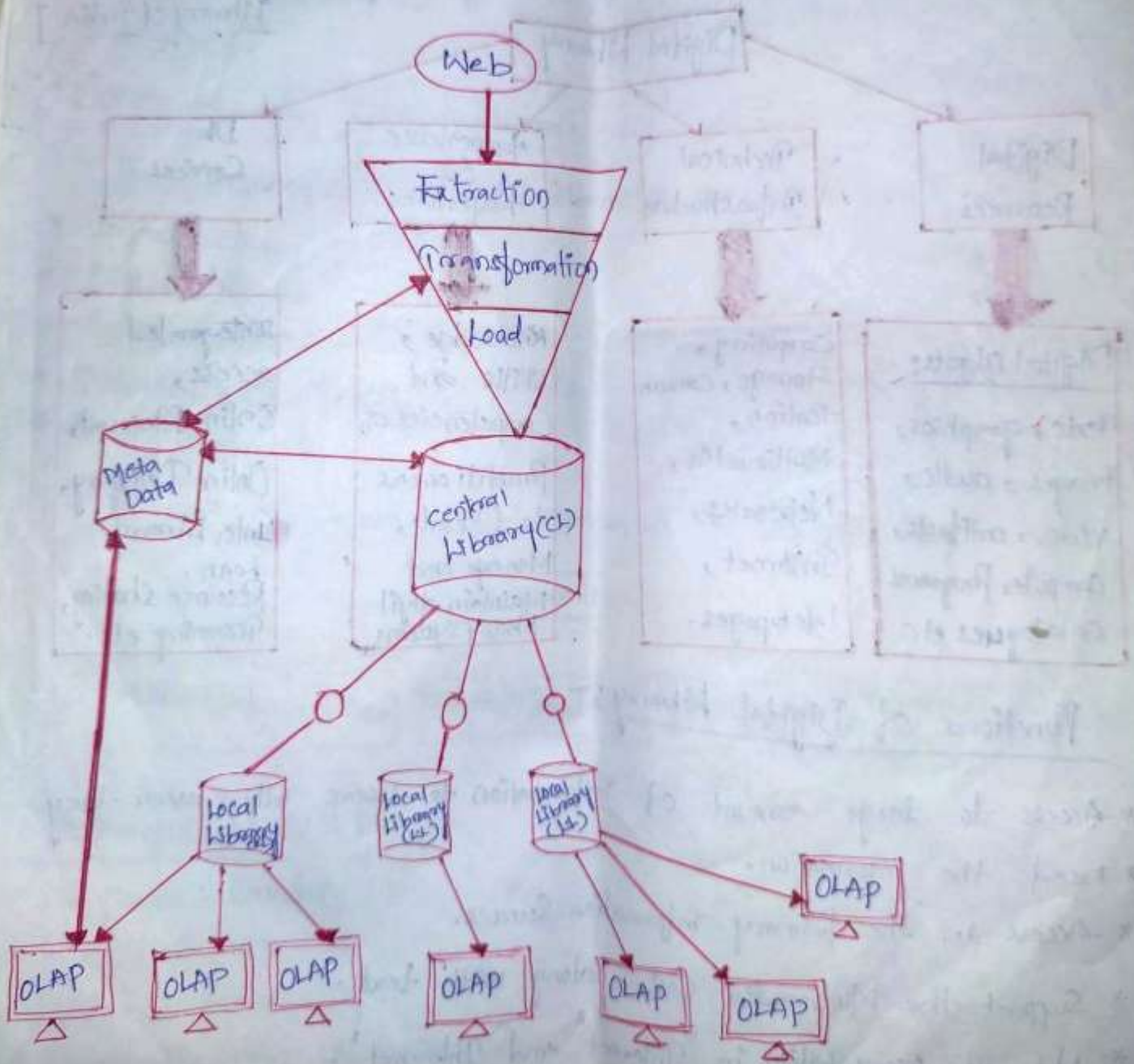


fig: Digital Library

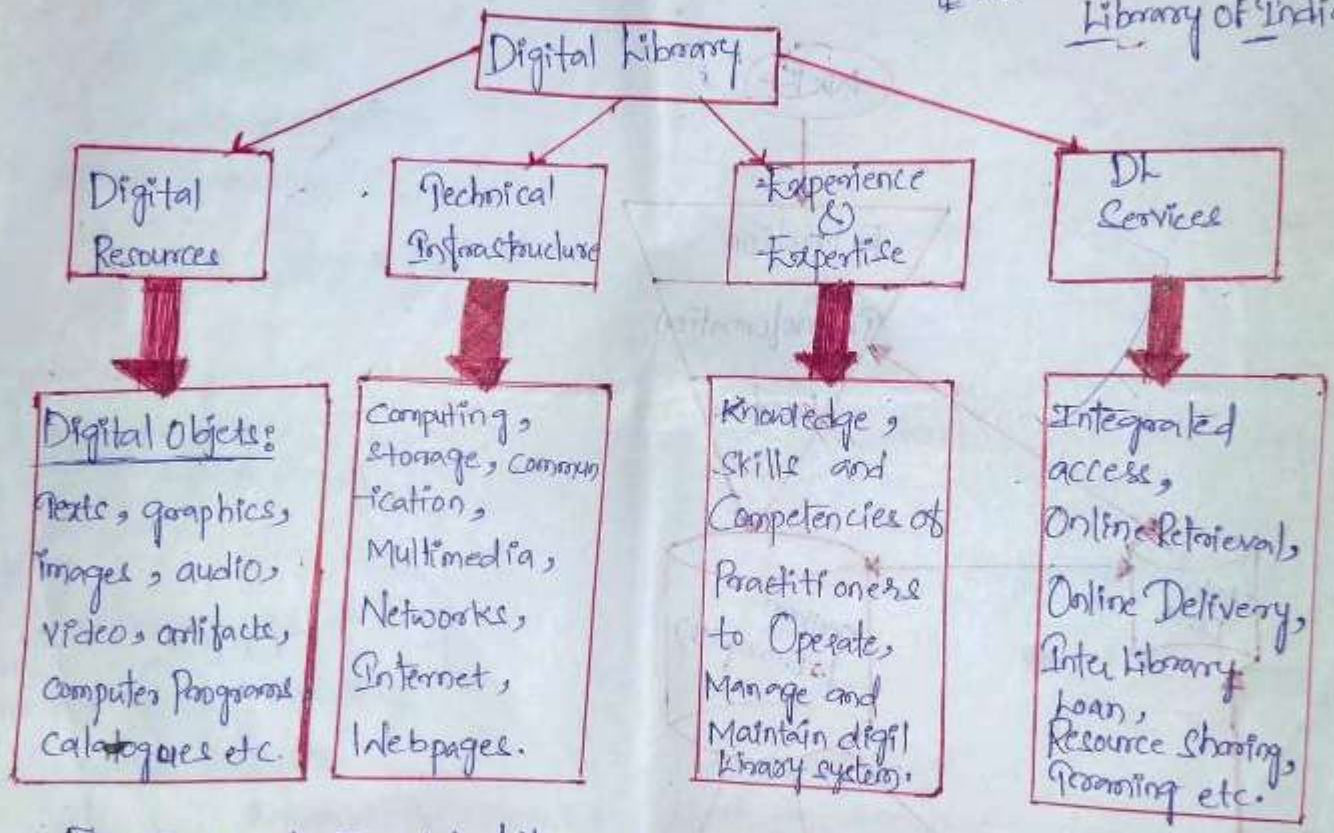
[: OLAP → Online Analytical Processing)

→ → The term DL's (Digital Library's) was first popularized by the NASA Digital Libraries initiated in 1994.

→ Digital Library is not a single Entity. It require technology that link the resources of many collections.

The links between the digital libraries and their resources are transparent to Users shown in following figure.

[NDLI → National Digital Library of India]



Functions of Digital Library :-

* Access to large amount of information to users when ever they need the information.
* Access to the Primary information Sources.
* Support the Multimedia Content along with text.
* Network Accessability in "Internet" and "Intranet".
* Client - Server Architecture
* Advanced Search & Retrieval. * Integration With Other Digital Libraries.

Purpose of Digital Library :-

→ To expedite the systematic development of procedures to collect, store and organize information in digital form.

→ It Promote efficient Delivery of information economically to all Users.

→ It Strengthen Communication and collaboration "between" and "among" educational Institutions.

→ It takes "leadership Role" in the generation and dissemination of knowledge.

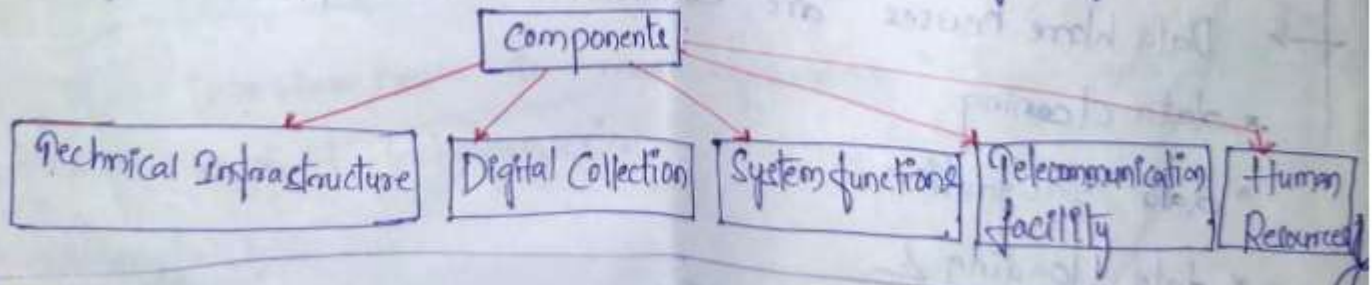
Components of Digital Libraries :-

The Components of Digital libraries are:

* Infrastructure
* Digital Collection
* System functions
* Telecommunication facility
* Human Resources.

External

You can Represent these Components in following way.



Advantages of DL's :-

→ No physical Boundary
→ Multiple Accesses
→ Universal Accessibility
→ Round the clock Availibility
Added Values
Enhanced IR.

Limitations:

* Lack of screening (or) Validation.
* Lack of Preservation of "Best in class".
* Lack of Preservation of "fixed Copy".

i.e for the record and duplicating scientific research.

Data Ware house (or) Data Marts :-

→ A data Ware house is a repository of information collected from multiple sources, stored under a "Unified Schema" and that are usually resides at a "Single site", (or)

~~The Process of collecting information aquisition~~

~~is called~~.

→ A data Ware house refers to a place where data can be stored for useful mining.

where "mining / data mining" refers to process of extracting useful data from the databases.

→ Data Ware houses are Constructed via aprocess of

* data cleaning
* data Integration
* data loading &
* periodic data refreshing.

→ The "term" "Data Warehouse" came from the Commercial Sector than Academic Sources.

Goal :-

⇒ The 'goal' of the data ware house is to provide the Critical information to decision makers so that they can answer the future queries.

⇒ The data ware house consist of the "data" cons" an information directory that describes the contents and meaning of data being stored.

Working:

An "Input function" that captures data and moves it into the data ware house.

↓

The "Search data & manipulation tools" that allow the user to access & analyze the warehouse data.

↓

The "Delivery Mechanism" export the data to other warehouses, datamarts and external systems.

⇒ focus of Data Ware house:

The Data ware house is more focused on "Structured data" & "decision support technologies".

for Example :-

The typical framework for Construction and Use of data ware house for all electronics.

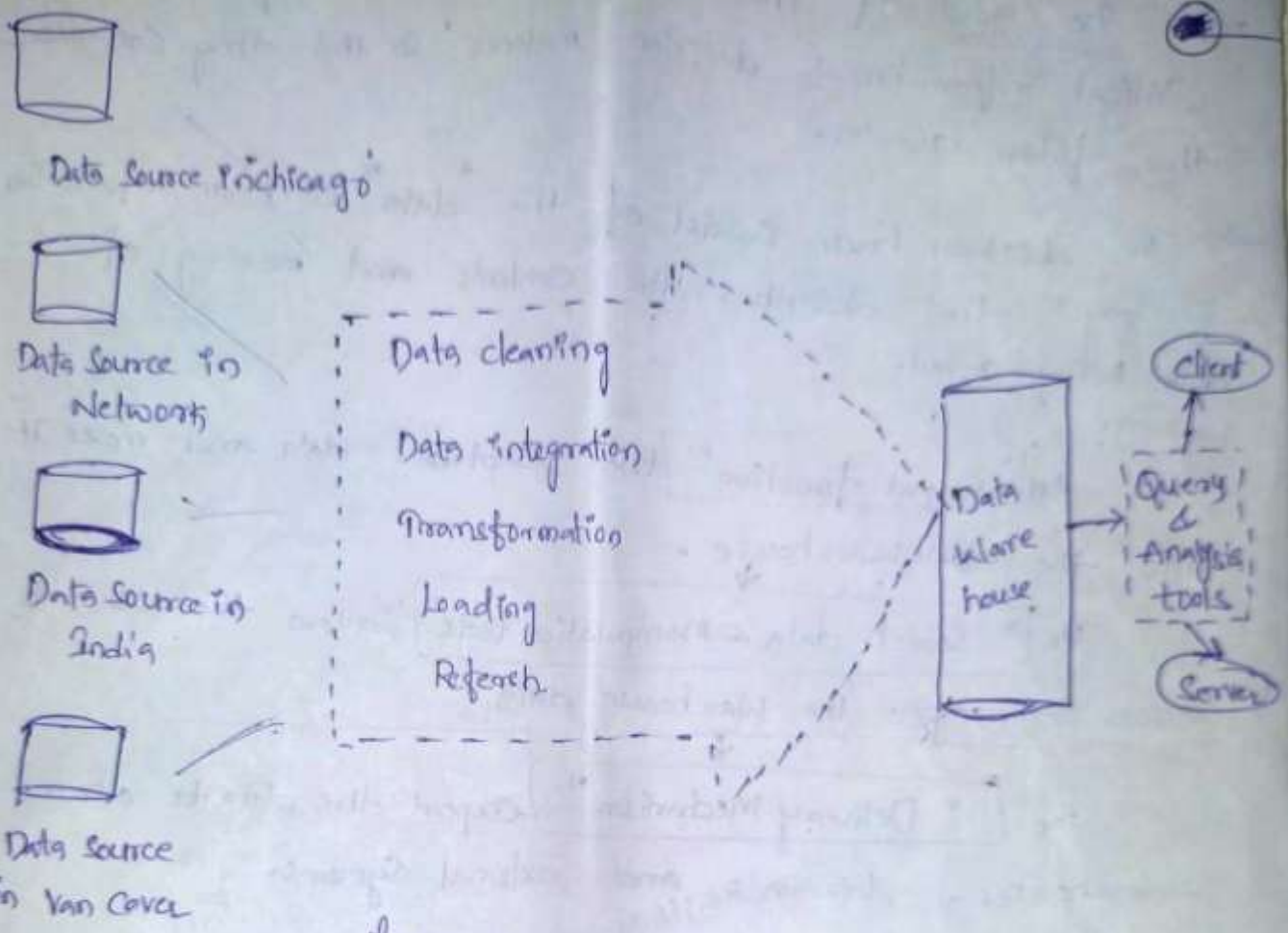


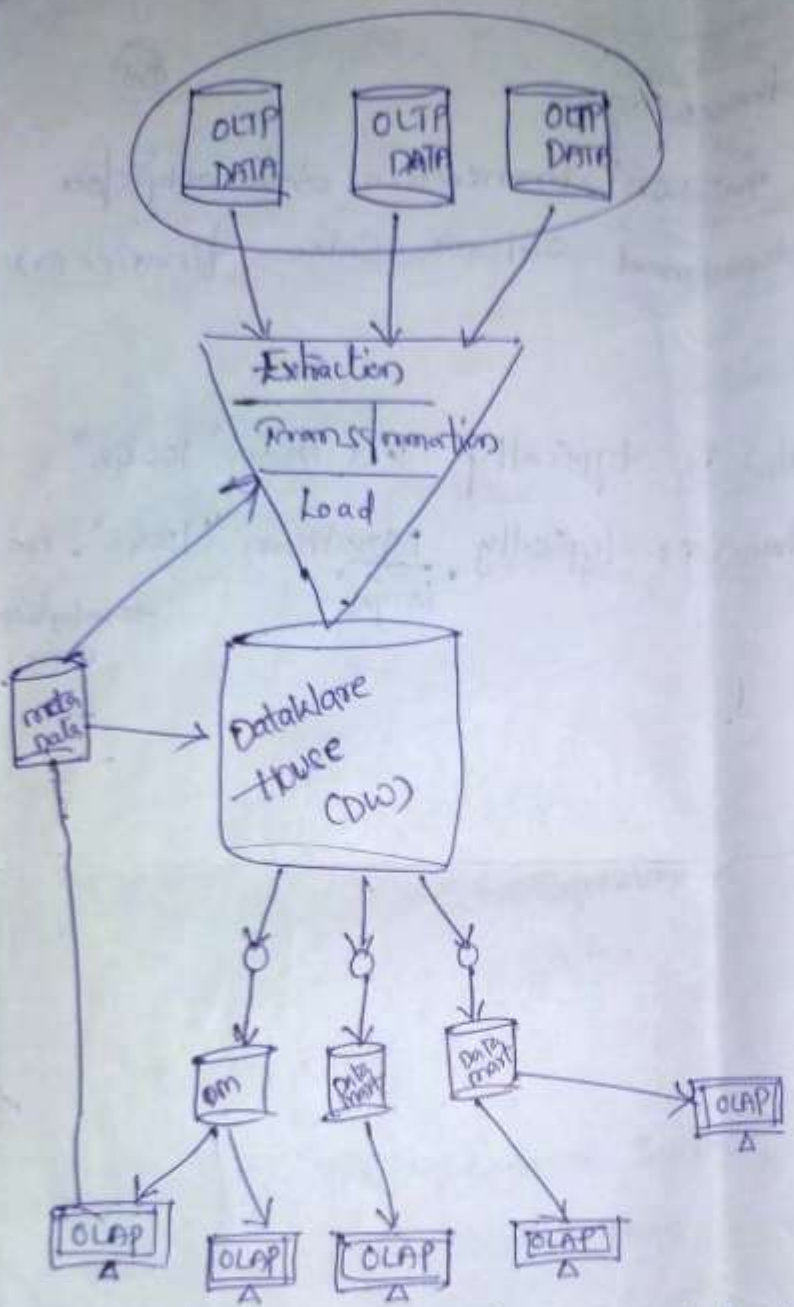
fig :- Data-Ware house.

Data mining :-

→ The term Data mining is also called "KDD" Knowledge Discovery in Data Base

→ KDD is the process of ~~converting~~ discovering valuable information from a collection of data. (or)

→ The process of converting the raw data into useful information.



DW → Data Warehouse
Dm → Data mart

fig (b) Datawarehouse.

OLAP :-

OLAP online analytical processing

→ This Approach is used to answer Multidimensional Data model.

