UNIT - IV

Information Retrieval using NLP

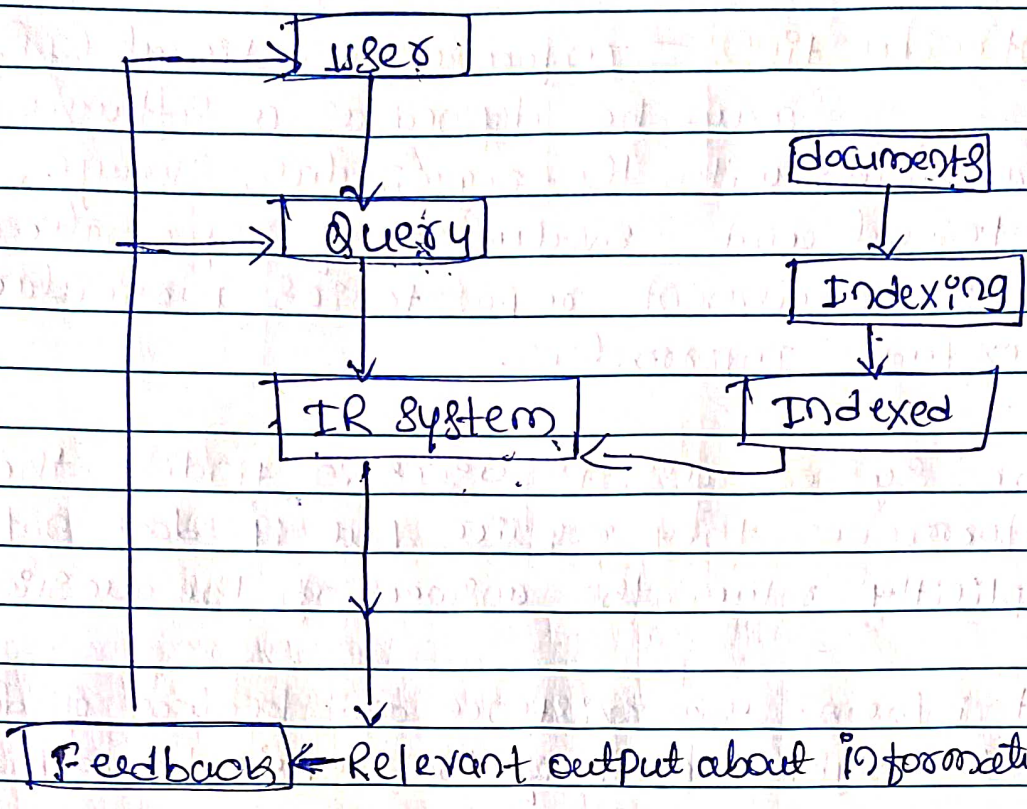
Introduction:- Information retrieval (IR) may be defined as a software program that deals with the organization, storage, retrieval and evaluation of information from document repositories particularly textual Information.

- The System assists users in finding the information they require but it does not explicitly return the answers of the questions.
- It informs the existence and location of documents that might consist of the required information.
- The documents that satisfy user's requirement are called relevant documents.
- A perfect IR system will retrieve only relevant documents.

With the help of the following diagram, we can under-stand the process of Information retrieval (IR).

Eg:- Information Retrieval can be when a user enters a query into the system.

* IR System searches a collection of natural language documents with the goal of retrieving exactly the set of documents that matches a user's question.



- It is clear from the above diagram that a user who needs information will have to formulate a request in the form of query in natural language.
- Then the IR system will respond by retrieving the relevant output, in the form of documents, about the required information.

Information Retrieval (IR) model:-

models are used in many scientific areas having objective to understand some phenomenon in the real world.

- A model of information retrieval predicts and explains what a user will find in relevance to the given query.

IR model is basically a pattern that defines the above mentioned aspects of retrieval procedure and consists of the following:

- A model for documents.
- A model for queries.
- A matching function that compares queries to documents.

Mathematically, a retrieval model consists of

D - Representation for documents.
R -> Representation for queries.
F -> The modeling framework for D, Q along with relationship between them.

R(q, dj) - A similarity function that orders the documents with respect to the query. It is also called as ranking.

Types of Information Retrieval (IR) model:-

An IR model can be classified into the following three models-

1) **Classical IR Model:-**

It is the simplest and easy to implement IR model.

