2) What is Stemming? Explain Porter Stemming Algorithm.

Stemming:

→ It is a simple rule based approach to remove suffixes.
→ It reduces words to their base form.
→ It eliminates suffixes from words
→ It is used to improve recall.
→ Also used to reduce index size.
→ It is used to enhance query matching.

Porter Stemming

1. measure(m) : counts Vowel-Consonent (VC) Sequences
2. <X> : Stem ends with X
3. v : Stem contains vowel
4. d : Stem ends in double consonent.
5. o : Stem ends in consonent - vowel sequence (excluding w, x, y)

measure(m).

Ex: ① F O R M A L I T I E S

C V C C V C V C V C

CV CC VC VC VV C

⇒ m=2

② I n f l a t e

V C C C V C C

VC CC VC V

⇒ m=2

Step	condition	Suffix	Replacement	Example
1a.	NULL	sses	ss	Stresses → Stress
1b.	*v*	ing	NULL	making → mak
1b1.	NULL	ate	at	Inflate → Inflat
1c.	*v*	y	I	happy → happi
2.	m>0	alti	al	Formalities → Formal
3.	m>0	Icate	IC	Duplicate → Duplic
4.	m>1	Able	NULL	Adjustable → Adjust
5.	m>1	e	NULL	Inflate → Inflat

5) Explain about Inverted File Structure with example

Inverted File Structure

→ It is a data structure, it allows efficient, full-text searches in the data base

→ It stores a mapping of words to their locations in the database table or document.

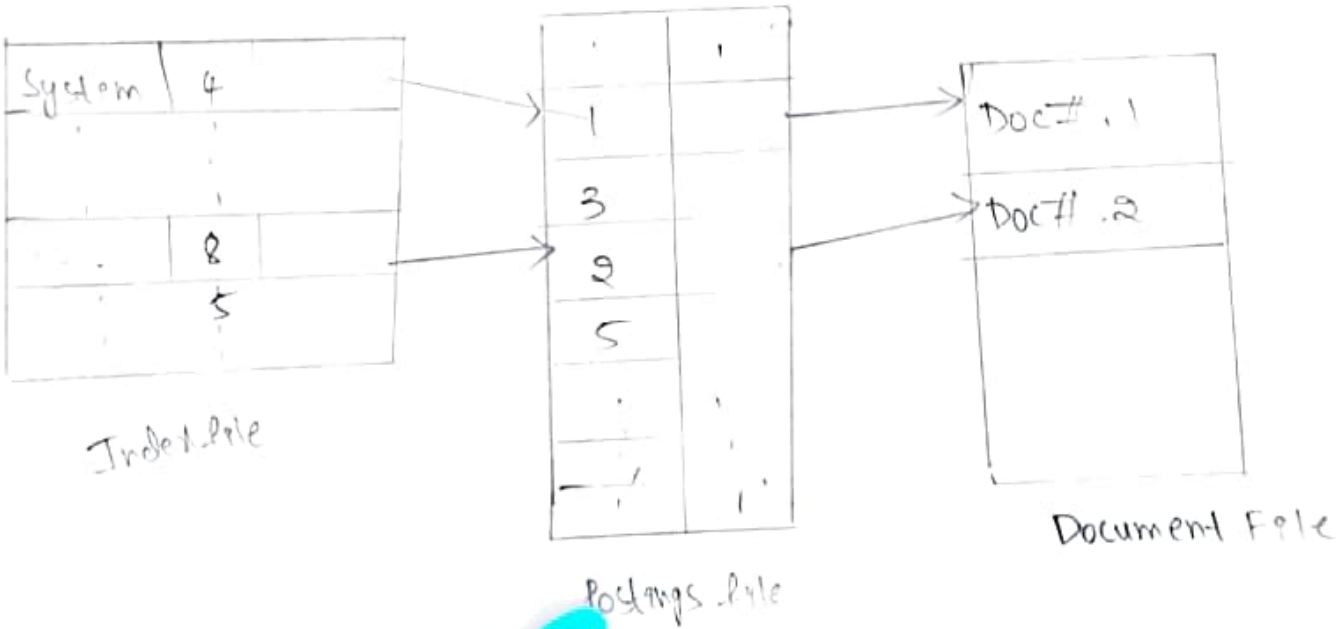
→ Inverted file based on methodology of storing an inversion of documents

→ For each word a list of documents in which the word is found is stored.

→ Each document is given a unique th numeric identifier that is stored in inversion list.

→ It is a data structure & used in IRS to organize data and allow for efficient full-text searches.

→ Each term is associated with a list of document identifiers.