(34)

* OLTP :-

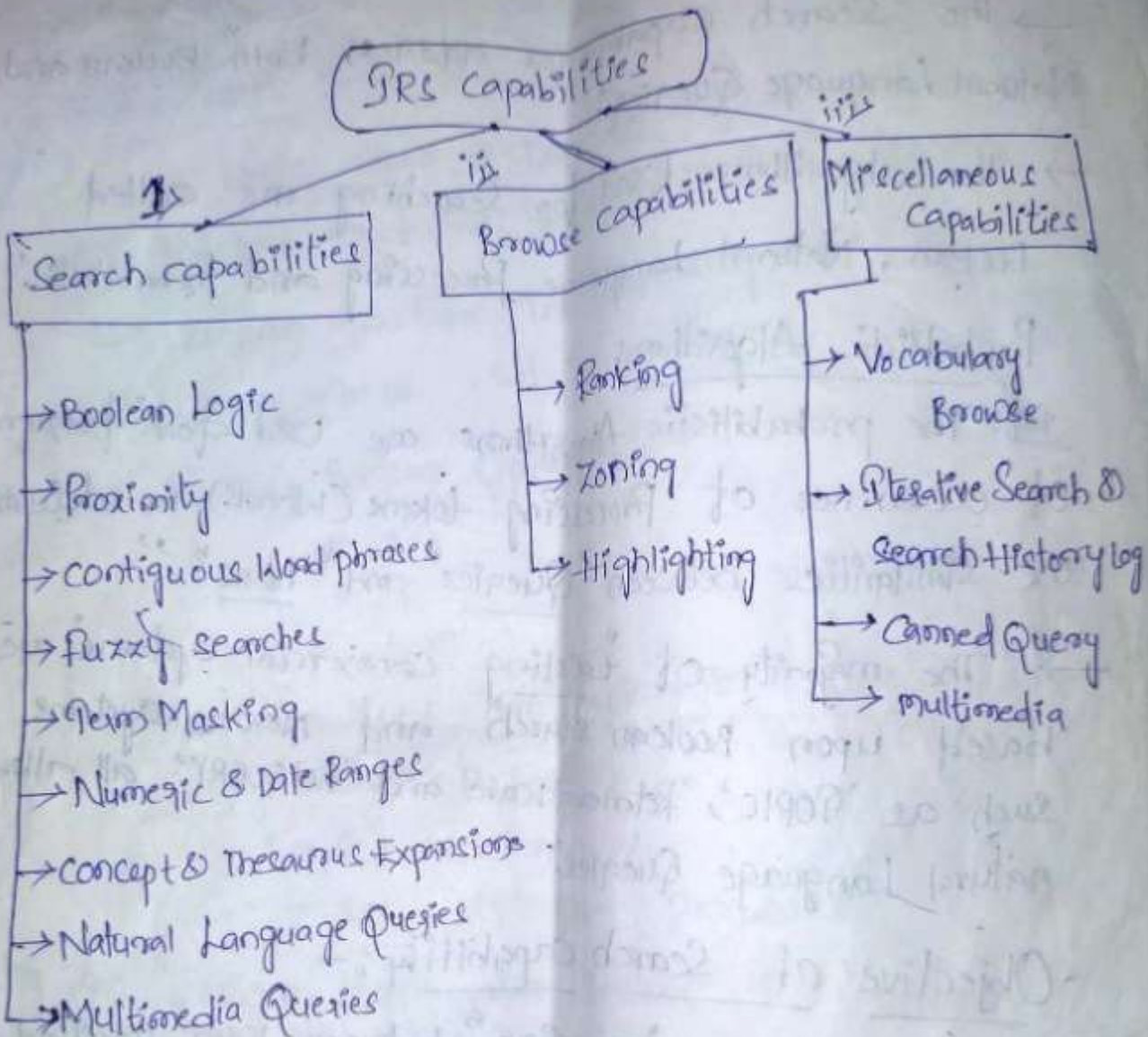
OnLine Transaction Processing.

→ It "captures and maintain" transaction data only for a Specific ~~document~~ department such as Sales, finance (or) HR (Human Resources).

→ The DM (Datamart) is typically less than "100GB"

→ The DW (DataWarehouse) is typically More than "100GB". i.e larger terabytes or more.

IRS Capabilities



In Information Retrieval System, search and browse capabilities are play crucial role to assist the user in locating the relevant items.

1) Search Capabilities :-

→ The Search Capabilities address both "Boolean and Natural Language Queries".

→ The algorithms Used for Searching are called "Boolean, Natural Language Processing and Probabilistic".

Probabilistic Algorithm :

Def: The probabilistic Algorithms are Used for "frequency of occurrence of Processing tokens (Words) in determining the similarities between "Queries" and "items"."

→ The majority of "Existing Commercial Systems" are based upon Boolean search and Newer Systems such as 'TOPIC', 'Retrieval ware' and 'INQUERY' all allow the natural Language Queries.

Objective of Search Capability :-

It allow the 'mapping' between User specified need and the items in the information database as a result ~~and~~ information database will answer the need.

→ One of the concept that has been occasionally in Commercial Systems are "Weighting".

Weighting :-

The Process of holding the location and ranking of relevant items.

IRS provides the different search capabilities They are:

1) Boolean Logic :-

→ Boolean logic allows a user to logically relate multiple concepts together to define what information is needed.

→ The Boolean functions in processing tokens are identified any where with in an item.

→ The Typical Boolean Operators are:

* AND
* OR
* NOT

These Operations are implemented using

→ Set Intersection
→ Set Union &
→ Set Difference procedures.

→ The Boolean logic also include the "Order of Boolean Operations" and "default precidence order (or) no precidence order".

Order of Boolean Operations :-

Placing the portion of search statement in parentheses are used to specify the Order of Boolean Operations.

default precidence order Operations :-

If the parentheses are not used the system follows a default precidence Ording of Operations.

EX:- typically NOT then AND then OR.

→ A special type of Boolean Search is called "M of N" logic.
i.e A set of possible search "terms", that are identified and accepted by any item that contains subset of the terms.

(38)

for example :-

Find any item containing any two of the following terms : ["AA", "BB", "CC"]

This can be expanded into a "Boolean search" that performs an AND operation between all the combinations of two terms and OR operation to results together.

i.e 

i.e (AA AND BB) Term1
(AA AND CC) Term2
(BB AND CC) Term3
} AND Operation b/w all combinations.

(AA AND BB) OR (AA AND CC) OR (BB AND CC)
Term1 Term2 Term3
} OR operation for result.

Example:- Use of Boolean Operations:-

Search statement	System Operation
* COMPUTER OR PROCESSOR NOT MAINFRAME	→ select all items discussing "computers" and/or "processors" that do not discussing mainframes.
* COMPUTER OR (PROCESSOR NOT MAINFRAME)	→ select all items discussing computers and/or the items that discuss processors and donot discuss Mainframes.
* COMPUTER AND NOT PROCESSOR OR MAINFRAME	→ select all items that discuss computers and not processors or mainframes in the item.

Fig: Use of Boolean Operations.

ii) Proximity:-

Proximity is used to restrict the distance allowed within an items between two search terms.

The typical format for proximity is:

Term1 within "m" "units" of Term2

(or)

TERM1 with in m units of TERM2

Here "m" is indicating the "distance Operator".

→ The distance Operator m is an integer number and units are in characters, Words, sentences and/or paragraphs.

Proximity Operators :

* Direction Operator → (i.e before / after)
* ADJ (Adjacent) Operator → (forward only direction i.e one direction)
* Distance is set to zero → (within same paragraph / unit)

* Direction Operator :-

Some times, the proximity relationship contains "Direction Operator" indicating the direction (before / after) that the second term must be found with in no:of (number of) units specified.

For Example :-

Search statement	System Operation
* "United" within five words of "American"	→ It would "Hit on", "United states and American interests"; "United Airlines and American Airlines". It would not "Hit on", "United states of America and the American dream".

* ADJ (Adjacent) Operator :-

A Special Case of ~~proxi~~ proximity ~~Operator~~ is the Adjacent ADJ operator, that normally has a distance operator of "one" direction i.e forward only direction.

For Example :-

SEARCH STATEMENT	SYSTEM OPERATION
* "Venetian" ADJ "Blind"	→ It would find the items that mention a "Venetian Blind" on a window but not items discussing a "Blind Venetian".

* Distance is set to zero :-

→ An another Special case is when "Distance is set to zero" meaning "with in the same semantic Unit".

for Example :-

SEARCH STATEMENT

"Nuclear" within zero paragraphs of "clean-up".

SYSTEM OPERATION

It would find the items that have "nuclear" and "clean-up" in the same paragraph.

iii) Contiguous Word phrases :-

→ In, A contiguous Word phrase is two or more Words that are treated as a "single semantic Unit".

for Example :-

"United states of America"

ie It is "four" Words that specify a search term representing single semantic Unit (i.e "country").

→ The single semantic Unit satisfies all the proximity operators i.e ADJ, Direction operator & Distance with is set to zero operator.

→ A CWP is specified in both ways

i.e * Query

* a special search Operator.

for Example :-

A Query could specify "manufacturing" AND "United states of Ameria".

i.e

Which returns any item that contains the word "manufacturing" and the contiguous words "United states of America"

→ The Proximity and Boolean Operators are "Binary Operators" but Contiguous Word phrases are "N" ary Operators Where "N" is the number of words in the CWP.

→ The Contiguous word phrases are called "Literal Strings" in "WAIS" and ~~Extract~~ Exact prases in "Retrieval Ware".

Where WAIS :- WAIS is a standard for

* indexing
* Storing &
* Retrieving the information from Source - document that can be located any where on the Internet.

→ In WAIS multiple Adjacency (Adj) Operators are used to define a Literal String.

Ex:- "United" ADJ "States" ADJ "Of" ADJ "America".

iv) Fuzzy search :-

Fuzzy search provide the capability to "locate the spellings of words" that are similar to the entered search term.

⟹ This function is primarily used to compensate for errors in spellings of words.

⟹ A fuzzy search on the term "computer" would automatically include the following words from the information database:

"computer"
"compiter"
"conputer"
"computter"
"compute".

⇒ Maximum utilization of fuzzy search in the system that accept is (OCR) optical Character Read.

OCR :- In OCR a hardcopy item is scanned in to a binary image. (or)

Process of scanning the hardcopy into binary image is called 'OCR'.

VI Term Masking :-

→ Term Masking is the ability to expand a query term by masking a portion of the term and that maps to the unmasked portion of the term.

→ The value of term masking is much higher in systems that do not perform "stemming".

→ There are two types of search term Masking:

i.e.

Some times the fixed-length and Variable-length

→ Some times both "fixed and Variable length" are called "don't care" functions.

* fixed length Masking :- (£)

→ fixed length Masking is a "single position Mask".

→ It Mask out any symbol in a "particular position".

→ fixed length term Masking is not frequently used.

→ It is typically not Critical to the Systems.

→ It is indicated by "£".

for example :-

SEARCH STATEMENT	SYSTEM OPERATION
* Multi£national	Matches "multi-national", "multiynational", "multinational" But it does not match the "multi national" since it is two Processing tokens.

* Variable length Masking :- (*)-

→ Variable length 'Dont-care functions' allows masking of any number of characters within a processing token.

→ The Variable length Masking may be in the front, at the end, at both ends.

i.e * Suffix search :- The front end means, if the masking is performed at the "starting position" of the processing token is called "Suffix Search.

i.e "*COMPUTER" → Suffix Search.

* Prefix Search :- If the masking is performed at the "ending position" of the processing token is called Prefix Search.

i.e "COMPUTER*" → Prefix Search.

* Imbedded String Search :- If the masking is Performed at "both ends" of the Processing token, is called "Imbedded String Search".

i.e "*COMPUTER*" → Imbedded String Search

→ The Variable length Masking is indicated by "*".

For Example :-

SEARCH STATEMENT	SYSTEM OPERATION
→ *computer*	✱ It Matches, "microcomputer", "minicomputer" (or) "computer"
→ comput*	✱ It Matches, → "computers", → "computing", → "computes".
⇒ *comput*	✱ It Matches, → "micro computers" → "minicomputing" → "Compute".

Fig. Term masking

VI) Numeric and Date Ranges :-

→ Term Masking is useful when applied to Words, but does not Work for finding ranges of numbers (or) Numberic dates.

→ To find the numbers larger than "125" using a term "125*" will not find any number.

→ As a part of system Normalization process, characterizes words as "numbers (or) dates".

For Example :-

* for Numbers and Dates :

If "125-425" means finding the numbers (range) in between two numbers.

iii) [1/2/1993 — 5/2/1995] means finding date range in between given dates. (HI)

* for Infinite Ranges :-

for representing the infinite ranges between the numeric data it uses special type of operators

i.e [> (or) >= (or) < (or) <=]

Example : "> 125" means Larger Numbers above the 125.

"<= 233" means Smaller (or) equal to 233.

vii) Concept & thesaurus Expansion :-

* Concept :-

→ A concept [class] is a "tree structure" that expands each meaning of a word into potential concept.

→ Concept classes are sometimes implemented as a "Network structure" that links the word stems.

→ "Concept class representation":

The Concept class representation, assist a user who has minimal knowledge of "concept domain". So that user can easily expand "related Concepts".

for Example :-

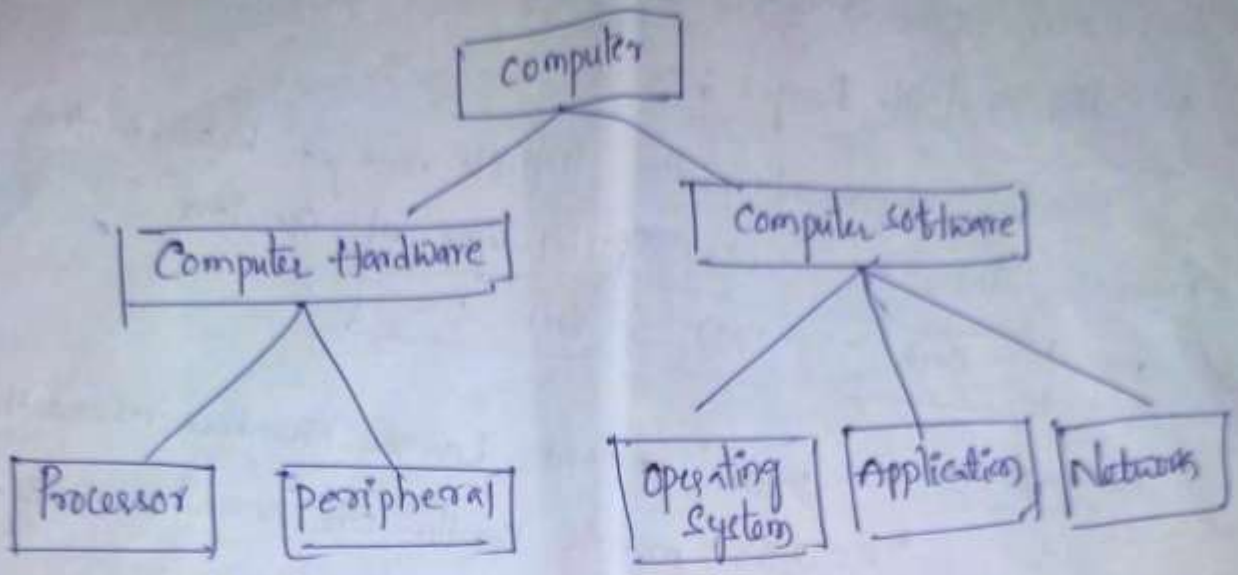


fig :- Hierarchical Concept class structure for "Computer"

Thesaurus Expansion :—

A Thesaurus is typically a one-level (or) two-level expansion of a term to other terms that are similar meaning.

For Example :

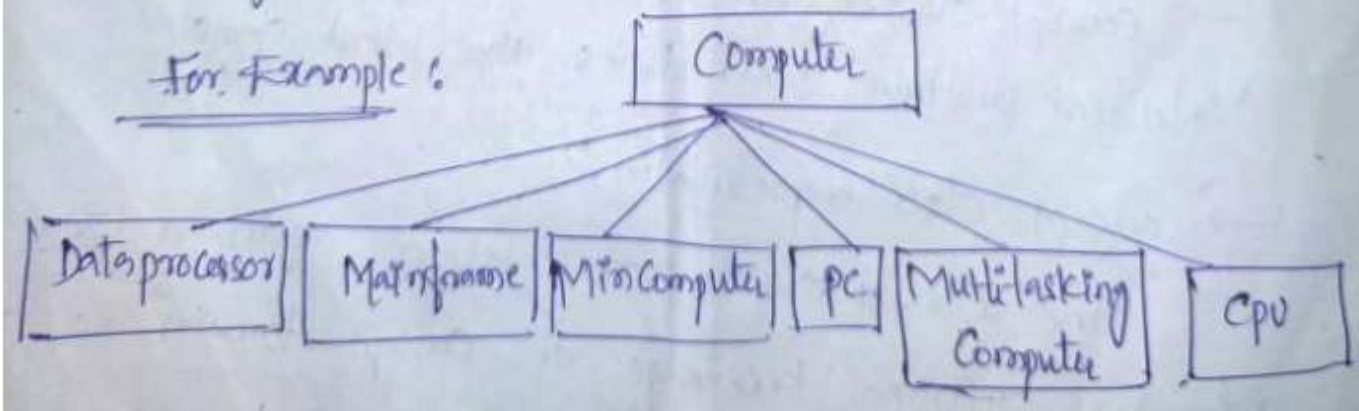


fig: Thesaurus for term "Computer".

(viii) Natural Language Queries:-

→ Natural language query consists only of "normal terms" in the user's language, without any special syntax (or) format.

Natural Language Queries allows a user to enter a prose statement that describes the information, ie that the user want to find.

The longer prose returned the more accurate results.

The Most difficult logic case associated with Natural Language Queries is the ability to specify negation in the search statement.

Example :- ie Example of Natural Language Query:

find for me all the items that discuss oil reserves and current attempts to find new oil reserves. Include any items that discuss the international financial aspects of the oil production process. Do not include items about the oil industry in the United States.

→ The minimized input for above example is

Oil reserves and attempts to find new oil reserves, international financial aspects of oil production, not united states oil industry.

(50)

The Boolean query for same search statement is:

("locate" AND "new" and "oil reserve") OR ("international" AND finance* AND "Oil production") NOT ("oil Industry" AND "United states").

ix) Multimedia Queries :-

The User Interface becomes far more complex, with the introduction of the availability of multimedia items.

→ The image (Could be) used to search the images that are part of an item.

→ They also could be used to locate specific scene in a video product.

- sub-division

(151)

2. Browse capabilities :-

Browse capabilities Provide the user, capability to determine which items are of interest and select those items to be displayed.

There are 2 ways of "displaying" summary of items that are associated with Query.

i.e i) Item line status

ii) Data Visualization.

Item line status :-

-from the summary,

The User can select the specific items and "Zones with in the items" for display.

Data Visualization :-

The systems allows easy transitioning between the "Summary display" and "review of specific items".

Browse capabilities are of 3 types.

i) Ranking

ii) Zoning

iii) Highlighting.

i) Ranking :-

⇒ The Introduction of Ranking based upon
- * Predicted relevance Values,
- * the status summary displays, &
- * the relevance score associated with item, along with - brief discription.

Relevance score :

→ The Relevance score estimate the item that satisfies the search statement.

→ The Relevance Scores are Normalized to a value between [0.0 and 1.0].

Here

"The Highest Value of 1.0" means the system is sure that the item is relevant to search statement.

The Lowest Value of 0.0 means "not relevant".

⇒ In addition to, ranking is based up-on the "characteristics of item" and the database.

⇒ Ranking uses "collaborative filtering" for selecting and Ordering outputs. → is a technique that can filtered the items that user might like.