- This model is based on mathematical knowledge that was easily recognized and understood as well.

Eg:- Boolean, vector and probabilistic are the three classical IR models.

* **Non-classical IR model:—**

* It is completely opposite to classical IR model.

Such kind of IR models are based on principles other than similarity, probability, Boolean operations.

Eg * Information logic model, situation theory model and interaction models are the examples of non-classical IR model.

* **Alternative IR model:—** (Improvement (increase))

It is the enhancement of classical IR model making use of some specific techniques from other fields.

o Cluster model, fuzzy model and latent semantic indexing (LSI) models are the example of alternative IR model.

* **Design features of IR systems:—**

⇒ **Inverted Index:—**

The primary data structure of most of the IR systems is in the form of inverted index.

It can define an inverted index as a data structure that list, for every word, all documents that contain it and frequency of the occurrences in document.

* **Stop word Elimination:-**

Stop words are those high frequency words that are deemed unlikely to be useful for searching. They have less semantic weights.

- All such kind of words are in list called stop list.

Eg: ~~like 'in', 'of', 'for', 'at'~~
like 'a', 'an', 'the' articles and prepositions like 'in', 'of', 'for', 'at' etc. are the examples of stop words.

- The size of the inverted index can be significantly reduced by stop list.
On other hand, sometimes the elimination of stop word may cause elimination of the term that is useful for searching.

Eg: If we eliminate the alphabet "A" from vitamin A" then it would have no significance.

Stemming:-

The simplified form of morphological analysis, is the heuristic process of extracting the base form of words by chopping off the ends of words.

Eg:
The words laughing, laughs, laughed would be stemmed to the root word Laugh.

* The Boolean model:-

It is the oldest information retrieval (IR) model.

The model is based on set theory and the boolean algebra, where, documents are sets of terms and queries are boolean expressions on terms.

The model

The boolean model can be defined as:

D -> A set of words, i.e the indexing terms present in a document. Here, each term is either present (1) or absent (0).

Q -> A boolean expression, where terms are the index terms and operators are logical products - AND, logical sum - OR and logical difference - NOT

F -> Boolean algebra over sets of terms as well as over sets of documents.

If we talk about the relevance feedback, then in Boolean IR model the Relevance prediction can be defined as follows.

R -> A document is predicted as relevant to the query expression if and only if it satisfies the query expression as -

(("text" information) ^ retrieval ^ ~ theory)

Advantages of Boolean model:

- the simplest model, which is based on sets.
- Easy to understand and implement.
- It only retrieves exact matches
- It gives the user a sense of control over the system.

Dis-Advantages of Boolean model:

- The model's similarity function is Boolean. Hence, there would be no partial matches. This can be annoying for the users.
- In this model, the boolean operator usage has much more influence than a critical word.
- the query language is expressive, but it is complicated too.
- No ranking for retrieved documents.

* **Vector Space Model :—**

Vector Space model (VSM) is a way of representing documents through the words that they contain.

- It is Standard technique in IR.
- The VSM allows decisions to be made about which documents are similar to each other & to keyword queries.
- Each document is broken down into a word frequency table.
- The tables are called vectors & can be stored as arrays.
- A vocabulary is built from all the words in all documents in the system.
- Each document is represented as vector based against the vocabulary.

Examples:-

Document A: A dog and a cat

a	dog	and	cat
2	1	1	1

- Document B: A frog

a	Frog
1	1

- The vocabulary contains all words used
  - a, dog, and, cat, Frog

- The vocabulary needs to be stored
  - a, and, cat, dog, frog.

Document A: A dog & a cat

a	and	cat	dog	frog
2	1	1	1	0

Vector : (2,1,1,1,0)

Document B: A Frog

a	and	cat	dog	frog
1	0	0	0	1

Vector (1,0,0,0,1)

- Queries can be represented as vectors in the same way as documents:
  Dog = (0,0,0,1,0)
  Frog = ( )
  Dog and frog = ( )

* **Similarity measures:**
  There are many different ways to measure how similar two documents are, or how similar a documents is to a query.
  - The cosine measure is a very common similarity measure.
  - Using a similarity measure, a set of documents can be compared to a query and most similar document returned.