Inverted File Structure

→ An inverted file structure is a data organization method.
→ This data organization method used in IRS
→ Also known as an inverted index

Key components

1. Index Terms: Unique words or phrases
2. Posting lists: Documents containing each term.

Benefits

→ Fast querying
→ Reduced storage
→ Improved scalability

Types

→ Simple Inverted Index
→ Compressed inverted index
→ Levelled inverted index

Applications

→ Search Engines
→ Digital libraries
→ Database Systems.

Example:

Document collection

Doc ID	Text
1	"The quick brown fox jumps"
2	"Brown cats are sleepy"
3	"Foxes are quick animals"

Inverted File Structure:

Term	Posting list
quick	1, 3
brown	1, 2, 3
fox	1, 3
jumps	1
cats	2
sleepy	2
animals	3

Query: "quick brown fox"

Result: Doc 1, Doc 3

Cluster 2 : T_3, T_4, T_9

Finally, T_5 is selected as a new seed because it is ... viz., cluster3.

Unit-2

* N-gram Data Structure

→ N-gram is one of the data structure considered as a special technique for Stemming.

→ It is not concerned about the semantics (meaning) of the word.

→ ~~Instead, they depend upon the fixed consecutive series of characters.~~ n

→ N-gram data structure can be viewed as Unique.

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

Specializations:

- Special Data structure
- Ignore words
- Ignore sentences
- N-gram = N-Length.
- Input as continuous data.
- Logical Linkages.

- Special Data Structure:

→ It is one of the special data ~~store~~ structure.

→ It is unique.

- Ignore words:

→ It ignores words, repeating a word once or twice.

N-gram:

→ Indicating as N-gram is equal to N-Length.

For n=1, 1-gram (Uni-gram)

For n=2, 2-gram (Bi-gram)

For n=3, 3-gram (Tri-gram)

⋮

n-gram

Example: Hello How are you today

1-gram → "Hello" "How" "are" "you" "today"

2-gram → "Hello How" "are you" "you today"

3-gram → "Hello How are" ~~"How are you"~~

"are you today"

→ 1-gram (no word history)

→ 2-gram (one word history)

→ 3-gram (two word history)

$$Predaction(w_q / w_{q-1}) = \frac{count(w_{q-1}, w_q)}{count(w_{q-1})}$$

w_{q-1} = do

w_q = Am

Ex: Do

I am Amit.

I like computer

Do Amit like computer

Amit I am

Do I like Amit

Do I like computer

I do like Amit

Next word	Frequency	Probability
I	6	2/4
Am	2	0/4
Amit	5	1/4
Like	5	1/4
Computer	3	0/4
do	4	0/4

$$= \frac{count(do, I)}{count(do)} = \frac{2}{4} = \frac{do, Am}{do}$$

$$P = \left(w_q / w_{q-2}, w_{q-1}\right) = \frac{\text{count}(w_{q-2}, w_{q-1})}{\text{count}(w_{q-2}, w_{q-1})}$$

$P \subseteq (w$

* PAT Data Structure

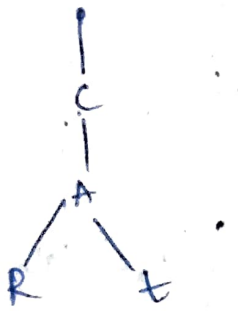
→ the name PAT comes from PATRICIA trees.
- which are used to search text.

→ PAT is one of the data structure.

→ It is also known as PAT tree or PAT array.

→ Input data is transformed into searchable data structure

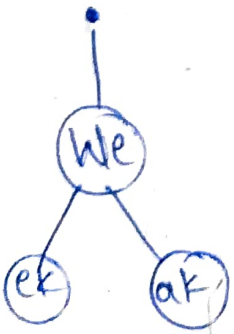
Ex: ① Cat
Car



Regular PAT Tree

Compact PAT Tree

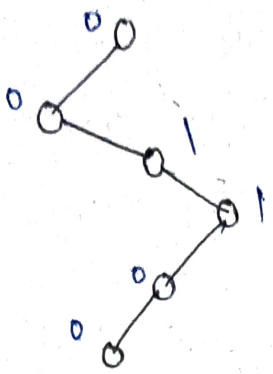
② Week, Weak



Binary representation

ex: 001100

* 0 - left side
* 1 - right side



→ PAT trees are created from the given input string.

→ PAT trees supports strings and binary numbers.

---

* Signature File Structure

→ The main aim of signature file structure is to perform a fast test

→ So that the unnecessary items that are not related to a query are removed.