In this case, when user reviewing the items it provide - feedback to the system on the relative value.

The "collaborative filtering" has been very successful in sites such as AMAZON.com, MovieFinder.com are deciding what products to display to users based on their queries.

ii) Zoning :-

→ The Process of dividing the standard input into logical sub-divisions is called zoning / zones.

Zone : The "title" is frequently called as zone.

It provide the additional information to the user with "relevance weight" to avoid selecting the non-relevant items for review.

→ The Zoning is used ~~for~~ in minimizing what an end user needs to review from a hit item in Passage based search and retrieval.

iii) Highlighting :-

→ Highlighting indicates how strongly the highlighted word participated in the selection of the item.

So that User quickly focus on potentially relevant parts of the text.

→ Highlighting has been useful in Boolean systems to indicate the cause of retrieval.

ie it provide the direct mapping between the "terms in the search" and "terms in the item".

→ The Highlighting may vary by introducing colors and intensities to indicate the importance of a particular word.

3. Miscellaneous Capabilities :—

The additional functions that facilitate the user's ability to input queries, and reduce the time taken to generate the queries".

The Miscellaneous Capabilities are 4 types:

i) Vocabulary Browse
ii) Iterative Search and Search-History log
iii) Canned Query
iv) Multimedia.

i) Vocabulary Browse :-

Vocabulary Browse Provides the capability to display in alphabetical sorted Order Words from the document database.

All "unique words / processing tokens" in the database are kept in sorted Order along with a number count of Unique items in which the word is found.

The User can enter "a word" (or) "Word fragment" and the system will begin to display the dictionary around the entered text.

i.e "The System indicates what word fragment the user entered. and then it alphabetically displays other words found in the database".

The User can continue scrolling in either direction of reviewing additional terms in the database.

Use: It helps the user to determine the impact of using a fixed (or) Variable length mask on a search term. and i.e Potential mis-spellings.

i.e compul* in effect is searching for "Compulsion" (or) "compulsive" (or) "compulsory".

for potentially misspelled (Some one entered the word) "Computes" (they really meant "computer").

For Example :- Term: Comput

TERM	OCCURRENCES
Comput	
i.e	
* Compulation	265
* Compute	1245
* Computed	1
* Computer	10,800
* Computerized	18
* Computes	29

fig: Vocabulary browse list with entered term "comput".

iii) Iterative Search and Search History log :-

"Iterative Searching" and "Search history logs" summarize Previous search activities so that user access the Previous results from the current user session.

Iterative Search :-

→ The Results of the previous search can be used to create a new query, rather than typing a complete new query, this has same effect as taking the original query. i.e "result is same as Original Query" by adding the additional "AND" condition to the search statement.

This Process is called "Iterative search".

Search History Log :-

→ The search history log is the capability to display "all the previous searches" that were executed during the current session.

→ It provide the facility to locating the previous searches as starting points for new searches.

iii) Canned Query :-

→ The capability to "Name" a Query, and "store" it to be retrieved, & "executed" during a later user session is called canned Query / canned Queries.

→ The canned Queries is also called as "stored Queries".

→ Queries that start with a canned Query are significantly larger than ad hoc Queries.

iv) Multimedia :-

Multimedia introduces new challenges, for potential users, that satisfy the discovered query, and multimedia techniques are used for display them.

The 'transcribed text' used as navigation technique through the audio.

i.e they Appropriately label this new paradigm

What You See Is (Almost) What You Hear

(WYSIAWYH).

→ The transcribed text could be used as an Index into future retrieval of the audio source.

i.e. The tonal information provided by the original speech.

Unit - II

Cataloging and Indexing

Cataloging :-

→ Cataloging is a process that creates "metadata" and represents "information sources" such as books, sound recordings, moving images etc.

→ The Cataloging provides the information such as

- the name of the creator,
- the title and the subject term that describes the

Source, through creation of a bibliography record.

Indexing :-

The indexing is one of the most important process in the IR system.

The transformation from the received item to Searchable data structure is called Indexing.

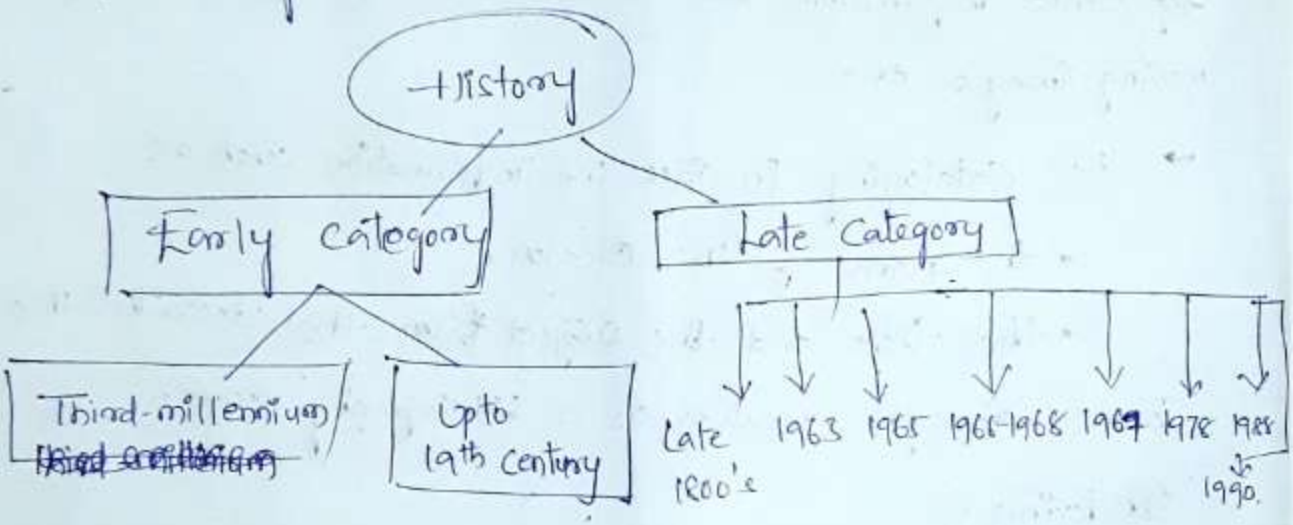
Indexing helps to retrieve the information "accurately" and "efficiently."

⇒ Indexing Originally called as "cataloging" and it is the oldest technique for identifying the contents of items.

⇒ Objective: The Objective of the cataloging is give the access points to a collection, that are expected and most useful to the users information.

* History and Objectives of Indexing:

The Basic information required for an item is "What is the item" and "What it is about", has not changed over the centuries.



Early Category :—

As Early as,

In third-Millennium, in "Babylon" The Libraries of "Cuneiform tablets" were arranged by the Subject (By Hyman - &q).

Up to 19th century :—

Upto 19th century there was little advancement in "Cataloging". ie The advancement is only changes in the methods used to represents the Basic Information.

Late category :-

* In the Late 1800 :-

The subject Indexing became "Hierarchical"

Eg: Dewey Decimal System (DDS)

(or)

Dewey Decimal Classification (DDC).

DDS/DDC :-

The Dewey Decimal system is the World's most widely used way to Organize Library collections.

i.e The Dewey Decimal System used by the Libraries to arrange "books" via "subject".

→ DDC allows newbooks to be added to a library in their appropriate location based on subject.

It was first published by "Melvil Dewey" in 1876, in United States.

* In 1963 :-

"The Library of Congress initiated".

i.e A study on the Computerization of bibliographic surrogates.

Library of congress (Lc) :-

The Lc is "research Library" that officially serves the "United states Congress".

The Library of Congress "Collections are Universal" not limited by subject (or) format (or) national - boundary.

It include "research materials" from all parts of "the World" and in more than 470 Languages.

The Library of congress
→ "Established" in April 24, 1800 and
→ "Located" in Washington.

* In 1965 :- "catalog System - DIALOG"

The earliest commercial cataloging system "DIALOG", which was developed by "Lockheed Corporation" for NASA.

* from 1966 - 1968 :-

The Library of Congress ran the "MARC I pilot project"

MARC → MAchine Readable Cataloging

It standardizes the * Structure,
* contents and
* coding of bibliographic records.

* In 1969 :-

The system (Congress Library System) became Operational in 1969.

* In 1978 :-

The DIALOG became commercial, with three government files of indexes to technical publications.

* In 1988 :-

The "DIALOG" was sold to "Knight-Ridder" at that time "DIALOG" contained "320" Index databases used by over "91,000" subscribers in 86 countries.

* In 1990 :-

The significant reduction occur in cost of Processing power and memory in modern computers along with full text of item in electronic form; from publishing stages.

Finally,

Until recent, Indexing was accomplished by creating a bibliographic citation in a structured file that refers to the Original text.

⇒ The Indexing Process is typically performed by "Professional Indexers" associated with Library Organization.

Cuneiform tablets

from the Ancients presents, clay tablets, cones and brick fragments "inscribed" using ancient writing system known as cuneiform from the Library of Congress' collections.

→ The Word "cuneiform" is derived from "Latin". Cuneus for "Wedge" and forma meaning "shape".

→ Cuneiform is a "Logo-Symbolic" script that was used to write several languages of the Ancient middle East.

→ Cuneiform was developed by the Sumerians, who thrived during the 3rd millennium B.C.

→ Sumerians influenced their culture and development beyond their original borders in mesopotamia (Present-day Southern Iraq) earliest civilization.

→ Originally Cuneiform signs were "pictograms", Later it also became Symbolic.

The materials used in Cuneiform — "clay and reeds".

→ The "clay and reeds" with are readily available.

→ "Reeds" were used as "writing implements".

A tall slender leaves of plants

→ The tip of a "reed style" was impressed into a wet clay surface to the draw strokes of the sign thus it acquiring a "wedge-shape".

i.e The Sumerians invented this writing system, which involves the use of wedge-shaped reed stylus to make impression in clay.

⇒ * The Library of congress acquired its collections of cuneiform materials in 1929 from Kirkor (an art dealer)

DIALOG :-

DIALOG Online search system - (1966 became available).

→ DIALOG was the first interactive, online search system addressing 'large databases' and allowing "Iterative refinement of results".

→ DIALOG was Developed at "Lockheed palo Alto Research Laboratory", and it extended through Contracts with NASA.

Why DIALOG became Popular:

Its speed, ease of use and wide range of data content attracted the professional users worldwide including

* Scientists
* attorneys
* educators &
* Librarians.

→ DIALOG preceded major "Internet search tools" by more than two decades.

By 1972,

The commercial introduction ~~extended~~ extended to all the professions by allowing "the Online access" to large collections of digitized materials by a way of "command language" that allowed the ~~searcher~~ searcher to interactively refine results.

DIALOG's ability to allow Interactive refinement of results:-

* DIALOG's "major technical innovation is reflected in its name."

i.e It enables a conversion between the "searcher" and the "computer".

* "Search at its best is a conversation".

i.e "An ~~it~~ iterative, Interactive Process where we find we learn".

User Characteristics of DIALOG Language:

There are 5 Important Characteristics.

1. The search question is constructed at search time.
2. Dialog is designed for nonspecialists.
   i.e It avoids Communication barrier.
3. Command language is independent of the particular data of searches.

4. As "an online-system".
   i.e It allows continual redefinition of the search question based on the examination of intermediate results.
5. Control of process lies with the user.
   i.e computer merly serves as a data-processing extension of user.

DIALOG Commands :-

The DIALOG system provides the number of Commands with which the "searcher" interact with the "Computer".

A search consist of

i) Identifying
ii) Selecting terms and phrases that reflect the user's interest.
iii) Combining descriptors into search expressions
iv) examining the retrieved citations and modifying search expressions.

functions

Each of these functions accomplished with a particular Command.

The 4 Principal Commands are:

i) EXPAND
ii) SELECT
iii) COMBINE
iv) DISPLAY

EXPAND:

EXPAND provides list of "synonyms", related terms and similar combination of words defining his/her (user) interest.

SELECT:

SELECT is used to selecting "list of terms" (i.e term a, term b, term c --), (or) "a range of terms" i.e (term a — term d) (or) a list of ranges and terms.

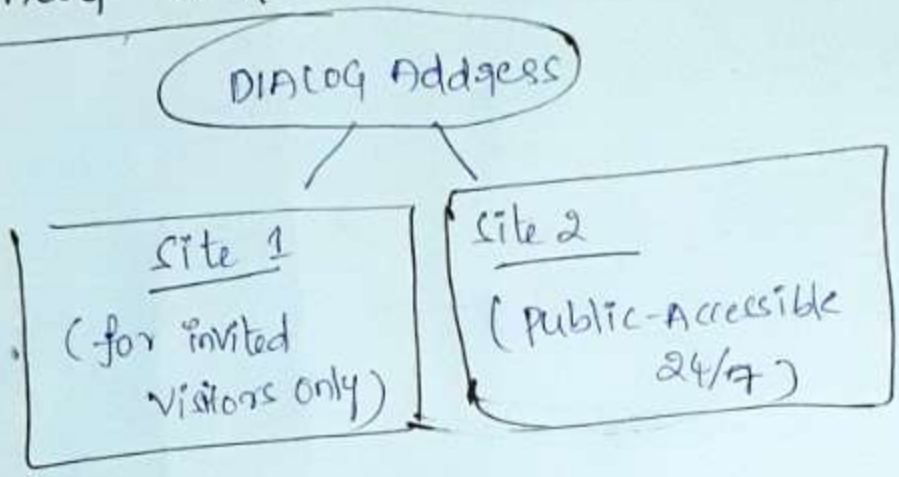
COMBINE:

The list of Boolean Expressions are used to combine the documents.

DISPLAY:

It allows the searcher to successively display the documents contained in the resultant set.

DIALOG Address :-



Site 1 :- (for invited - visitors only)
Lockheed Martin Advanced Technology center,
formely Lockheed palo Alto Research Laboratory
Bldg. 201 (front lobby site)

Site 2 : (Publicly-Accessible 24/7): mountain view, "Computer History museum", (on inside face of front patio brick wall.)

Security / protection of site1 & site2 :-

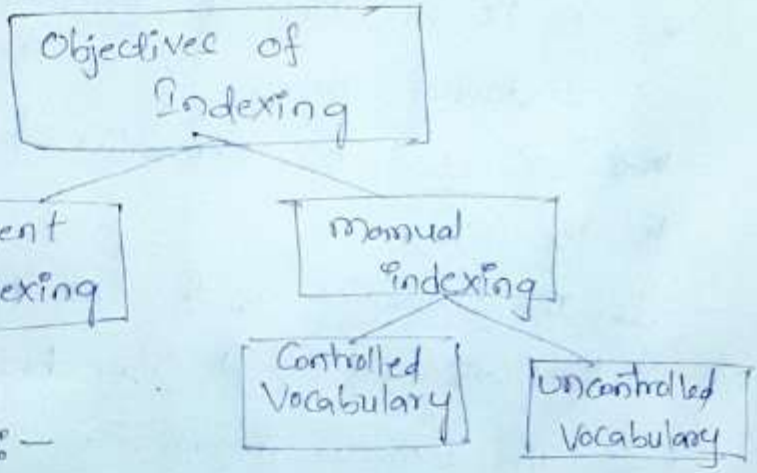
Site 1 : It requiring a specific US security clearence to access site. (with in a secure facity).

site 2 : freely accessible to the public 24/7.

Objectives :-

The Objectives of "Indexing" change with evolution of Information Retrieval system.

The Indexing Define the "Source" and "major concepts" of an item that Provide the standard Mechanism for "Index terms".



Document Indexing :-

The Document file provides the newclass of Indexing called "Document Indexing".

Manual Indexing :-

In Manual Indexing Environment,

* **Controlled Vocabulary :-**

The Use of controlled Vocabulary makes the Indexing Process "slower" but Potentially Simplifies the Searchprocess.

* **Uncontrolled Vocabulary :-**

The Use of uncontrolled Vocabulary makes the Indexing Process "faster" but Search process much more difficult.

The availability of "items in "electronic form" changes the objectives of manual indexing.

i.e The source information can be automatically extracted.

(*) Indexing Process

⇒ It is an important Process in IRS.

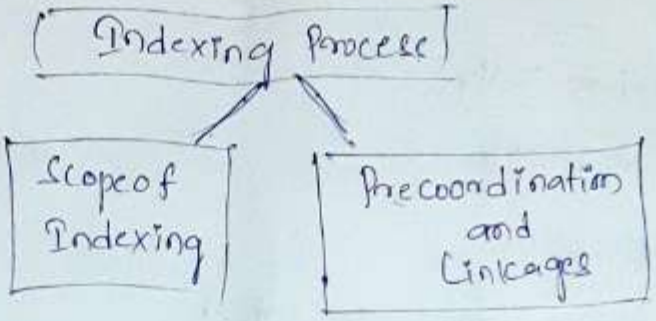
⇒ Indexing Process is a transformation of an item that "extracts" the semantics of the topics discussed in the item.

⇒ The Semantics of the item not only refers the subject discussed in the item but also weighted systems.

⇒ The extracted information is used to create "the processing tokens" and "the searchable data structures".

⇒ The index can be based on the
- full text of the item,
- automatic or manual generation of the item
- natural language representation of the item.

⇒ The Result of this Process stored in one of the data structure called "inverted data structure".



Scope of Indexing :-

When scope of Indexing performed manually, The following problems are occure with the interaction of two sources :

i. The Author
ii. The Indexer.

The Author :-

The vocabulary domain of the author may be different than that of the Indexer.

The Indexer :-

→ The Indexer is not an expert on all the areas the item is being presented and the result is in different quality levels of Indexing.

→ The Indexer must determine when to stop the Indexing process.

factors effecting the scope of Indexing:-

i) Exhaustivity
ii) Specificity

Exhaustivity:-

The extent to which
extension
→ The different concepts in the item age indexed.

Specificity:-

The preciseness (or) The exact index terms used in indexing called Specificity.

Linkages & precoordination:-

Linkages:

The Linkages are used to correlate the related attributes associated with concepts discussed in the item.

Pre co-ordination:-

The process of creating the term "Linkages" at index creation time is called precoordination.

Post Co-ordination:-

When the index terms are not co-ordinated at index time, the co-ordination occure at search time is called Post co-ordination.

The post Co-ordination is implemented by "AND" in index terms.

Example :-

The different types of Linkages as shown below. (73)

i.e

"The drilling of oil wells in Mexico by CITGO and
The introduction of oil refineries in peru by the US."

INDEX TERMS	Methodology
→ Oil, wells, Mexico, CITGO, refineries, Peru, BP, drilling.	No linking of terms.
→ * (Oil wells, Mexico, drilling, CITGO) * (US, oil refineries, peru, introduction)	linked (pre-coordination).
→ * (CITGO, drill, Oilwells, mexico) * (US, introduction, Oil refineries, peru)	linked (pre-coordination) with position indicating role.
→ * (SUBJECT: CITGO; ACTION: drilling; OBJECT: Oil wells; MODIFIER: in mexico) * (SUBJECT: U.S; ACTION: introduces; OBJECT: Oil refineries; MODIFIER: in peru).	linked (pre-coordination) with modifier & indicating role.

drill c → drilling

modifier

introduction to introduces

modifier

fig: Linkage of Index Terms

(*) Automatic Indexing :-

Automatic Indexing is "automatically" determine the "Index terms" assigned to an "item".

-Automatic Indexing is deals with "simplest case" and "complex case":

* Simplest case :-

The simplest case deals with the total document of -Indexing.