* **The cosine measure:-**

For two vectors d & d' the cosine similarity between d & d' is given by

$$\frac{d \times d'}{|d|\,|d'|}$$

Here $d \times d'$ is the vector product of d & d' calculated by multiplying corresponding frequencies together.

The cosine measure calculates the angle between the vectors in a high-dimensional virtual space.

let,

$d = (2,1,1,1,0)$ & $d' = (0,0,0,1,0)$

$d \times d' = 2\times0 + 1\times0 + 1\times0 + 1\times1 + 0\times0 = 1$

$|d| = \sqrt{(2^2+1^2+1^2+1^2+0^2} = \sqrt{7} = 2.646$

$|d'| = \sqrt{(0^2+0^2+0^2+1^2+0^2)} = \sqrt{1} = 1$

$$\text{Similarity} = \frac{1}{(1\times2.646)} = 0.378$$

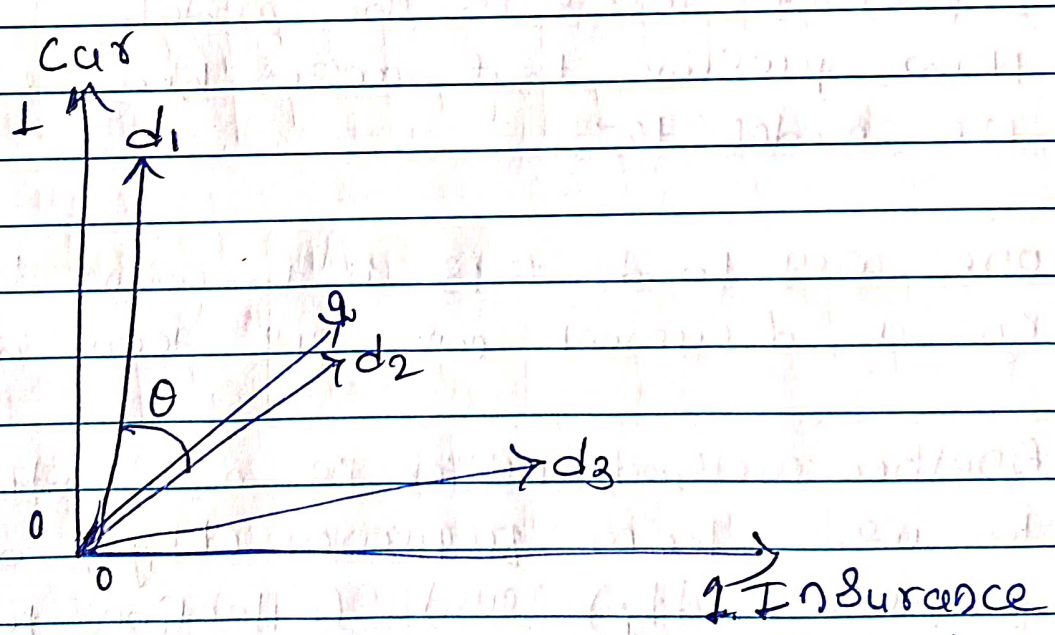
let $d = (1,0,0,0,1)$ & $d' = (0,0,0,1,0)$

$$\text{Similarity} = \frac{0}{(1.414\times1)} = 0$$

- **Ranking documents:-**
  - A user enters a query.
  - The query is compared to all documents using a similarity measures.
  - The user is shown the documents in decreasing order of similarity to the query term.

Vector Space Representation with Query and Documents:

The query and documents are represented by a two-dimensional vector space.
The terms are car & insurance. This is one query and three documents in the vector space.



The top ranked document in response to the terms car and insurance will be the document d_2

because the angle between q & d_2 is the smallest.

- The reason behind this is that both the concepts car and insurance are salient in d_2 and hence have the high weights.

- On the other side, d_1 & d_3 also mention both the terms but in each case, one of them is not a centrally important term in the document.

Term weighting:-

It means the weights on the terms in vector space.

- Higher the weight of the term, greater would be the impact of the term on cosine.
- More weights should be assigned to the more important terms in the model.

Now question that arises here is how can we model this.

One way to do this is to count the words in a document as its term weight.

Another method, which is more effective, is to use term frequency (tf_{ij}); document frequency (df_i) and collection frequency (cf_i).