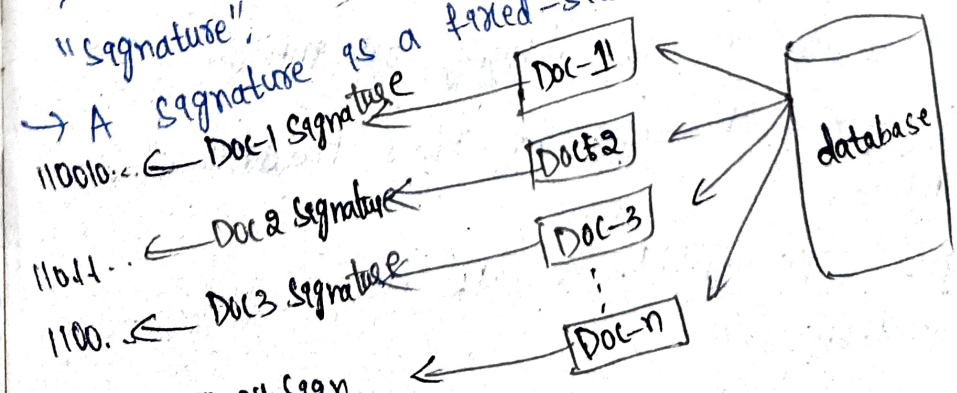
→ File structure is highly compresses & unordered.

→ It requires significantly less space than an Inverted file structure.

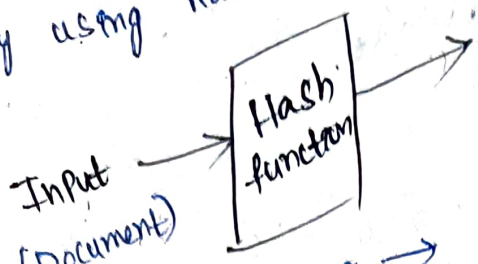
→ It is a method used for indexing & searching text documents

→ Each document in the database is represented by a "signature".

→ A signature is a fixed-size bit string.



→ By using hash



Ex: this as a string → 0001000

Ex: Computer

User → Query → Computed 11100001 10 → Doc9 → 1010100110

AND 1110000110
1010100110
1010000100 → match found

→ If we get all zero, then match not found.

* Hypertext & XML Data structure

→ Hypertext as a data structure.

→ Basically, Internet as a global information network which introduced a new storage data structure called Hypertext.

→ this Hypertext data structure display information on a WEB Page.

→ this d. structure is used largely on internet.

→ It is different from the other traditional data structures.

→ Languages like HTML & XML are used to store hypertext.

HTML – Hyper Text Markup Language
XML – Extendable Markup Language

1. HTML

→ HTML is a standard markup language.
→ It is used for creating web pages.

Key features

→ Tags (<P>, <img>)
→ Attributes (href, src)
→ Hyperlinks

Ex:
```
<html>
  <head>
    <title> Example Page </title>
  </head>
  <body>
    <P> Welcome all </P>
  </body>
</html>
```

2. XML

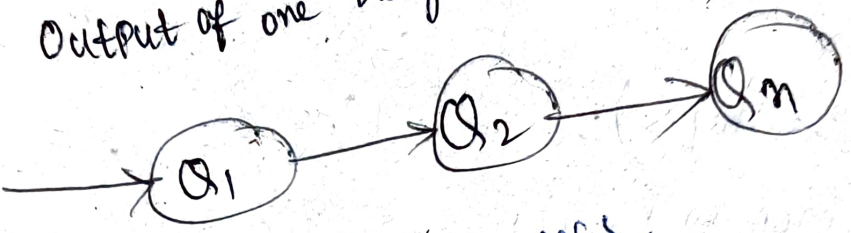
→ XML stands for "Extendable Markup Language"
→ It is a flexible markup language.
→ Used to store & exchange data.

Ex:
```
<Person>
  <name> abc </name>
  <address>
    <street> .... </street>
    <city> .... </city>
    <state> .... </state>
  </address>
</person>
```

* Hidden Markov model (HMM)

→ It is used for searching as Textual queries.
→ the output of one query as supplied / given to ~~the~~ another query as input

Output of one query = Input of another query

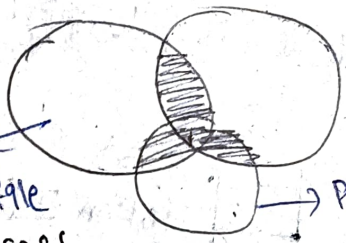


→ It is a chain process.
→ Q_1 takes input & produce output.
→ Output of Q_1 is given as input Q_2
→ Q_2 produce some ouput
→ It is repeated until end of the process.
→ Development for HMM approach
- begins with by applying Bayes Rule to Conditional Probability.

$$P(A/B) = \frac{P(B/A)\,P(A)}{P(B)}$$

* Indexing

→ It is a process of organizing & structuring data.
→ It is the oldest technique for finding the items.
→ Originally, indexing is called as "Cataloging".
→ the evolution of IRS have changed the objectives of indexing.
→ It is an important process in IRS.
→ Indexing process can be manual or automatic.



Indexing process