* Complex case :-

The complex case deals with the "Human Indexer".

The Human Indexer determines the "limited no. of "Index terms" for major concepts in the "item".

Result of Automatic Indexing :-

The automatic Indexing results fall into two classes :

i) Weighted Indexing system ii) Unweighted Indexing system

i) Weighted Indexing :-

→ Weighted Indexing represents the "term values" placed inside the "Index terms".

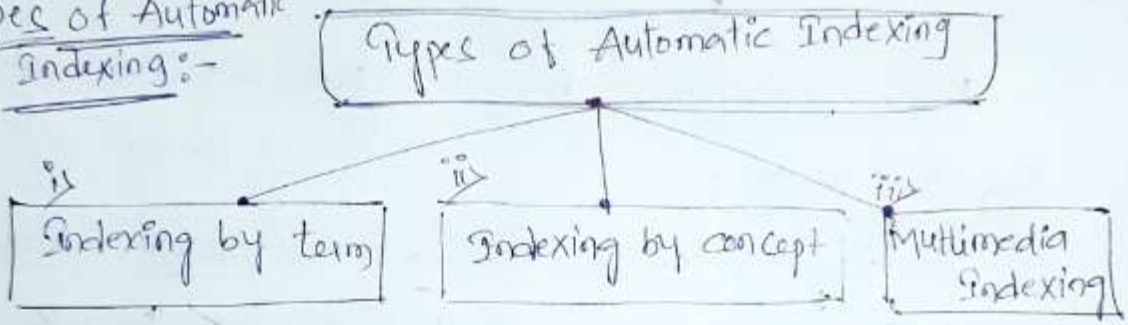
⇒ The Basic function of the weighted indexing associated with "frequency of occurrency of an "item".

ii) Unweighted Indexing :-

⇒ The unweighted indexing represents the existence of an index term in a document. (15)

⇒ In unweighted indexing some times, "word locations" are kept as a part of searchable datastructure.

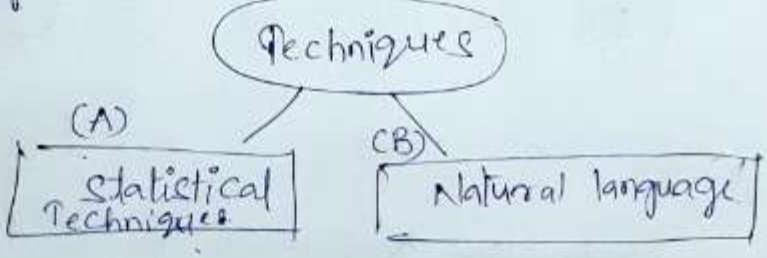
Types of Automatic Indexing :-



i) Indexing by term :-

⇒ The "terms" of original item are used as basis of "index process".

There are 2 major techniques used for creation of the index are :



(A) Statistical Technique :

The statistical techniques can be based on the "vector Models" and "probabilistic models" with a special case being is "Bayesian models".

2

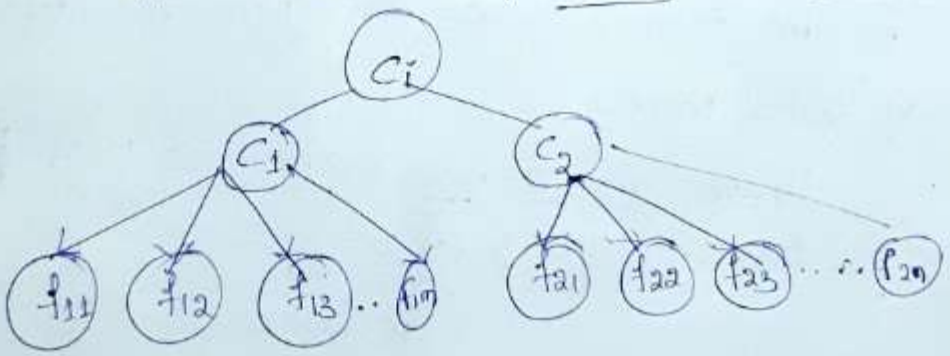
All the models (i.e Vector model, Probabilistic model and "Bayesian model) classified as "statistical" because their calculation of weights use the statistical information such as

"frequency of occurrence of words" and their distributions in the searchable database.

Bayesian Model :-

A Bayesian model is a "directed acyclic graph" in which the node indicate the random variable.

It is the combination of "nodes" and "Arcs".



where C → concept, f → features.

To calculate the probability of "G" from given "Fij".

G from given Fij

To perform the calculation 2 sets of probabilities are needed:

1. The Prior Probability" · P(Ci).

means the item is relevant to concept C.

2. The "Conditional probability"

The conditional probability is indicated by $P(f_{ij}/c_i)$.

Where, P → Probability (11)

f_{ij} → features.

C_i → Concept.

→ The features F_{ij} where $j=1,m$ are present in an item means the item contains topic c_i.

The automatic indexing task is used to calculate the Posterior probability:

i.e $P(C_i/f_{i1}, \ldots, f_{im})$ where $j=1,m$

The probability of the item that contains concept(C_i) given the presence of features(f_{ij}).

The Bayes inference / Bayesian formula used for Automatic indexing is:

$$P(C_i/F_{i1}, \ldots, F_{im}) = \frac{P(C_i)\,P(f_{i1}, \ldots f_{im}/C_i)}{P(f_{i1}, \ldots f_{im})}$$

(or)

$$P(C_i/f_{i1}, \ldots, f_{im}) = P(C_i)\,P(f_{i1}, \ldots f_{im}/C_i)\,/\,P(f_{i1}, \ldots f_{im})$$

=) The Result of search by 'posteriors', the Bayes rule can be simplified to linear decision rule:

$$g(C_i/F_{i1}, \ldots F_{im}) = \Sigma_k I(F_{ik}) W(F_{ik}, C_i)$$ (18)

Where,

'g, w' are interpreting the coefficients

i.e $w \rightarrow$ weights corresponding to each feature / Concept Pair.
Ex: index terms.

$g \rightarrow$ 'g' as the sum of the weights of the features.

- $I(F_{ik}) \rightarrow$ an indicator variable that equals 1 if F_{ik} is present in the item (otherwise it is equals zero).

$(C/F_{i1}, \ldots F_{im}) \rightarrow$ Posterior probability.

(B). Natural Language processing :—

→ The another approch to defining "indexes to items" is called Natural Language processing.

→ The DR-LINK (Document Retrieval through Linguistic Knowledge). System processes items at the morphological, lexical, semantic, syntactic and discourse levels.

Each level uses information from the previous level to perform its additional analysis.

ii) Indexing by concept:-

→ The basis for "concept indexing" is that, there are "many ways to express" the same idea.

→ Indexing by term treat each of these occurrences as a "different index" and then uses the other query expansion techniques to expand a query to find the different ways the same thing has been represented.

Concept Indexing:

Concept Indexing determines a "canonical set of concepts" based upon a test set of "terms".

The concept indexing is also called as "Latent Semantic Indexing" because it indexing the latent Semantic information in items.

In concept indexing, it does not associated with each 'concept' (ie words / set of words that can be used to describe it) but it is a mathematical representation (Ex. a vector / context vector)

→ The word items & queries are represented by high dimensional vectors called 'context vectors'. i.e (High dimensional at least 300 dimension).

Example of Concept Indexing:-

The example of Concept Indexing is the "Matchplus System" Developed by HNC Inc.

The Matchplus system uses "neural networks" to facilitate.

facilitate → "machine learning concepts of relationships" and (8)
→ "sensitivity to similarity of use.

The two neural networks are used in concept indexing, are

1. One Neural network : (Stem context vectors)

One Nural network learning algorithm generates "Stem Context Vectors" that are sensitive to similarity of use.

2. Second Neural network :

It performs query modification based upon user feedback.

~~Context vector / vectors :~~

Context Vector :

Words stems, items and queries are represented by "high dimensional" Vectors called "context vectors".

To define context vectors, a set of 'n' features are selected on an adhoc basis.

ie for any word stem k, its context vector $\underline{V^k}$ is is an n-dimensional Vector with each component 'j' interpreted as follows :

V^k "Positive" if k is strongly associated with feature j.
$V^k \approx 0$ if word 'k' is not associated with feature j.
V^k "negative" if "the" word "k" contradicts feature j.

Once the context Vectors for Stems are determined, they are used to create the index for an item.

iii) Multimedia Indexing :—

→ Indexing associated with multimedia differs from the Previous discussions of indexing.

→ The automated indexing take place in "multiple passes" of the "informations" "Versus" direct conversation to the indexing structure.

first pass : The first pass in most cases is a "Conversation" from the analog input mode into a "digital structure".

The digital structure algorithms are "extract" the different modalities that will be used to represent the item.

Next pass :—

Indexing video (or) images can be accomplished at the "raw data level".

Ex:— The aggregation of raw pixels.

i.e The Raw pixels (distinguish) contains the primitive attributes such as "color" and "luminance" and at Semantic level meaning full objects are recognized.

Ex: an airplane in image/video frame.

Video processing :-

→ The system will periodically collect a frame of video input for processing. (82)

→ It might compare that frame to the last frame captured to determine differences between the frames.

→ If the difference is below the threshold it will discard the frame.

Ex: Virage.

Image processing :-

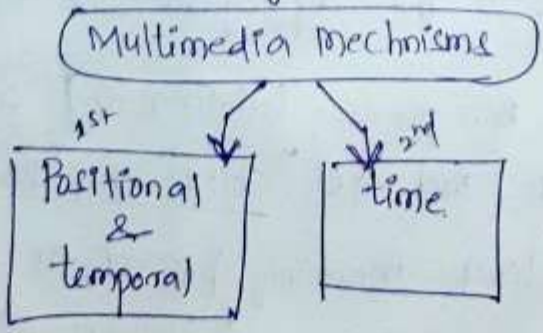
In Image Processing,

→ Semantic level indexing requires pattern recognition of objects with in the images.

Ex: • MIT's PhotoBook,
• IBM's QBIC &
• Multimedia Data Made from Informix | Virage.

finally :→ (multimedia mechanisms to store,

The Multimedia system need some mechanism to correlate the different modalities during the search.



first mechanism:

* Positional & temporal mechanism :—

This mechanism is used when the modalities are interspersed / interspersed in a linear sequential composition.

Example :—

A document that has "images" (or) "audio" inserted can be considered a linear structure.

* Second mechanism: i.e "time" :—

The second mechanism is based upon time because the modalities are executing concurrently.

i.e - Ex:- Television , SMIL

The typical Video source off television is inherently a multimedia source.

It contains Video, audio and potentially closed captioning.

SMIL → Synchronized Multimedia Integration Language.

It is a mark-up language designed to support multimedia presentations that integrate text with audio, images and video.

Extraction means → removal, taking out, drawing out, pulling out etc.

(*) Information Extraction :-

→ Information Extraction is a task of extracting the structured information from unstructured and/or semi-structured machine-readable documents.

→ There are two processes (or) cases associated with Information Extraction :

i) Determination of facts
ii) Extraction of text

i) Determination of facts :-

Determination of facts go into "structured fields" in a database.

In this case, only a subset of the important facts in an item may be identified and extracted.

ii) Extraction of text :-

Extraction of text can be used to "summarize -an item".

In summarization, all the major concepts in the item should be represented in the summary.

→ The process of extracting facts to go into indexes is called "Automatic file Build".

It's "goal" is to process "incoming items" and "extract index terms" that will go into a structured database.

→ An Information Retrieval System's "goal" is to provide an "in-depth representation of the contents of an item.

→ An Information Extraction System only analyzes those portions of a document that potentially contain information relevant to the extraction criteria.

Metrics to compare Information Extraction :-

The previously defined measures of precision and recall are applied with slight modifications to their meaning.

Precision :- Precision refers to how much information was extracted accurately Versus the total information extracted.

Recall :- Recall refers to how much information was extracted from an item Versus how much should have been extracted from the item.

i.e It shows, The amount of correct and relevant data extracted Versus the correct and relevant data in the item.

Additional Metrics :-

- Overgeneration
- Fallout

The Additional Metrics used for information Extraction are : "Overgeneration" & "fallout".

Overgeneration :—

Overgeneration measures the amount of irrelevant/non-relevant information that is extracted.

Fallout :—

Fallout measures how much a system assigns incorrect slot fillers, as the number of potential incorrect slot fillers increases.

These measures are applicable to both "human" and "automated" extraction process.

⇒ The best source of analysis of data extraction is from the "Message Understanding conference proceedings".
⇒ The conferences were held in 1991, 1992, 1993 & 1995.
⇒ The conferences are sponsored by the
- Advanced Research Project Agency :

Objective of data Extraction :— "Slot"

Objective of data extraction is update structured database with additional facts.

⇒ The term "slot" is used to define a particular category of information to extracted.
⇒ Slots are Organized into templates / semantic frames.
⇒ Information extraction requires multiple levels of analysis - of text of the item.
ie It must understand the words and their context.

⇒ This processing is very similar to Natural language processing.

Data structure

(1) Introduction to Data structure :— (87)

→ A Data structure is a "storage" that is used to store and Organize data.

It is a way of arranging data on a computer so that it can be accessed and updated efficiently.

There are usually "two major" data structures in any information system.

first/One major Data structure :—

→ One major Data structure "stores and manages" the received items in their "Normalized form".

→ The process of supporting this structure is called the "document Manager".

Other Major Data structure :—

Other major Data structure contains the "processing tokens", and "associated data" to support search.

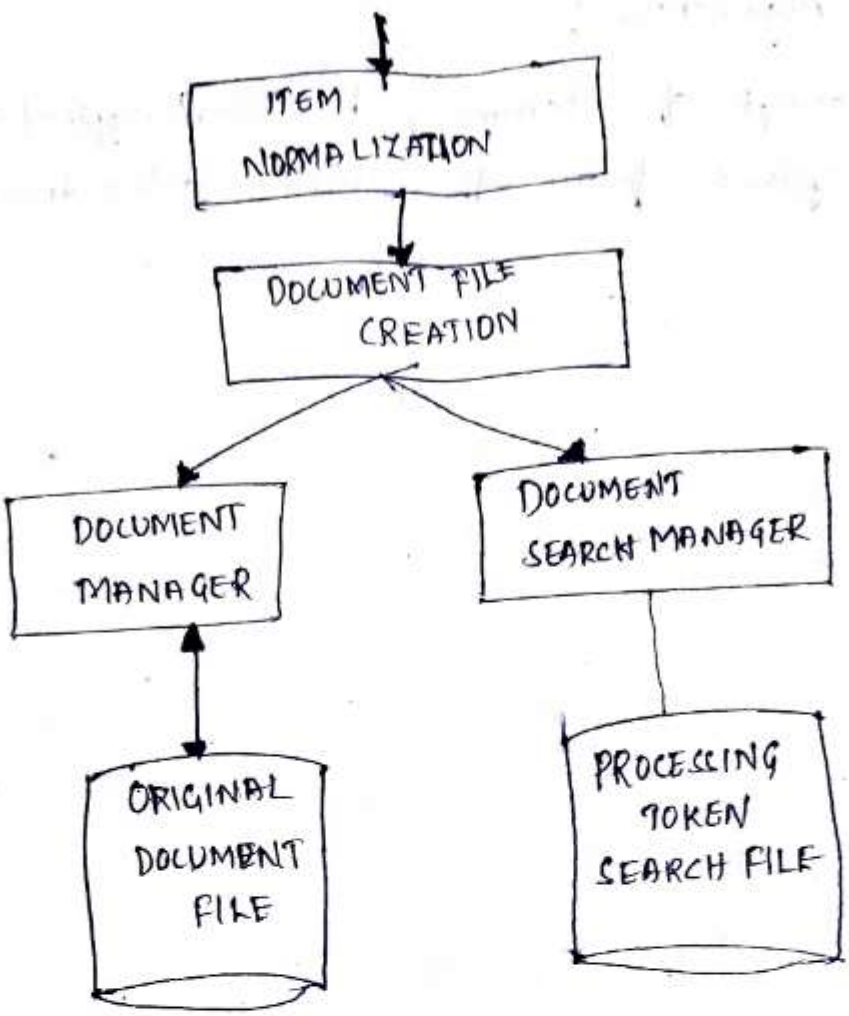


fig: Major Datastructures

→ The above figure expands the document file creation functionchene discussed in 1st chapter).

→ The Results of a search are references to the items that satisfies the search statement, which are passed to the document manager for retrieval.

→ To understand this datastructures background, the reader / user should pursue a text on finite automata and language (regular expressions).

Scanned with ACE Scanner

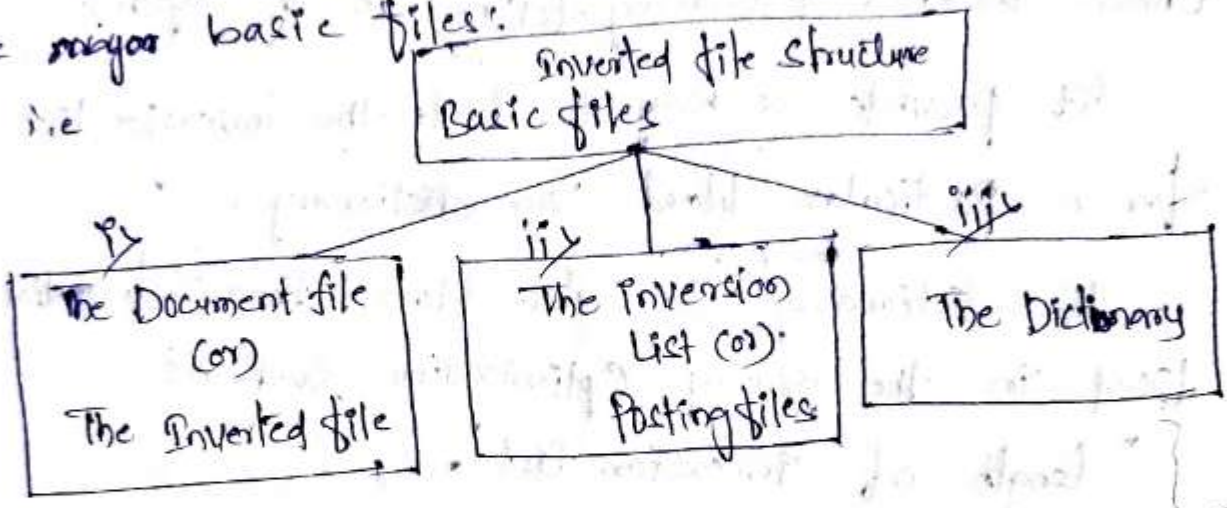
(*) Inverted file structure:-

The most common data structure used in both Database management and Information Retrieval systems is the Inverted file structure.

It is used to minimize the secondary storage -access when multiple search terms are applied across the total database.

The Inverted file structures are composed of three major basic files:

i.e



i) Inverted file / Document file:-

The name Inversion "inverted file" comes from its underlying methodology of storing an inversion of the documents.

from the inversion of document perspective, for each word a list of documents in which the word is found.

ii) The Inversion list :-

Inversion list sometimes called as "posting list". In this each document in a system give a "Unique numerical Identifier".

The identifier is stored in the inversion list.

iii) The Dictionary :-

The Dictionary is typically a stored list of all Unique words (processing tokens) in the system.

It provide a way to locate the inversion list for a particular word via dictionary.

The dictionary can also store other information used in the query optimization such as "length of inversion list".