Term Frequency (tf_{ij}):-

It may be defined as the number of occurrences of w_i in d_j. The information that is captured by term frequency is how salient a word is within the given document or in other words we can say that the higher the term frequency the more that word is a good description of the content of that document.

① Document Frequency (df_i)

It may be defined as the total number of documents in the collection in which w_i occurs.

It is an indicator of informativeness. Semantically focused words will occur several times in the document unlike the semantically unfocused words.

Collection Frequency (cf_i):-

It may be defined as the total number of occurrences of w_i in the collection.

Mathematically,

$$df_i \leq cf_i \text{ and } \sum_j tf_{ij} = cf_i$$

Term Frequency Factor:—

which means that if a term t appears often in a document than a query containing 't' should retrieve that document.

We can combine words term frequency (tf_{ij}) and document frequency (df_i) into a single weight as follows-

$$\text{Weight}(i,j) = \begin{cases} (1 + \log(tf_{ij}) \log \frac{N}{df_i} & \text{if } tf_{ij} \geq 1 \\ 0 & \text{if } tf_{ij} = 0 \end{cases}$$

Here N is the total number of documents.

Inverse document Frequency (idf):—

- This is another form of document frequency weighting and often called idf weighting or inverse document frequency weighting.

The important point of idf weighting is that the term's scarcity across the collection is a measure of its importance and importance is inversely proportional to frequency of occurrences

Mathematically:

$$idf_t = \log\left(1 + \frac{N}{n_t}\right)$$

$$idf_t = \log\left(\frac{N - n_t}{n_t}\right)$$

Here,

N = documents in the collection.

n_t = documents containing term t.

User Query Improvement:-

The primary goal of any information retrieval system must be accuracy - to produce relevant documents as per the user's requirement.

The o/p of any IR system is dependent on the user's query and a well-formated query will produce more accurate results.

User can improve query with the help of relevance feedback.

Relevance Feedback:-

It takes the output that is initially returned from the given query. This initial output can be used to gather user information and to know whether that output is relevant to perform new query or not.

The feedback classified as:

1) **Explicit feed back:** —
It may defined as the feedback that is obtained from the assessors of relevance.
These assessors will also indicate the relevance of a document retrieved from the query.
To improve query retrieval performance, the relevance feedback information needs to be interpolated with the original query.

2] **Implicit feed back:** —
It is the feedback that is inferred from user behavior.
The behavior includes a document, which document is selected for viewing and which is not, page browsing and scrolling actions etc..

Eg:- dwell time, which is a measure of how much time a user spends viewing the page linked to in a search result.

Pseudo feed back: —
It is also called Blind feedback.
It provides a method for automatic local analysis.

* The manual part of relevance feedback is automated with the help of pseudo relevance feedback so that the user gets improved retrieval performance without an extended interaction.

* The adv. of this system is that it does not require accessors like an explicit relevance feedback system.

* **Named Entity Recognition:- (NER)**

NER is one of the most popular data preprocessing tasks.

→ It involves the identification of key information in the text and classification into a set of predefined categories.

→ An entity is basically the thing that is consistently talked about or refer to in the text.

→ Named Entities (NEs) are proper names in texts ie the names of persons, organizations, locations, times & quantities.

* **NER System Building process:-**

- NER involves identification of proper names in texts and classification into a set of predefined categories of interest.

Three universally accepted categories:
-> Person
-> location
-> organisation
other common tasks:
* recognition of date / time expressions.
* measures (percent, money, weight, etc).
* Email address etc--

Other domain specific entities
* names of drugs, genes
* medical conditions
* names of ships
* Bibliographic references etc--

eg. John sold 5 companies in 2023.

<ENAMEX TYPE="PERSON"> John </ENAMEX>
Sold <NUMEX TYPE="QUANTITY"> 5 </NUMEX>
Companies in <TIMEX TYPE="DATE">
2023 </TIMEX>

Evaluating NER System

1) **precision**

Measures how precise / accurate your model is. It is the ratio b/w the correctly identified & all identified +ves.

precision = True positive / (True positive + false +ve)

2) **Recall:** —

Measures the model's ability to predict actual positive classes. It is the ratio between predicated true +ves & wat was actually tagged.

$$\text{Recall} = \frac{TP}{(TP + FN)}$$

3) **F1 score:** — F_1 score is a function of precision & Recall. It's needed when you seek a balance b/w precision & Recall.