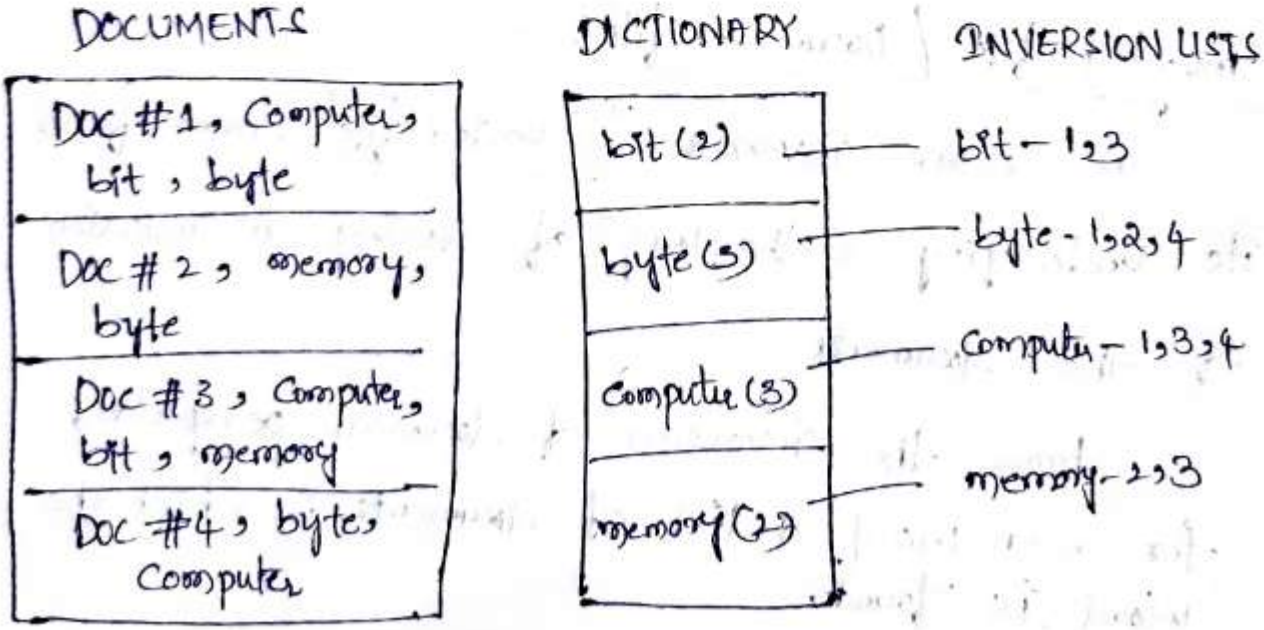


Fig : Inverted file Structure

Scanned with ACE Scanner

from the figure diagram :-

→ The Documents contains information of words (such as bit, computer, bytes) i.e how many words are there in a particular documents.

→ The Dictionary contains how many number of times the particular word is appear in the document is written inside paranthesis ().

→ The Inversion List contains in which document the particular word is appearing.

finally,

The inverted file structure uses some silent features i.e

1. It increases precision and Recall.
2. It also uses browse capabilities such as Ranking and Zoning.
3. It uses the NLP (Natural Language processing)
4. It is mainly used to store the concepts and Relationships.
5. Rather than using a dictionary to point the inversion list B-Trees can be used.

Scanned with ACE Scanner

In B-Trees inversion list may be appear at "leaf level / leaf node".

A B-tree of Order "m" is defined as:

* A "root node" with between "2 and 2m keys".
* All other internal nodes have between "m" and "2m" keys.
* All keys are kept in order from smallest to largest.
* All leaves are at the same level (or) differ by at most one level.

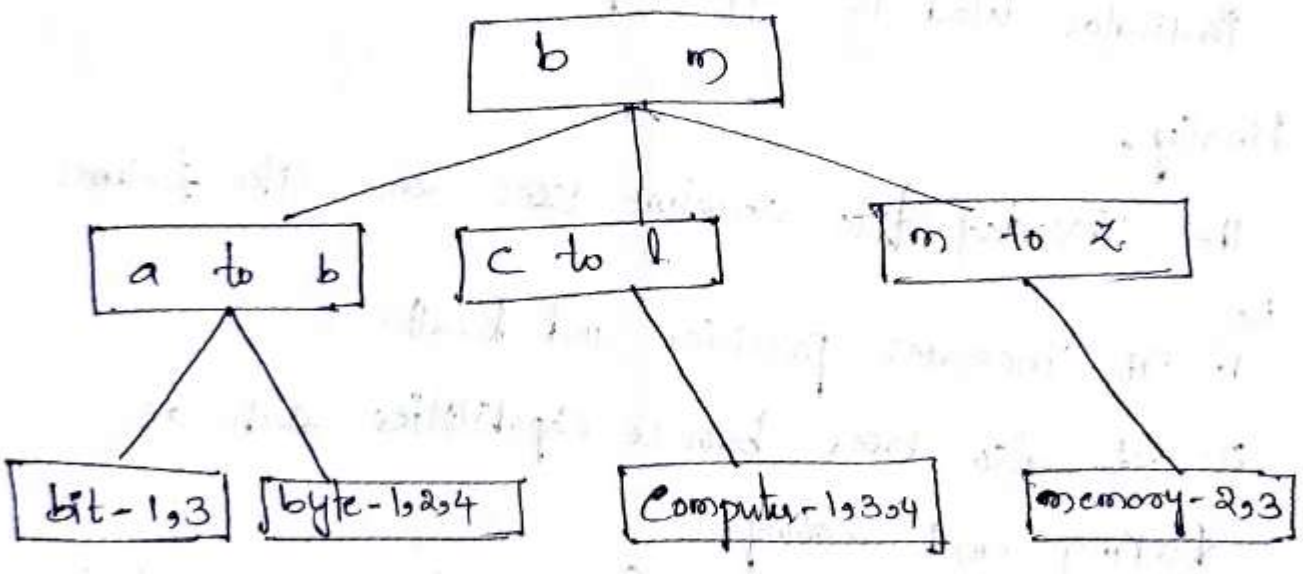


Fig: B-Tree Inversion list.

(*) N-Gram Data structures :-

⇒ A Variant of the "Searchable data structure" is N-Gram Data Structure, that breaks processing tokens into smaller string units and uses the token fragments for search.

⇒ N-grams can be viewed as a special technique for conflation (stemming). It is used as a unique data structure in Information systems.

⇒ N-grams are "fixed-length" consecutive series of "n" characters.

Unlike stemming,

i.e The stemming determines the stem of a word that represents the semantic meaning of the word working => work

but n-grams do not care about semantics.

Instead of that, they are {algorithmically} based upon a fixed number of characters.

⇒ The searchable data structures is transformed into Overlapping n-grams, which are then used to create the searchable data base.

Scanned with ACE Scanner

Examples : Bigrams, Trigrams and pentagrams for the word phrase "sea colony".

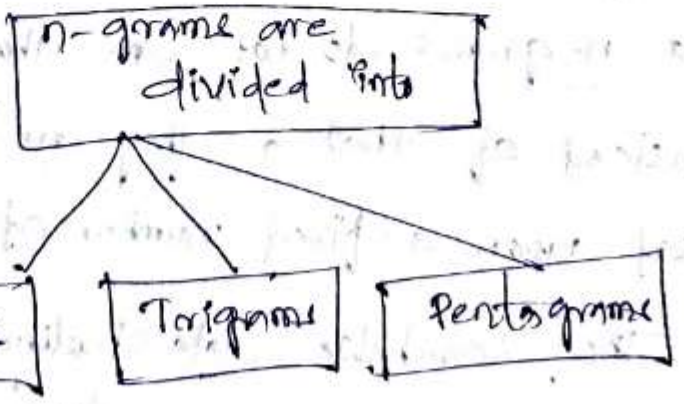
⟹ for n-grams with $n > 2$, some systems allow Interword symbols to be part of the n-gram. The symbol "#" is used to represent the Interword symbol.

i.e # ⟹ Interword Symbol

⇒ The # may be represented any set of symbols such as "blank, period, semicolon, colon" etc.

⇒ It is possible that same "n-gram" can be created "multiple times" from a single word.

⇒



For example :-

Bigrams, trigrams and pentagrams for the word phrase "Sea colony".

* Bigrams for word "Sea colony"

Word		Bigram
Sea	→	se ea
colony	→	co ol lo on ny

(Bigrams no inter word symbols)

∴ se ea co ol lo on ny

* Trigrams for word "Sea colony"

Word		Trigram
Sea	→	Sea
colony	→	col olo lon ony

(Trigrams no inter word symbols)

ie ∴ Sea col olo lon ony

* Pentagrams for word "Sea colony"

Word		Pentagram
Sea	→	Sea (no change)
Colony	→	colo olon lony

(Pentagrams no inter word symbols)

∴ Sea colo olon lony

* Trigrams with Interword Symbol "#" for word "Sea colony"

Place the Trigrams inside bigrams and place # interword symbol at both ends.

i.e # Se sea ea → bigrams
bigram → # se sea ea #. → Inter word Symbol
inter word symbol

co col olo lon ony ny

co col olo lon ony ny

∴ # se sea ea # # co col olo lon ony ny #

* Pentagrams with interword Symbol "#" for word "Sea colony"

Place the Pentagrams inside trigrams and place # interword Symbol at both ends.
above trigrams
4-grams

sea

i.e # sea #

colony ⇒ colo olon lony → 4-grams

colony ⇒ colon olony → 5-grams.

colo colon olony lony #.
4-grams 5-grams 5-grams 4-grams

∴ # sea # # colo colon olony lony #
4-grams 5-grams 5-grams 4-grams

Scanned with ACE Scanner

Example 2 Retail Details find Bigrams, trigrams & Pentagrams.

⊛ Uses of N-Grams :-

① The first Use of N-grams are at the time of "World War-II", it was used by "crypto graphers".

The Crypto graphers namely "Fletch pratt" states / Defines the "bigram and trigram tables" so that any Cryptographer can dismember (taken by Partition/divides) "simple substitution cipher".

Another Crypto grapher namely "Adamson" describes the "Use of Bigrams", as a method for "conflating terms" (or) stems.

i.e The conflating terms doesnot follow normal definition of stemming because the "Stemming" has "semantic meaning" where as "n-grams" as "no semantic meaning"

② Another Major Use of N-grams (particulary trigrams) is in "Spelling error detection and correction".

i.e

Scanned with ACE Scanner

i.e There are 4 Catagories of errors.

Error Category	Example → Computer
* Single character Insertion	Computuer → Insertion of single character i.e 'u'
* Single character Deletion	Compter → Single character deletion i.e 'u'
* Single character Substitution	Compiter → Substitution i.e 'u' → 'i'
* Transposition of two Adjacent Charaters	Computer → computer (u ⇄ t)

fig: Categories of Spelling Errors.

(3) N-Gram patterns also used for "Identifying the language of the/an item".

(4) Trigrams have been used for "text compression to manipulate the length of index terms".

(*) Advantages & Disadvantages of N-grams:-

Advantages:-

* Longer n-grams, ignoring the word boundaries.
* Theorem: A finite limit on the no. of Searchable tokens.

$$\boxed{\text{Max seq}_n = (\lambda)^n}$$

Where, Max seq_n ⇒ Maximum no. of Unique "N-grams" that can be generated.

Scanned with ACE Scanner

i.e
Max seq → calculated as a function of 'n'.
n → Indicates the length of N-gramme.
λ → Indicates the no. of Processible symbols from the alphabet.

Disadvantages :-

i) The disadvantage of N-gram is "Increased size of inversion list" that store the linkage datastructure.

ii) N-gramms are poor representation in "Concepts and their relationships".

(*) PAT Data Structure

The name PAT is the short form for Patricia trees. PATRICIA

PATRICIA stands for

P → Practical
A → Algorithm
T → To
R → Retrieve
I → Information
C → Coded
I → In
A → Alphanumerics.

i.e Practical Algorithm To Retrieve Information Coded In Alphanumerics.

Scanned with ACE Scanner

⇒ The PAT Data structures were described by

Frakes - 92
Gonnet - 83
Knuth - 73 &
Morrison - 68 has "Patricia trees".

⇒ Concept:

The Original Concepts of PAT tree data structure were described as "patricia trees", and they are used for "searching" text and "images".

These applications are in "Genetic database".

The PAT datastructure represent the information in 2ways.

ie * PAT trees &
* PAT Arrays.

PAT trees and PAT arrays are addressing the different View of continuous "text ~~ie~~ Input".

The Input stream transformed into a searchable data structure consisting of "substrings".

Creation of PAT trees:—

In creation of PAT trees each position in the Input String is the "Anchor Point" for a sub-string.

The substring start at "Anchor point" and include all new text upto end of the Input.

Scanned with ACE Scanner

→ All substrings are "Unique".

Substring :-

* A substring can start at "any point in text" and uniquely indexed by its "starting location" and length.
* If all the strings are to the end of the input, Only the starting location is needed. Because the length is different from the location and the total length of the item.
* It is possible to have a substring go beyond the length of the input stream by adding the additional null characters.

These substrings are called sistring. Sistring means (Semi-infinite String).

Example : Some possible sistrings for an "input text" given below.

TEXT :	Economics for Warsaw is Complex.	→ with space count
sistring 1	Economics for warsaw is Complex.	→ without space count
sistring 2	conomics for warsaw is complex.	
sistring 5	omics for warsaw is complex.	
sistring 10	for warsaw is complex. (without space count)	
sistring 20	w is complex. (with space count)	
sistring 30	ex. (with space count).	

fig: Examples of sistring.

In PAT trees,

* The key values are stored at the "leaf nodes" in the PAT tree.

for a text input of size "n" there are "n" leaf nodes and "n-1" at most higher level nodes.

Example of the Sistrings used in generating a PAT.

Input: 100110001101 (∵ Binary representation of HOME

H = 100
O = 110
M = 001
E = 101

i.e 100110001101
H O M E

Sistring 1	1001....
Sistring 2	001100.... (2nd place)
Sistring 3	01100.... (3rd place)
Sistring 4	1100....
Sistring 5	1000....
Sistring 6	000....
Sistring 7	001101
Sistring 8	01101

fig: Sistrings for input "100110001101".

→ the word "home" produces the input

∴ 100110 001101

⇒ No. of sistring = 8 (node's)

i.e $\boxed{n=8}$

⇒ We get $\boxed{n-1}$ = levels

8-1 = 7

i.e for 'n' no. of node. we get the 'n-1' levels

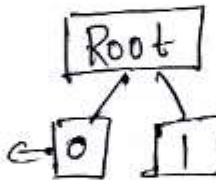
Rules for PAT Tree :—

The individual bits of sistring decides the branching

→ The pattern with Zero (0-bit) branching 'left'.

→ The pattern with one (1-bit) branching 'Right'.

i.e



Note

Take any query | sistring to reach external node.

query = 001100 (6 bits)

1st bit first

We first inspect bit-1 (if it is '0', branching will be done at left)

then bit-2 (If it is '0' branching will be done at left)

bit-3 (if it is '1' branching will be done at Right)

bit-4 (If it is '1' branching will be done at right)

bit-5 (if it is '0' branching will be done at left)

Once we reach desired node we have to make one final comparision with one of sistring.

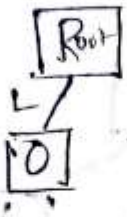
Scanned with ACE Scanner

i.e Example: INPUT : 100110001101

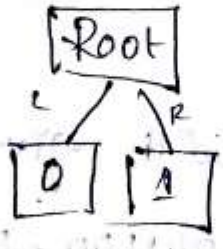
nodes: n = 8 , n-1 = 7 (levels)

i) Initial Tree contains 'Root - R'.

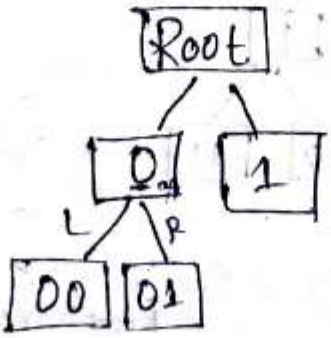
ii) add "0-bit" as left branching



iii) add "1-bit" as Right branching:

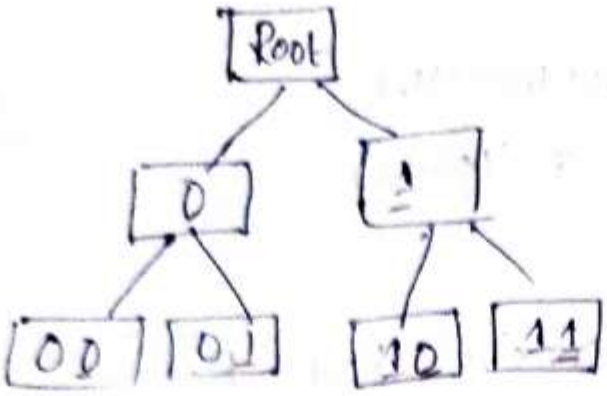


iv) for desired position '0' (Zero) compare with two bits i.e again Zero & one.

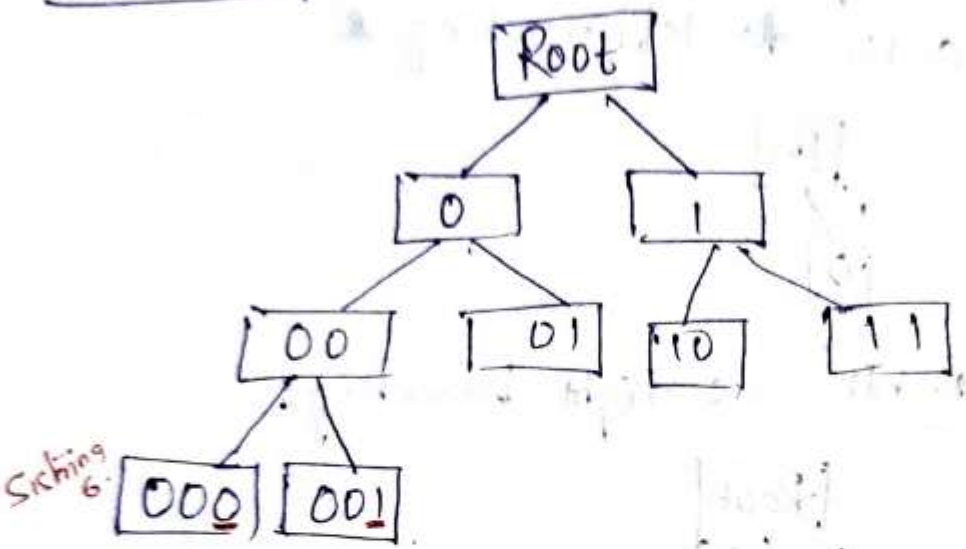


v) Similarly for 1 (Right most bit.) compare with two bits i.e again Zero & one.

Scanned with ACE Scanner



from Given sistrings we are going to construct PAT tree

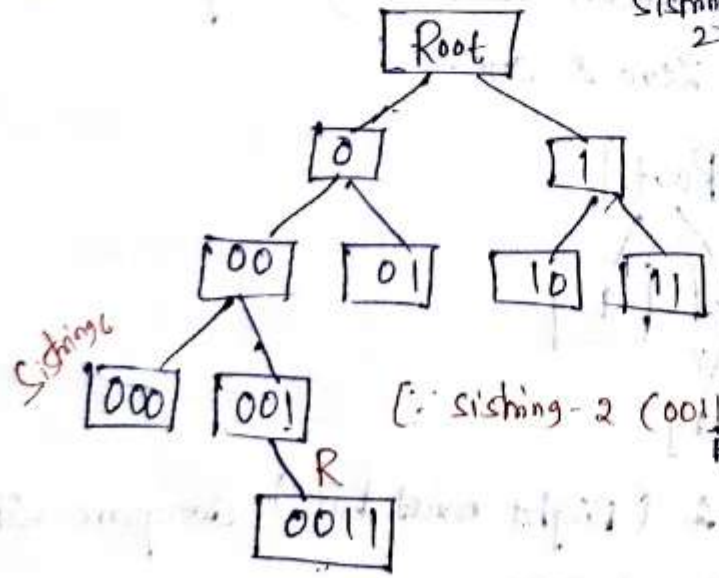


→ Now Compare 001 encode with given sistrings.

In given input sistrings : 001100, 001101

sistring 2 sistring 7

i.e

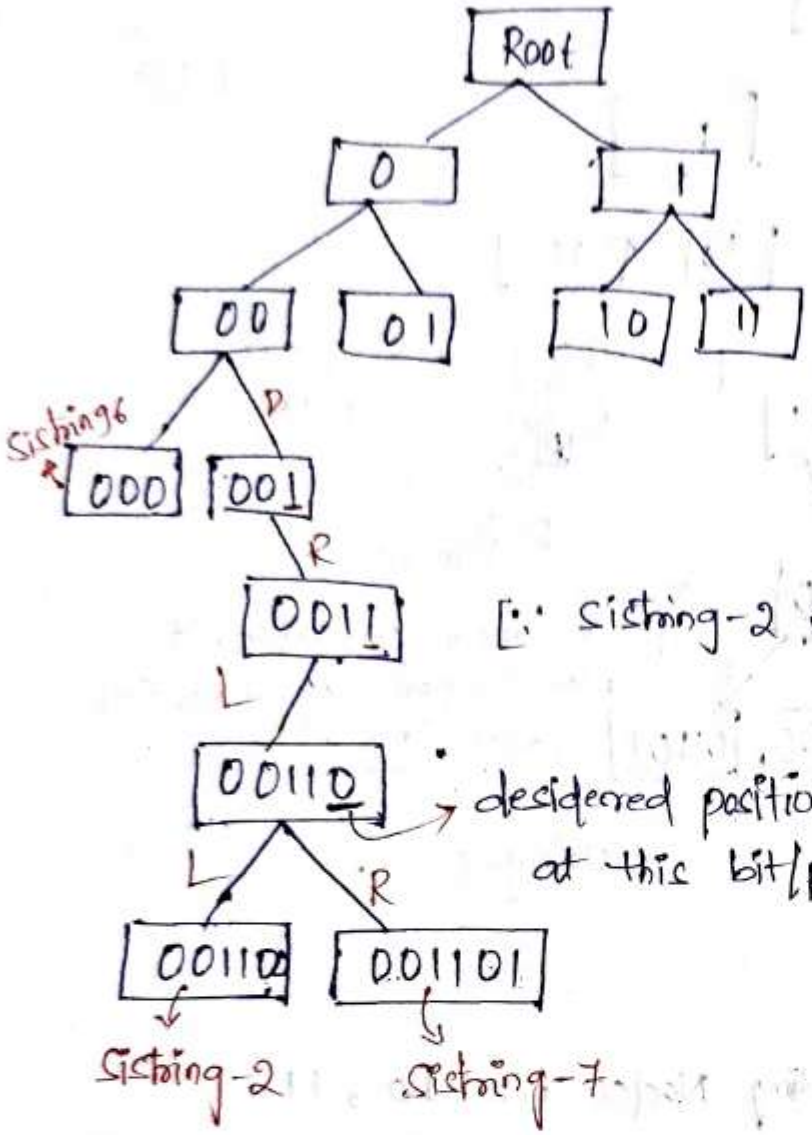


∴ sistring-2 (001100) - Right move

∴ sistring-2 001100 move Right

Next compare remaining bit in sistring 2. i.e 001100 Left (zero) move

Scanned with ACE Scanner



[∵ sistring-2 i.e 001100
↓ Left move

desidered position" i.e before last bit
at this bit/place compare both bits "0 & 1"

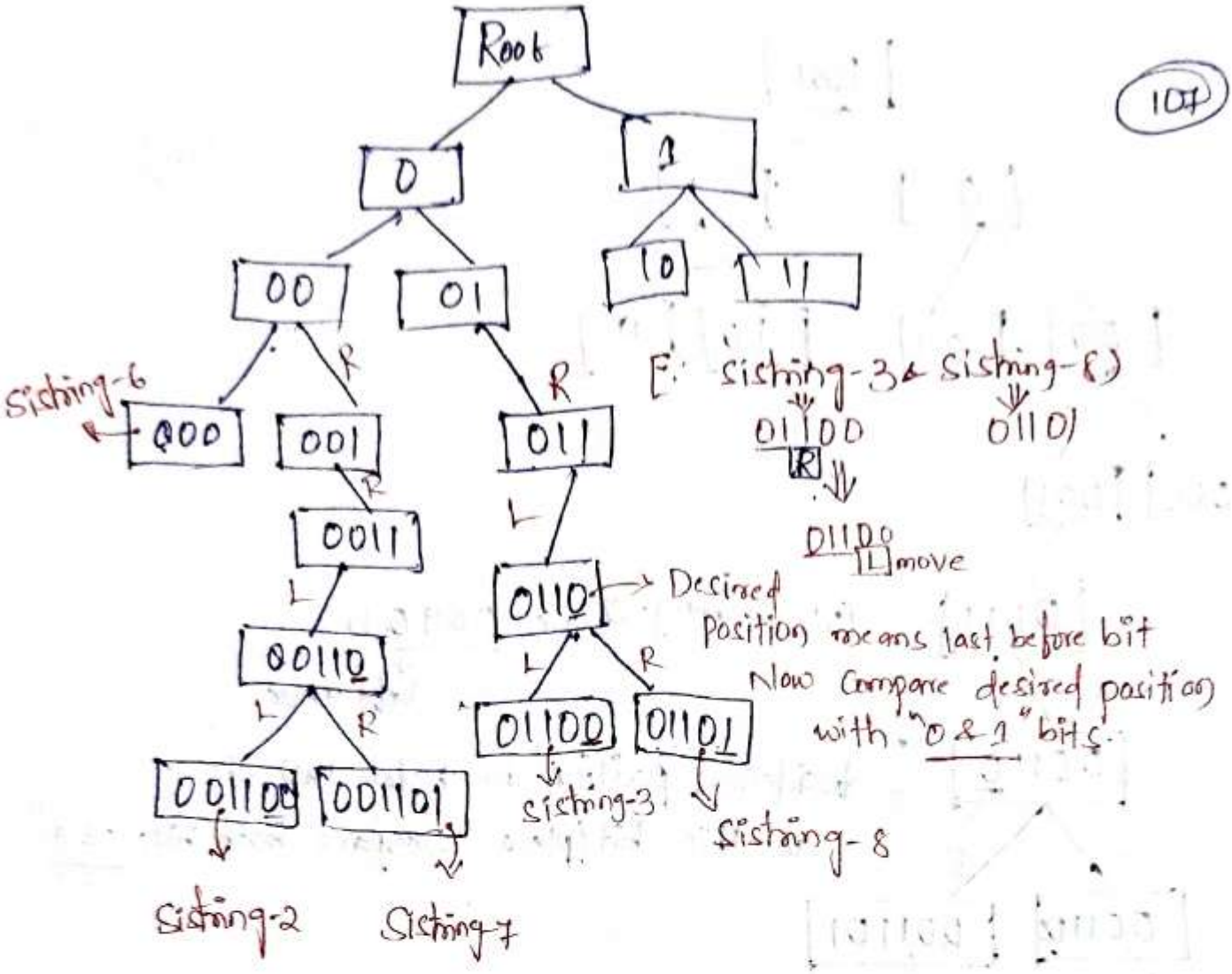
⇒ Similarly Compare Remaing Nodes. ~~Now Comp~~.
i.e (01, 10, 11)

Now compare 01 node :-

→ check is there any sistring start with "01"
at sistring-3 (i.e 01100) & sistring-8 ~~(i.e~~ (01101)
~~start~~ start with 01.

Now construct Remaing PAT Tree.

Scanned with ACE Scanner



⇒ Similarly Compare Remaing Nodes i.e 10, 11.

Now Compare "10" Node.

* Check is there any sistring start with 10.
* At Sistring-1 (1001) and Sistring-5 (1000) are start with "10".

Now construct Remaing PAT tree.

Scanned with ACE Scanner

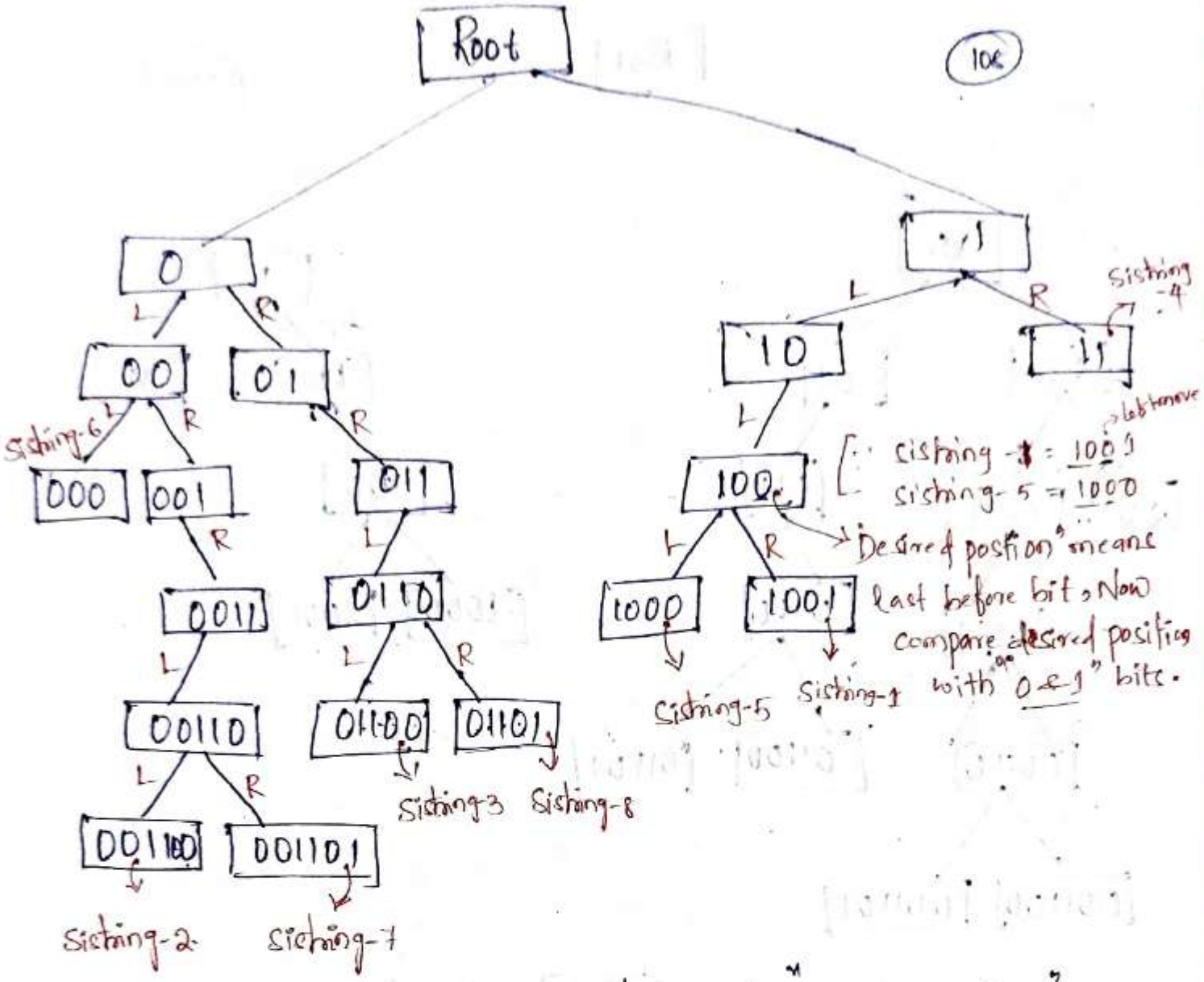
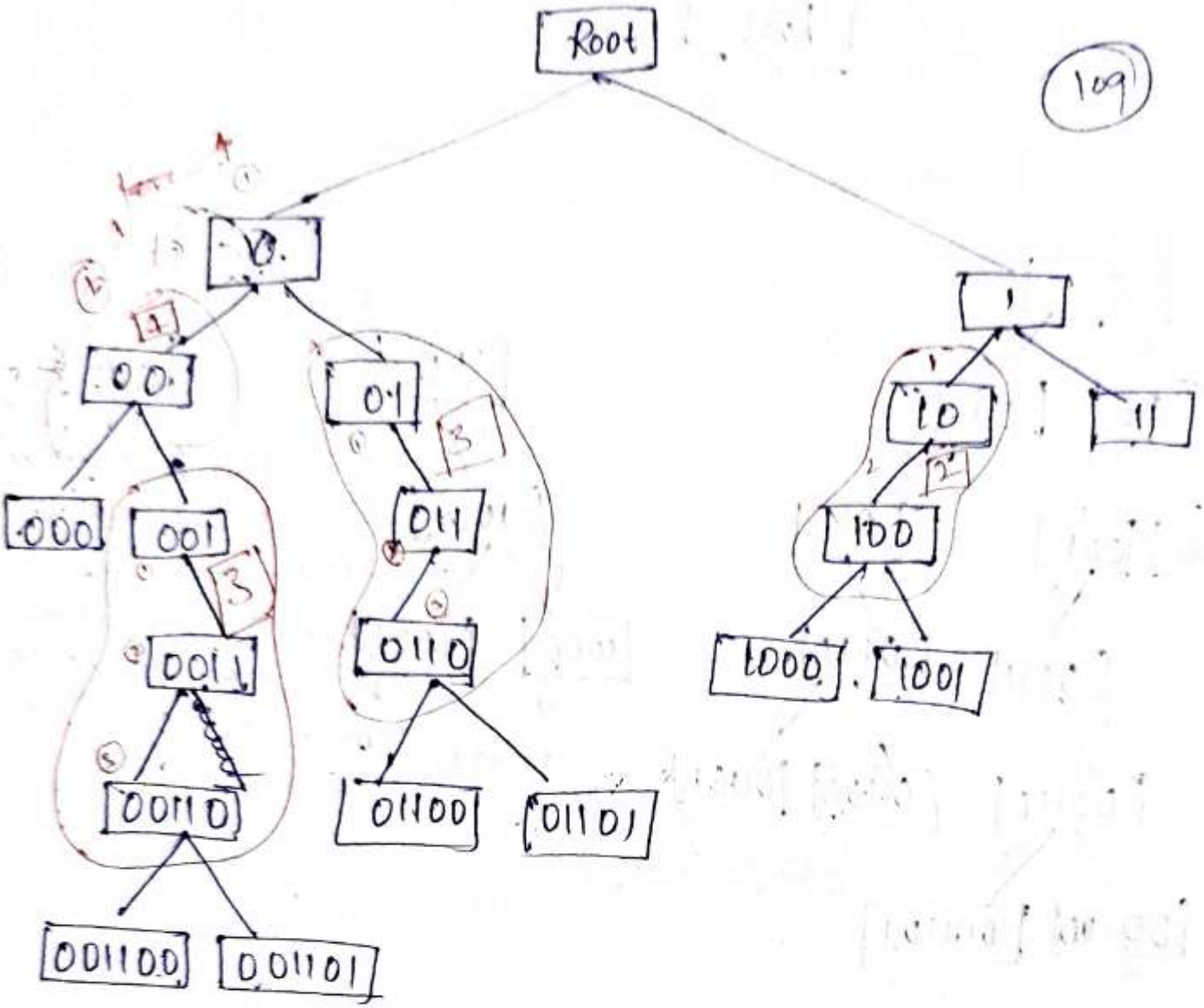


fig: PAT Binary Tree for input "100110001101"

⇒ for the above PAT tree ("skip-bits") some bits are skipped through its path lengths.



⇒

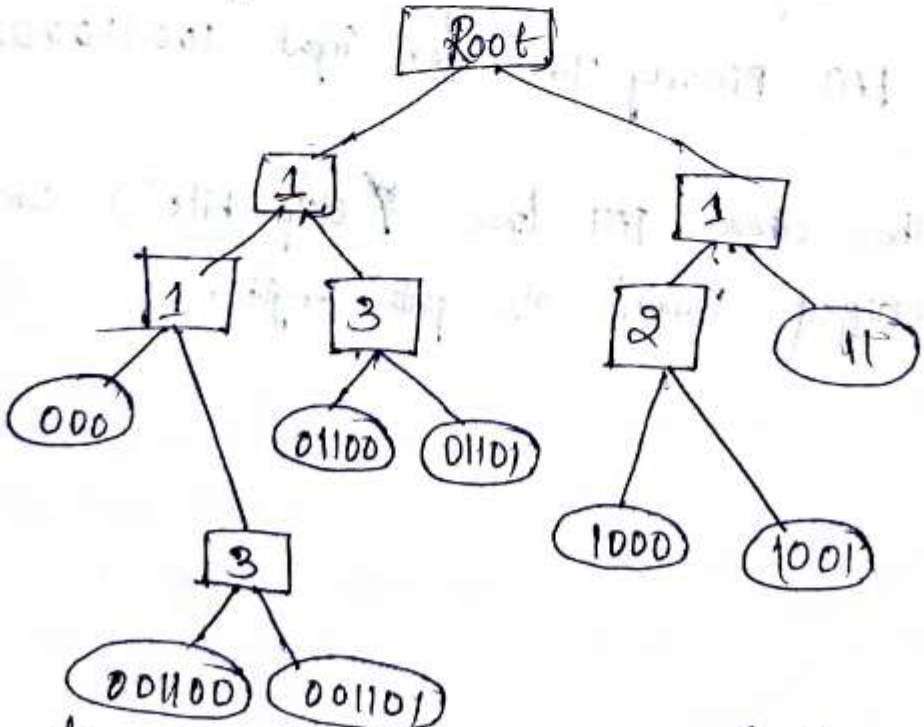


fig: PAT tree skipping bits for "100110001101"

Scanned with ACE Scanner

(*) Signature file structure :-

A signature is a compressed version of a database.

All the signatures that represent the documents are kept in a file called "Signature file".

Goal :

The Goal of a signature file structure is to provide a "fast test" to eliminate the majority of items that are "not related" to a query.

The items that satisfies the test can either be evaluated by another search algorithm to eliminate the additional false delivered to the user to review.

The text of the items is represend in a highly "Compressed form" that facilitate the fast test.

Because,

file structure is highly compressed and "unordered," so it requires significantly less space than inverted file structure, and

New items can be concatenated to the end of the structure 'versus" significant inversion list update.

In this, the items are deleted form information database marked as deleted.

Scanned with ACE Scanner

Signature file Search is a "linear scan" of compressed version of items that producing the response time linear with respect to file size.