F_1 score = 2 * precision * Recall / (precision + Recall)

* **Entity Extraction:** —

It is a text analysis technique that uses NLP to automatically pull out specific data from unstructured text and classifies it according to predefined categories.

These categories are named entities, the words or phrases that represent a noun. This includes proper names but also numerical expressions of time or quantity such as phone no.s, monetary value / dates.

* **Relation Extraction:** —

It is task of predicating attributes & relations for entities in a sentence.

Ex: Barack obama was born in Honolulu Hawaii
A relation classifier aims at predicting the relation of "bornInCity".

* **Reference Resolution:—**

Reference may be defined as linguistic expression to denote an entity or individual.

Eg: In the passage, Ram, the manager of ABC Bank, saw his friend Shyam at shop.
He went to meet him, the linguistic expressions like Ram, His, He are reference.

* **Conference Resolution:—**

CR is the task of finding all linguistic expressions in a given text that refer to the same-real world entity.

CR is using in a variety of NLP tasks such as:

-> Text understanding
- Document summarization
- Information extraction
- Sentiment analysis
- Machine translation

After finding & grouping these expressions we can resolve them by replacing, as pronouns with noun phrases.

Ex: "I gave my laptop to Andrew because he ~~told~~ needs it to do his assignment" Peter said.

In this sentence the main entities are
1) Andrew 2) peter 3) peter's laptop.

After that, it replaces all the pronouns with relevant nouns.

[Peter] gave [Peter's laptop] to [Andrew] because [Andrew] told [Peter] that [Andrew] needs [Peter's] [laptop] to do [Peter's] assignment" [Peter] said.

CR is using in a variety of NLP tasks such as:

- Text understanding
- Document summarization
- Information extraction
- Sentiment analysis
- Machine Translation.

From page 8 **Continue of Reference Resolution**

Reference Resolution may be defined as the task of determining what entities are referred to by which linguistic expression.

* **Terminologies:-**

1) **Referencing Expression:-**
   Expression that is used to perform reference passage.

2) **Referent:-**
   It is the entity that is referred

Eg: Ram is a referent

3) **Coreferi:—**
When two expressions are used to refer same entities
Eg: Ram & he are corefer.

4) **Antecedent:—**
The term has the license to use another term.
Eg: Ram is the antecendent of the reference he.

5) **Anaphora & Anaphoric:—**
It may be defined as the reference to an entity that has been previously introduced into the sentence. And the referring expression is called Anaphoric.

6) **Discourse model:—**
The model that contains the representations of the entities that have been referred to in the discourse & the relationship they are engaged in.

* **Reference Resolution Tasks:—**

1) ~~Confer~~ Coreference Resolution
2) Constraint on coreference resolution
3) pronominal Anaphora Resolution.

1) **Coreference Resolution:—**

It is the task of finding referring expressions in a text that refer to the same entity.

2) **Constraint on coreference resolution:-**

In English, the main problem for coreference resolution is the pronoun it.

- The reason behind this is that the pronoun it has many uses.

Eg:- It can refer much like he & she

- The pronoun it also refers to the things that do not refer to specific things.

Eg: It's raining. It is really good.

3) **Pronominal Anaphora Resolution:—**

It may be defined as the task of finding antecedent for a single pronoun.

Eg: The pronoun is his and the task of pronominal anaphora resolution is to find word Ram Because Ram is the antecendant.

Preprocessing in CR:—

- **Named Entity Recognition:—**
  which finds & classifies NE in a text into predefined categories such as locations, organizations or names of persons.
- **Entity Linking or Wikification:—**
  which aligns textual mentions of named entities to their corresponding entries in a knowledge base
- **Parts of Speech Tagging:-** A process of speech to word in a sentence.

Lemmatization:—
Do a morphological analysis of words with the aim of remove inflectional endings & return the base or dictionary form a word.

Stemming: cutting beginning/end of the word.

* **Cross Lingual Information Retrieval:—** (CLIR)

It is the task of retrieving relevant information when the document collection is written in a different language from the user query.

Translation Approaches:—
CLIR requires the ability to represent the match informally in the same representation space even if the query & the document in different languages.

Translation process can be in several ways:

1) **Document translation**: It is to map the document representation into query representation step.

2) **Query translation** is to map the query representation into the document representation space.

3) **Pivot language or Interlingua**: is to map both document & query representations to a third space.

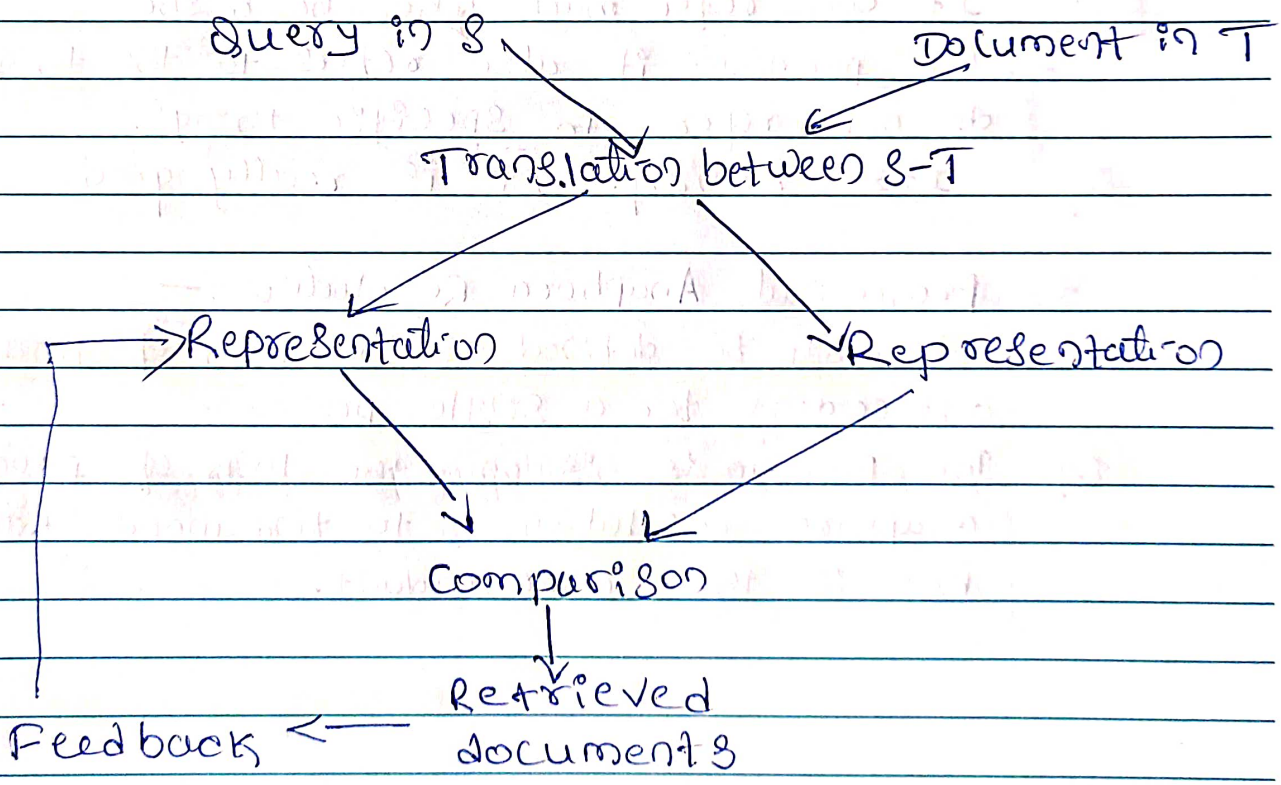


Fig: Typical architecture of CLIR System.

* **Factors affecting the performance of CLIR**
  - Limited size of Dictionary
  - Query translation/ transliteration performance.

Challenges in CLR:—

1) **Translation Ambiguity:**—
While translating from source language to target language, more than one translation may be possible
Eg: the word गला (maan, respect / neck) has two meanings neck & respect

2) **Phrase identification & translation:**—
Identifying Phrases in limited context & translating them as a whole entity rather than individual word translation is difficult.

3) **Translate / Transliterate a term:**—
There are ambiguous names which need to be transliterate instead of translation.

4) **Font:** Many documents on web are not in unicode format these documents need to be converted in unicode format for further processing & storage.

5) **Morphological analysis:**—
Different for different languages.

6) **Out-of-Vocabulary (OOV) problems:** - New words get added to language which may not be recognized by the system