Creation of Signature file:-

The signature file typically uses the "Superimposed coding" to create the signature of the document.

The coding is based up on the words in the item. The words are mapped into "word signatures".

Word Signatures—

A word signature is a "fixed-length code" with a fixed number of bits set to "1".

The bit positions that are set to "1" are determined via a "hash function" of the word.

The longer to long sing. signatures with "1" partitioned into "blocks" of size.

for Example:— The Block size is set at five words, the code length is "16-bits" and number of bits that are allowed to be "1" for each word is five.

i.e here * Block size is indicated by (D) → distinct non common words
* Code length / fixed length indicated by (F)
* Bits per word is set to one indicated by (on)

Scanned with ACE Scanner

method-I

TEXT : Computer science graduate students study (assume block size is five words).

WORD	Signature (4 bits)			
Computer	0001	0110	0000	0110
Science	1001	0000	1110	0000
graduate	1000	0101	0100	0010
students	0000	0111	1000	0100
study	0000	0110	0110	0100
Block signature :	1001	0111	1110	0110

fig: superimposed coding.

i.e,

Here (i) Block size = "5" five words

five words are : (1) computer, (2) science, (3) graduate, (4) students, (5) study.

(ii) code-length = "16-bits"

$16 = 2^4$

so bit length is "4"

(iii) no. of bits allowed for "1" is "5" (no. of bits per word)

* The search of the signature file requires O(N) search time.

Scanned with ACE Scanner

for Example:2 II method

TEXT : Data Base Management System (assume block size is 4)

code length / fixed-length of bits = 20 bits

WORD	Signature				
Data	0000	0000	0000	0010	0000
Base	0000	0001	0000	0000	0000
management	0000	1000	0000	0000	0000
system	0000	0000	0000	0000	1000
Block Signature	0000	1001	0000	0010	1000

Here

D → Distinct non-common Words $D = 4$

They are : Data, Base, management, system

F → Signature size / fixed-length bits.

$F = 20$ bits

m → (How many bits set to 1 } no. of bits per Word).

$m = 4$

⇒ The signatures have been used in following environments:

i) PC-based medium size databases.

ii) WORM devices Write Once Read many (optical disks).

iii) Parallel machines.

iv) distributed Environments.

(*) Hypertext and XML Data structures :

Subtopics
- i) Definition of Hypertext structures
- ii) Hypertext History
- iii) XML -

* Hypertext document is a "Non-linear" document with many links.
* Father of hypertext is "Vannevar Bush" - 1985.
* He wrote an article in 1945 on Hypertext.

Introduction & Basic points :

The advent / arrival of the Internet and its exponential "Growth" and "wide Acceptance" has introduced new mechanisms for representing information.

This structure is called "Hypertext" and it is different from traditional information storage data structure in format and use.

The Hyper text is stored in Hypertext Markup Language (HTML) and extensible Mark up language (XML).

i) Definition of Hypertext :-

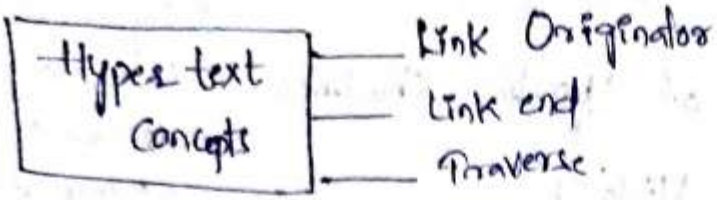
The Hypertext data structure is used extensively in the Internet environment and it requires an electronic media storage for the files. (or)

⇒ The HTML defines the internal structure for information exchange across the world wide web on the Internet.

⇒ The HTML document is composed of text of the files along with HTML tags that describes how to display the document.

⇒ Tags are expressed b/w "<" and ">" symbols. i.e <head> ... <body> ... </body> </head>

Hypertext concepts :



Link Originator :-

* It is the starting point to the link.
* It is surrounded by the " symbols" to indicate that it is a
* hypertext link.
* The link-Originator is an anchor.

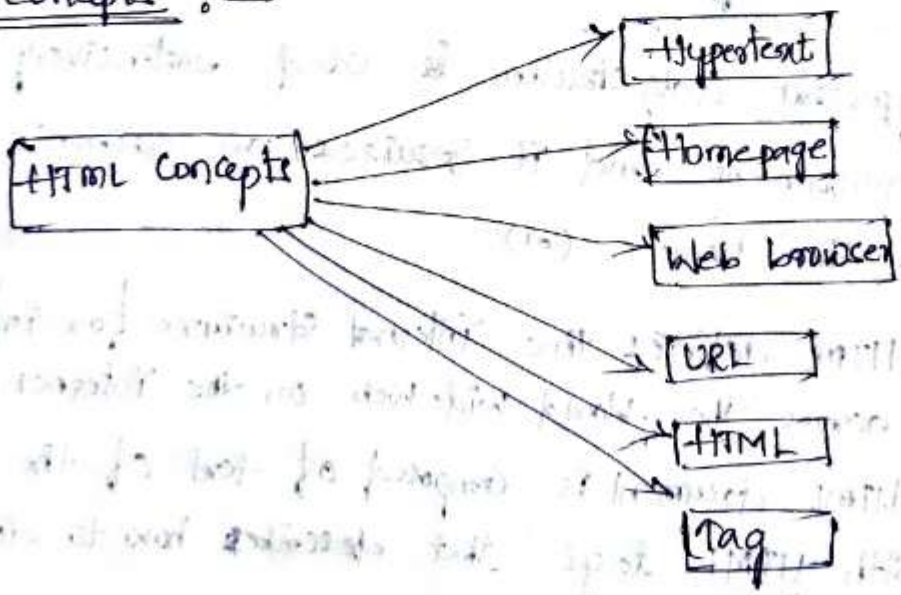
Link end :-

The other side of the link Originator is link end.

Traverse :-

The acting of travelling from a link Originator to its associated link end.

HTML concepts :-



Scanned with ACE Scanner

Hypertext :-

Text with links the other text.

Homepage :-

The Root of the Hypertext structure.

Web browser :-

A tool to retrieve a document using "URL" from the "Web server".

URL :- Uniform Resource Locator.

The address of Webpage is called "URL".

HTML :- Hypertext Markup Language.

HTML the language is used to design the webpage.

Tag :-

HTML is made with the "tags" and they are enclosed in "angular brackets" i.e < >.

iii) XML :-

The extensible markup language is starting to become a standard datastructure on the web.

Its first recommendation (1.0) was issued on February 10th 1998.

Hyperlinks for XML are being defined in the "XLink" (XML linking Language) and XPoint (XML pointer language) Specifications.

Scanned with ACE Scanner

XML Objective is extending HTML with Semantic Information.

The logical data structure within XML defined by a (DTD) Data Type Description

XML was designed to store and transport the data.

→ It is designed to be both human and machine readable.

The W3C (World Wide Web Consortium) is developing HTML as a suite of XML tags.

The following is a simple example of XML tagging:

<company> Wirdgets</company>
<City> Hyd </city>
<state> TS </state>
<product> Wirdgets </product>.

The W3C is also developing a Resource Description format (RDF) for representing Properties.

Scanned with ACE Scanner

(*) Hidden Markov Model :- (HMM)

In Hidden Markov Model the data is hidden to you / unknown to you.

Q : Why it is called - Hidden Markov model ?

A : The reason is to constructing an interface model based on the assumptions of Markov process.

The Markov process assumption is simply that "future is independent of the past given the present".

* Andrei Andreyvich Markov (1856-1922)

Andrei Andreyvich markov was a Russian Mathematician. He is best known for his work on stochastic processes. (→ random probability distribution)

A Primary Subject of his research later become known as "Markov chains" and "Markov process".

⇒ Hidden markov models (HMM) have been applied for the last 20 years to solving problems in "Speech recognition", and lesser extent in the areas

* locating named entities (Bikel-97),
* Optical character recognization (Bazzi-98) &
* topic identification (Kubala-97).

Scanned with ACE Scanner

⇒ More recently HMM's have been applied to information retrieval search with good results. (119)

⇒ One of the first comprehensive and practical descriptions of Hidden Markov Models was written by Dr. Lawrence Rabiner (Rabiner-89).

The easiest way to understand HMM is by an example.

Example:

Lets take the example of three State Markov Model of the "Stock Market".

The States will be one of the following that is Observed at the closing of the Market.

i.e State 1 (S_1): Market Decreased.

State 2 (S_2): Market did not change.

State 3 (S_3): Market Increased in Value.

The movement between "States" can be defined by a "State transition matrix" with "State transitions" (this assumes you can go from any state to any other state):

$$A = \{a_{ij}\} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \cdots \\ a_{21} & a_{22} & a_{23} \cdots \\ a_{31} & a_{32} & a_{33} \cdots \end{bmatrix}$$

Scanned with ACE Scanner

$$A = \{a_{ij}\} = \begin{bmatrix} 0.5 & 0.3 & 0.4 \\ 0.1 & 0.6 & 0.3 \\ 0.6 & 0.4 & 0.5 \end{bmatrix}$$

Given that the Market fall (fell) on one day (state-1), the matrix suggests that, the probability of the market not changing the next day is "0.1".

then it is allow questions such as

The probability that the Market will increase for the next 4 days then fall.

This would be equivalent to the sequence of

$$\boxed{SEQ = \{S_3, S_3, S_3, S_3, S_1\}}.$$

In Order to simply this model, Lets assume that instead of the current state being dependent upon all the previous states, Lets assume it is only dependent upon last state. (This order is called discrete, first order markov chain).

Scanned with ACE Scanner

⇒ It is calculated by the formula:

$$P(SFQ) = P[S_3, S_3, S_3, S_3, S_1]$$

$$\Rightarrow P[S_3] * P[S_3/S_3] * P[S_3/S_3] * P[S_3/S_3] * P[S_1/S_3]$$

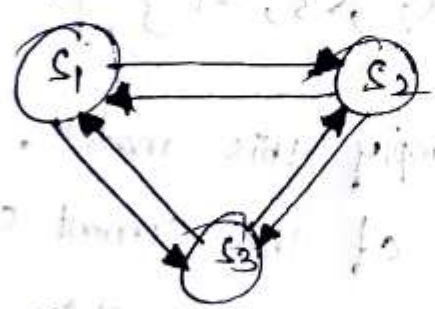
$$\Rightarrow S_3(init) * a_{3,3} * a_{3,3} * a_{3,3} * a_{1,3}$$

$$\Rightarrow (1.0) * (0.5) * (0.5) * (0.5) * (0.4)$$

$$\Rightarrow 0.05$$

⇒ In this equation we also assume the probability of the initial state S_3 is $\boxed{S_3(init) = 1.0}$.

The following "graph" depicts the Model.



From Graph:

* The Directed lines indicates the state transition Probabilities & $a_{i,j}$.
* These also an "implicit loop from every state back to itself.
* In Example every State Corresponded to Observable event. (i.e Change in the Market).

Formal Definition of HMM:-

A More formal definition of a discrete Hidden Markov Model is summarized by Mittendorf and Schauble (Mittendorf-94): as consisting of the following.

1. $S = \{S_0, \ldots, S_{n-1}\}$ as a "finite set of states" Where "S_0" is always indicates the "initial state". Typically these states are "Inter connected" such that any state can be reached from any other state.

2. $V = \{v_0, \ldots, v_{m-1}\}$ is a "finite set of output symbols".

3. $A = S \times S$ is a "transition Probability Matrix" Where a_{ij} represents the probability of transitioning from "State 'i' to State 'j'" such that $\sum_{j=0}^{n-1} a_{i,j} = 1$ for all $(\forall)\ i = 0, \ldots, n-1$.

* Every value in the Matrix is a positive value between 0 and 1.

* for the case;

Where every state can be reached from every other state, every value in the matrix will be non-zero.

Scanned with ACE Scanner

4. $B = S \times V$ is an "Output Probability Matrix",

Where element $b_{j,k}$ is a function determining the probability

and $\sum_{k=0}^{m-1} b_{j,k} = 1$ for $\forall$ (for all) $j = 0, \dots n-1$.

5. The Initial state distribution.

From the Given HMM Definition;

it can be used for both "sequence of output" and their "Probabilities".

The Complete specification of HMM requires :-

* Specification of states,
* Output Symbols &
* Three Probability Measures for
  - ① → The state transitions,
  - ② → Output probability functions
  - ③ → The initial states.

The distributions are frequently called A, B and π.

The following notation is used to define the model:

$$\lambda = (A, B, \pi)$$

Stemming Algorithm :—

→ Stemming means cutting (or) trimming the unnecessary data.

→ Stemming is a technique used for base/former abstract the base form of "affixes".

Affixes indicates the ending terms.

→ Stemming Programs are commonly referred as Stemming Algorithms and Stemmers.

The "main goal" is to "improve the performance of the system".

"Stemming is just like cutting the branches of trees".

Examples :—

1. eating, eaten, eats → Eat

2. Calculations, Calculate, Calculating → Calculate

3. Computerise, Computing, computers, Computerizing → Compute

Scanned with ACE Scanner

Subtopics of stemming algorithms are:

i) Porter stemming Algorithm
ii) Dictionary look-up stemmers
iii) Successor stemmers.
iv) conclusions.

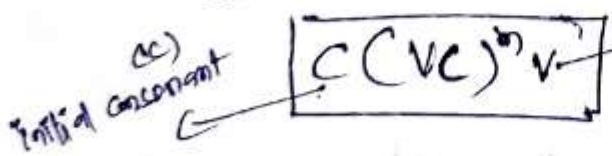
i) Porter stemming Algorithm:-

* It is introduced in 1980 to perform "actions" based on "conditions".
* It is a rule based "Algorithm", based on conditions.
* The stemming/porter algorithm based on set of stem, suffix, Prefix and associate conditions contains the

Conditions:-

(1) The measure "m" of a stem is a function of sequence of vowels (a, e, i, o, u) followed by a consonant.

* If "V" is a sequence of "vowels" and
* "C" is a sequence of "consonants", then "m" is:



Where:
Initial "C" and final "V" are Optional and "m" is the number of "VC" repeats.

Scanned with ACE Scanner

Examples :—

① FREE, Why

Find the given Word no. of consonants and no. of Vowels :—

Consonants ← C C V V → Vowels

F R E E

Why
↓↓↓
C C C

m=0

Here m=0 Because there is no "VC" combination.

② FREES, Whose

F R E E S

C C V C V

Whose
↓↓↓↓↓
C C V C V

① ①

Here m=1 Because there is 1 "VC" combination.

③ Prologue, Compute, Private

For above Words find no. of Vowels & Consonants and later find the no. of (VC) combination.

Prologue
↓↓↓↓↓↓↓
C C V C V C V

① ②

Compute
↓↓↓↓↓↓↓
C V C C V C V

① ①

Private
↓↓↓↓↓↓↓
C C V C V C V

① ②

Scanned with ACE Scanner

② *<X> :

This statement is indicating "stem ending with letter X."

Example :-

Index , tax , xerox , prefix , suffix etc.

③ *V* :

Stem contains a vowel.

Example :

Pen → vowel.
↓↓↓
CVC

, Pin ... etc

④ *d :-

Stem ending with "double consonant".

Example :

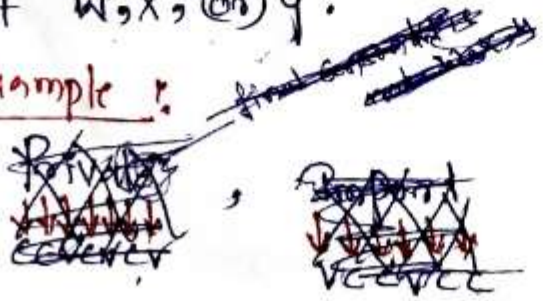
sky , dory , strong
↓↓↓ ↓↓↓ ↓↓↓↓↓
CCC CCC CCCVCC

↳ double consonants ↲

⑤ *O :-

It indicates the stem ends with "consonant-vowel-consonant sequence (C-V-C)" where final consonant is not W, X, (or) Y.

Example :



(*) Pen
↓↓↓
CVC ⇒ C-V-C sequence
last consonant is is not w, x, y.

Scanned with ACE Scanner

Rules :

The Rules are divided into steps to define the Order of applying the rules.

The following are some examples of rules:

PORTER STEMMER STEPS

Step (1) ⇒ (A), (B), (C)

(A)

i) SSES → SS Eg: Processes → Process, Stresses → Stress

ii) IES → I Eg: ties → ti

iii) SS → SS Eg: class → class

iv) S → ϵ (null) Eg: Cats → ~~cate~~ ⇒ i.e Cat

(B)

i) (m>1) EED → EE

If the stem is ending with "EED" is replaced by "EE".

Example :—

* Agreed

→ for the above word find the no. of (VC) combination (V → vowel, C → consonant) so that we get "m" value. If the "m" value is (>) greater than 1 then it satisfies the above condition.

Scanned with ACE Scanner

i.e Agreed

V C C V V C

① ②

$m = 2$. for above word 'm' value is '2'.

So that it satisfies the condition.

$m = 2$

(m>1) EED → EE

→ (2>1) Agreed → Agree

ii) *V* ED → €

If the stem containg Vowel, followed by "ED" is replaced by "€" (empty).

Example: Died → Di

Died → Di€ ⇒ Di (€ = null)

iii) (*V*) ING → €

If the stem Ending with "ING" is replaced by "€".

Example: Dancing → Danc ("null")

* Dancing → Danc€ ⇒ Danc

Working → work

Walking → walk

Making → Mak

* Making → Mak€ ⇒ Mak ("null")

Searching → search.

i) AT → ATE

Ex: Conflated

Conflated → Conflat (null) [∵ from 1(b) condition *V* ED → ∈]

i.e Conflated → conflat → conflate [∵ from 1(c) condition AT → ATE]

ii) BL → BLE

Example : 1 : Troubling → another condition in (1c)

For Word Troubling → (1b) condition form.

i.e. Troubling → Troubl [∵ (1b) condition *V*ing → ∈]

Remaining Word: Troubl

Troubl → Trouble [∵ BL → BLE (1)c condition]

Example : 2 "Bubbled"

→ Bubbled find "BL" in word i.e Bubbled

Bubbled → Bubble [∵ *V*ED → ∈]

Bubbled → Bubbl → Bubble [∵ BL → BLE]

Scanned with ACE Scanner

iii) "(*d ! (* L (or) * S (or) * Z)) → single character." (131)

The stem is ending with "double consonant" and that consonant is "not L (or) S (or) Z"

Example :-

① Hopping

↓↓↓↓↓↓↓
CVCCVCC

Hopping → Hopp∈ [∵ *V*ing → ∈]
(null)

CVCC
Hopp → Hop (single consonant.)

→ single character

∴ Hopping → Hopp → Hop

② Panned

Panned → Pann [∵ *V*ed → ∈ ; ed → ∈]
↳ ∈

Ramaning stem: Pann
CVCC
Pann → pan

- Ending with double consonant is replaced by single character.

Pann → Pan

∴ Panned → Pann → Pan

Scanned with ACE Scanner

③ swimming

→ swimming ⟶ swimm [∵ * V* ing ⟶ ϵ]

Remaining stem "swimm" :

swimm ⟶ swim

c c v c c

↳ double consonants replaced by single consonant

∴ swimming ⟶ swimm ⟶ swim

④ Falling

falling ⟶ fall [∵ *v*ing ⟶ ϵ]

Remaining stem : fall :

Fall ⟶ Fal (condition not Verified)

∴ Falling ⟶ Fall ⟶ Fal*

iv) (m = 1 & *o) ⟶ E

→ stem ending with c-v-c sequence and final consonant is not w, x, y.

The measure "m" (vc combination) is 1 and the stem is ending with cvc (CVC) combination is replaced by "E".

Scanned with ACE Scanner

Example : (initial) m=0

CVCVCC

1) Filing ⟶ Fil [∵ *v* ing ⟶ ε

Fil ⟶ File [cvc sequence followed by E.

cvc

(vc) m=1

∴ vc combination m=1, cvc followed by E)

| Filing ⟶ Fil ⟶ File | Condition Verified. |

2) Fail ⟶ fail (∵ not Verified)

cvvc (vc combinations m=1)

m=1 but cvc sequence is not there

so condition failed.

V) (*V*)Y ⟶ I

Our aim is to take out "Y" from the Word and the remaining stem containing atleast "one Vowel". If the Condition is satisfied "Y" is replaced with "I".

Example : | Happy ⟶ Happi |

Step(2) :- Derivation Morphology - I

i) (m>0) ATIONAL → ATE

Example :

Relational → Relate

∵ Relational
↓↓↓↓↓↓↓↓↓
CVCVCVVCVC
① ② ③ ④
∴ m=4
m>0 i.e 4>0]

ii) (m>0) IZATION → IZE

Example :-

Generalization → Generalize

Generalization
↓↓↓↓↓↓↓↓↓↓↓↓
CVCVCVCVCVCVC
① ② ③ ④ ⑤ ⑥

∴ m=6 ⇒ 6>0 & IZATION → IZE

iii) (m>0) BILITI → BLE

Example : Sensibiliti → Sensible

↓↓↓↓↓↓↓↓↓
CVCCVCVCVCV
1 2 3 4

∵ ve combinations = 4
m=4

∴ m=4 i.e 4>0 BILITI → BLE

Scanned with ACE Scanner

Step ③ :- Derivation morphology - II

i) (m>0) ICATE → IC

Ex 3: duplicate

Example 1: Triplicate → Triplic

CCVCVCVCV

① ② ③

m=3 ← ICATE → IC

i.e (3>0) ICATE → IC

ii) (m>0) FUL → ϵ

Example : HOPEFUL → HOPE

HOPEFUL

CVCVCVC

① ② ③ m=3

∴ 3>0 FUL → ϵ

i.e Hopeful → Hope (FUL = ϵ)

iii) (m>0) NESS → ϵ

Example : Goodness → good

CVCCVCC

m=2 — VC combinations 2

2>0 Ness → ϵ ⇒ Goodness → Good (ness = ϵ)

Scanned with ACE Scanner

Step 4 Derivation Morphology - III

i) (m>0) ANCE → ϵ

Example : Allowance → Allow
ϵ

Allowance
↓↓↓↓↓↓↓↓↓
VCCVCVCCV
① ② ③

m=3

∴ (3>0) Allowance → Allow.

ii) (m>0) ENT → ϵ

Example : Dependent → Depend
↓↓↓↓↓↓↓↓↓
CVCVCCVCC
① ② ③

m=3

∴ (3>0) Dependent → Depend
ϵ

iii) (m>0) IVE → ϵ

Example : Elective → elect

Elective
↓↓↓↓↓↓↓↓
VCVCCVCV
① ② ③

m=3

∴ (3>0) Elective → Elect
ϵ

Scanned with ACE Scanner

Step (5) A & B :-

(A)

i) $(m>1)$ E $\longrightarrow \epsilon$ Bye → By

Example : Probate → Probat

ii) $(m>1 \& !*O)$ NESS $\longrightarrow \epsilon$

condition $m>1$ and stem not ending with CVC combination.

Example : Goodness → Good

C V V C C V C C

1 2

not in CVC combination.

M=2 Goodness → Good

(VCC)

$\Downarrow \epsilon$

(B) $((m>1) \& *d \& *L)) \longrightarrow$ Single letter

Example : controll → control → Replaced by single letter 'L'

C V C C C V C C

① ②

check the condition m (i.e m=2) and the stem is ending with double consonant and 'l' is replaced by single letter.

Example 2 : Roll → Roll

m=1, Condition not verified.

Scanned with ACE Scanner

The two other techniques in stemming are

* ii) Dictionary lookup stemmers
iii) Successor stemmers
iv) Conclusions

ii) Dictionary lookup stemmers:

An alternative way to determine a "stem" is a "Dictionary lookup mechanism".

In this Approch, simple stemming rules are may be applied.

The Original term (or) stemmed version of the term is looked up in a dictionary and replaced by the stem that is the best representation of it.

This technique has been implemented in the "INQUERY and Retrieval ware system"

The INQUERY system uses a stemming technique called "K-stem".

K-stem: K-stem is a morphological Analyzer that conflates word variants to a root form.

Eg:- "Memorial" and "Memorize" reduced to "memory". But memorial and "memorize" are not synonyms and they have different meanings.

Scanned with ACE Scanner

K-Stem requires a word to be in the dictionary before it reduces one word form to another.

Some endings are removed, even if the root form is not found in the dictionary.

Ex: "ness", "ly"

* K-stem uses the following "six major" data files to control and limit the stemming process:

① Dictionary of Words (lexicon).

② "Supplemental list" of words for the dictionary.

③ "Exception list" for those words that should retain an "e" at the end (Ex: "suites" → "suite" but "suited" → "suit")

④ "Direct - conflation": → allows definition of direct conflation via word pairs that override the stemming Algorithm.

⑤ "Country - Nationality":

It indicates conflations between nationalities and countries". ("British" maps "Britain").

⑥ "Proper Nouns":

A list of Proper Nouns that should not be stemmed.

RetrievalWare System :-

This system lies in its thesauruses / semantic network support the data structure that contains over 4,00,000 words.

Scanned with ACE Scanner

iii) Successor stemmer :-

Successor stemmers are based on the "length of prefixes" that optimally expand the stem. It is also known as "successor variety".

It is used to determine the "word", word is divided into "segments".

for Example :-

* The successor variety for the first three letters (i.e word segment) of a five letter word is no. of words that have the same first three letters but fourth letter is different.

Graphical Representation :-

→ A Graphical representation of successor variety is shown in fig. Symbol tree.

→ The "symbol tree" for the terms

bag, barn, bring, both, box and bottle.

→ The successor variety for any prefix of word is the no. of childrens that are associated with the node in the "symbol tree". representing that prefix.

for example :- The successor variety for the first letter "b" is "three".

The successor variety for the prefix "ba" is "two".

Scanned with ACE Scanner

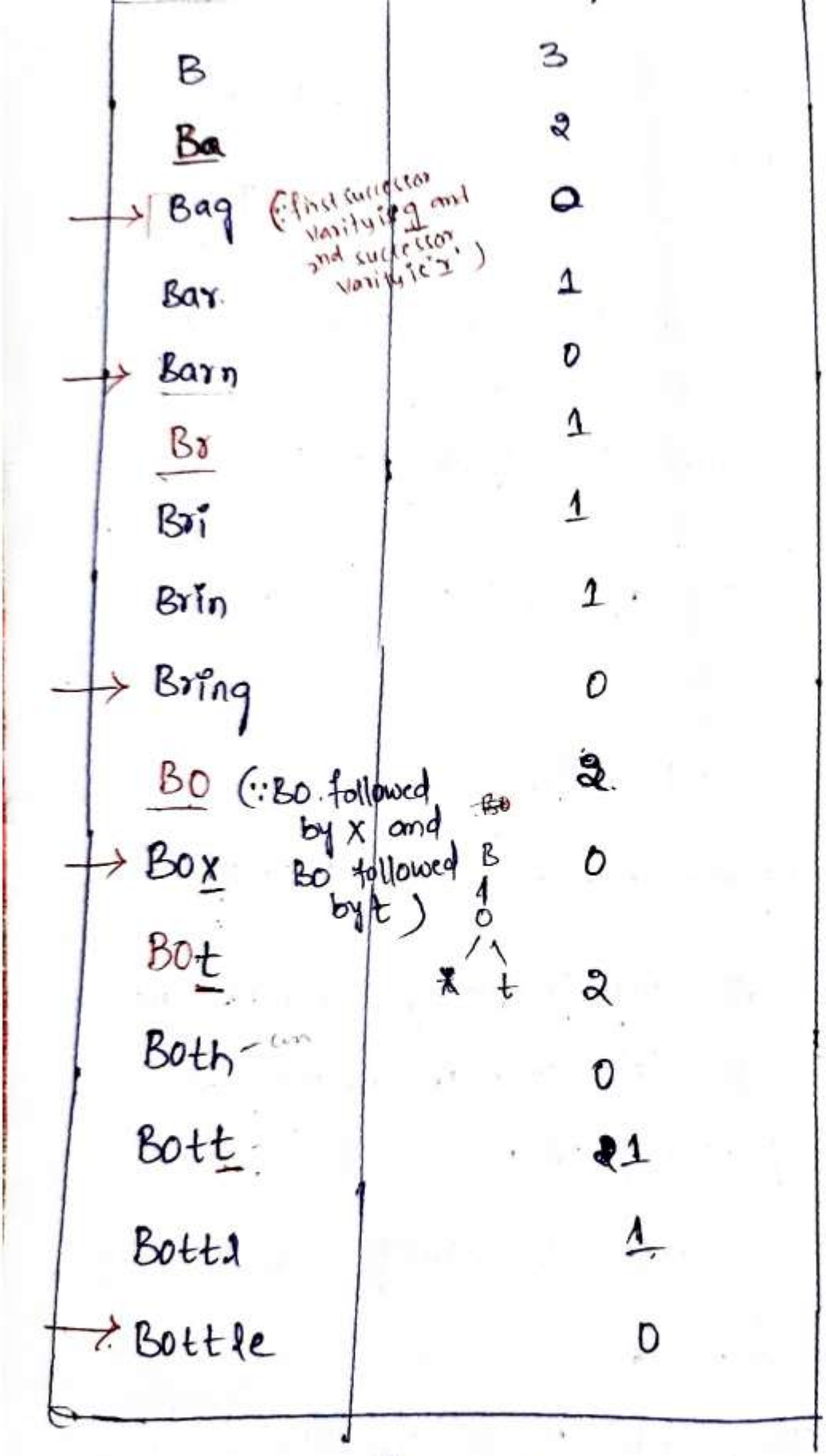


Fig: Successor Vari

Scanned with ACE Scanner

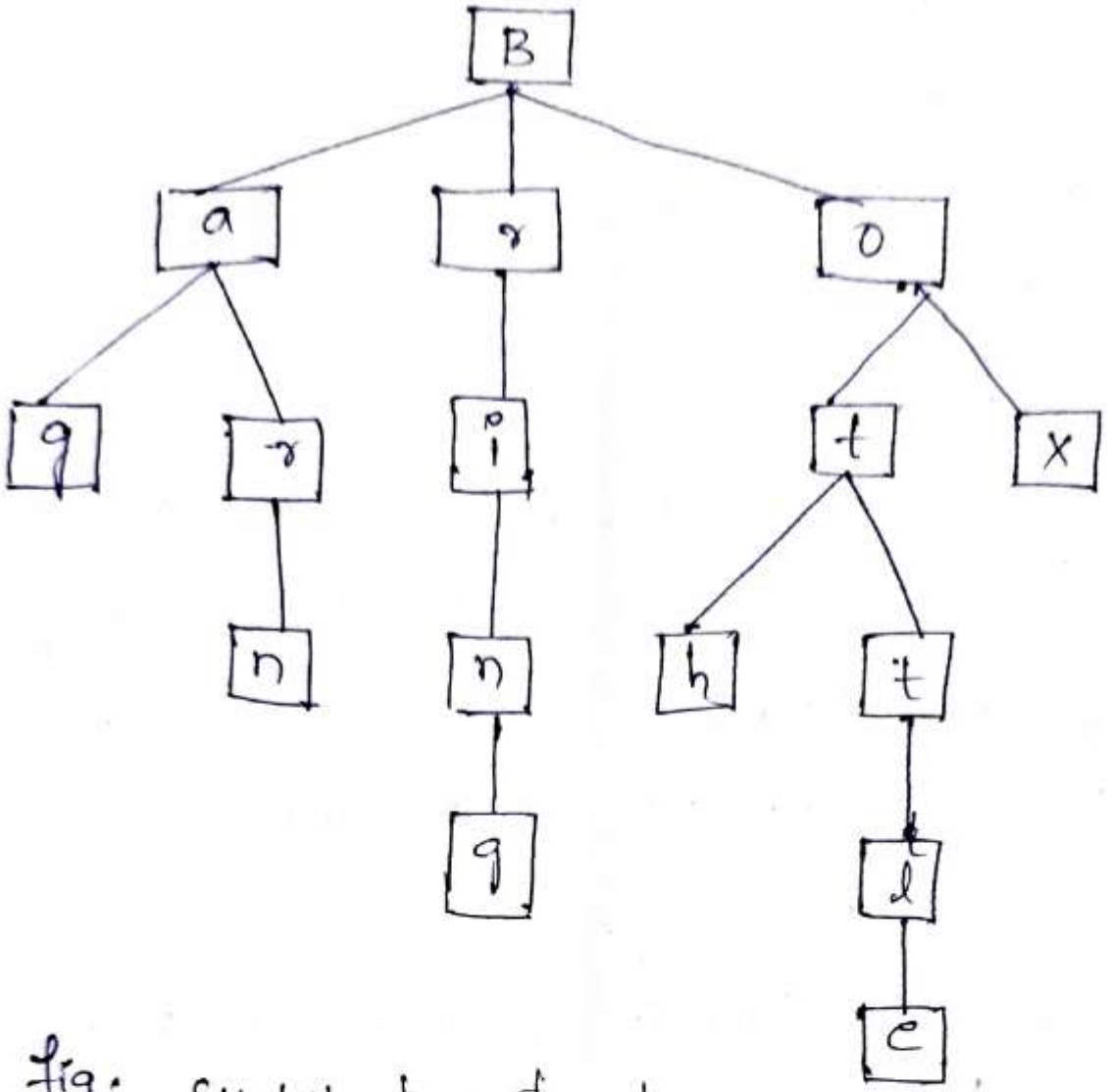


fig: Symbol tree for terms bag, barn, boring, box, bottle and both.

Successor Variety methods :-

The successor varieties of a words are used to segment a word by applying ~~of~~ one of the following 4- methods:

1. cutoff Method
2. Peak and plateau
3. Complete Word Method
4. Entropy Method.

Scanned with ACE Scanner

① cutoff Method :-

cut off value is selected to define "stem length" and the values are varies from word-to word / each possible set of words.

② Peak and plateau :-

"a segment break" is made after a character whose successor variety exceeds that character and immediately preceding it.

③ Complete Word Method :-

Break on boundaries of complete words.

④ Entropy Method :-

Used to Distribution of Successor Variety letters.

(Let $|D_{ak}|$ be no-of words begin with "k" length. Sequence of letter "a".)

iv) Conclusions

frakes summarized the study of various stemming. frakes come to following conclusions :

i) Stemming can effect retrieval (recall) and where effects were identified they were positive.

Scanned with ACE Scanner

ii) Stemming is as effective as manual conflation.

iii) Stemming is dependent upon the nature of the Vocabulary.

Extra Information:

⇒ To quantify the impact of stemmers, "paice" has defined a stemming performance measured, called "ERROR RATE RELATIVE TO TRUNCATION (ERRT).

ERRT :- ERRT can be used to compare stemming algorithms.

→ This approach depends upon 'concept groups'.

After applying a stemmer that is not perfect, "concept groups may still contain: multiple stems rather than one."

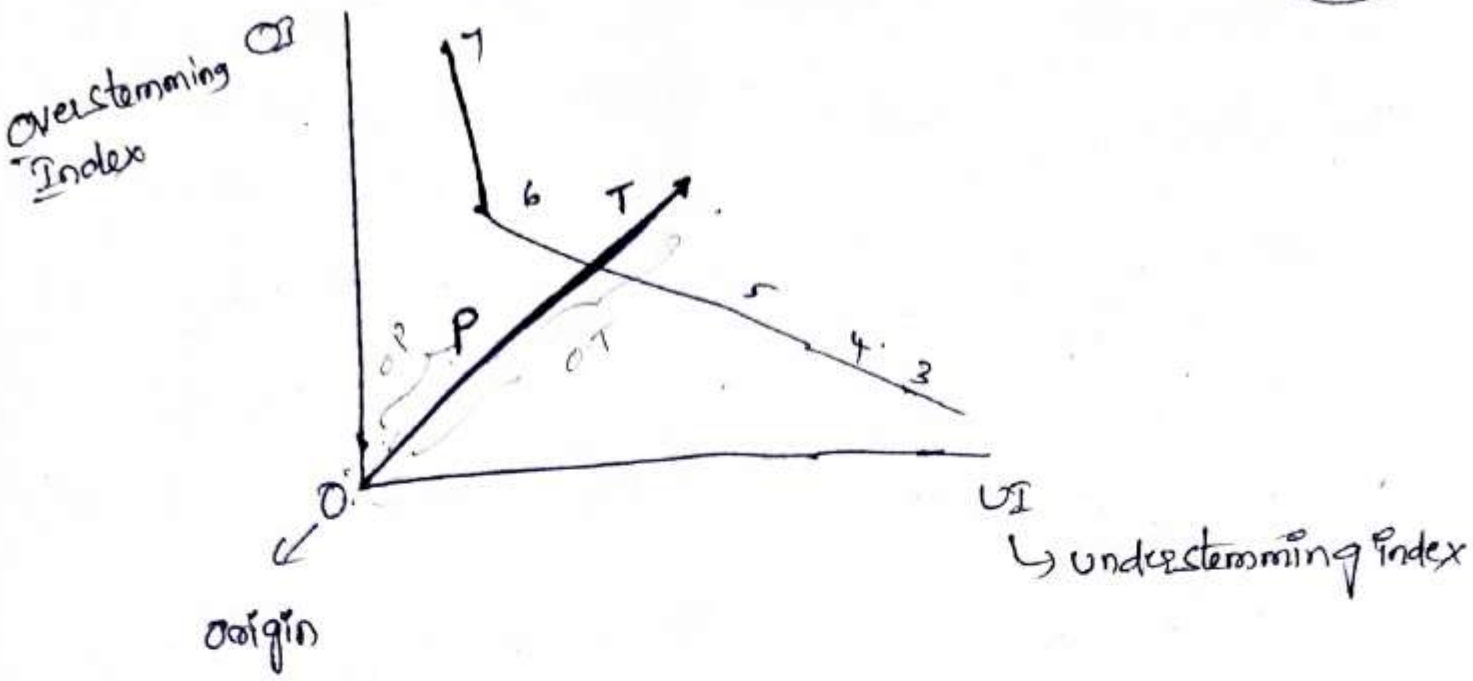
→ This introduces an error reflected in the Understemming Index (UI).

This also possible that same stem is found in multiple groups, is error state reflected in the Overstemming Index (OI).

⇒ The UI & OI values can be reflected calculated based on truncated word lengths.

The Perfect case is where UI & OI = 0(zero).

Scanned with ACE Scanner



⇒ ERRT is calculated as the distance from the Origin to the (UI, OI) coordinate of the stemmer being evaluated (OP) Versus distance from origin to the Pure truncation (OT).

Scanned with ACE Scanner