* Information Retrieval => Activity of finding material (usually documents) of an unstructured nature (usually text) that satisfies an information need (query) from within large digital collections.

* Data => Raw, unprocessed facts and figures.

* Information => Data that has been processed into meaningful context

* Knowledge => Combination of information, experience and understanding that provides a context for decision-making.

* Wisdom => Application of knowledge and experience to make informed decisions, often in complex and uncertain situations.

Dated:

★DB V/S IR =>

A DBS is a software application that is designed to store and manage large amounts of structured data. It provides an organized and systematic way of storing data and allows for efficient retrieval through methods such as SQL queries.

IR is concerned with storage, retrieval and ranking of unstructured data such as documents, images and videos. It involves the development of algorithms and methods that allows users to quickly search and retrieve information from large collection of data.

★ Data Mining V/S Text Mining =>

Data Mining is the process of discovering hidden patterns, relationships and insights from large datasets. It involves the use of statistical techniques, machine learning algorithms and data visualization to extract meaningful information from large datasets. The goal of data mining is to find correlations in data that can help make informed decisions.

Text mining is a specialized area of data mining that is focused on extracting informations from large collection of unstructured text data. It involves the use of Natural Language Processing techniques (NLP), ML algorithms and IR methods to extract meaningful information from unstructured text data. The goal is to discover hidden relationships, sentiments, opinions and topics from large collections of text data.

Dated:

★ IR V/S Text Mining =>

IR is concerned with finding, organizing and retrieving information from large collections of data. IR systems use algorithms and methods such as keyword-base search, Boolean search and probabilistic retrieval to match user queries with relevant information from the collections. The goal of IR is to provide users with the most relevant information in response to their queries as efficiently and effectively as possible.

✦ TEXT MINING EXPLAINED EARLIER ✦

★ IR V/S Information Extraction =>

IE is the process of automatically extracting structured information from unstructured or semi structured data. IE involves use of NLP, ML and pattern recognition techniques to identify and extract relevant information such as entities, relationships and events. The goal is to transform the data into structured data that can be easily analyzed and utilized for decision making and other applications.

✦ IR EXPLAINED EARLIER ✦

★ Adhoc Retrieval Systems => IR systems designed to handle unstructured or semi-structured data. They are based on keyword-based search or probabilistic algorithms and do not have a predefined schema. They are commonly used in web search engines, intranet search, e-discovery, digital libraries. They are known for their ease of use and flexibility but can also be less precise than structured DBS.

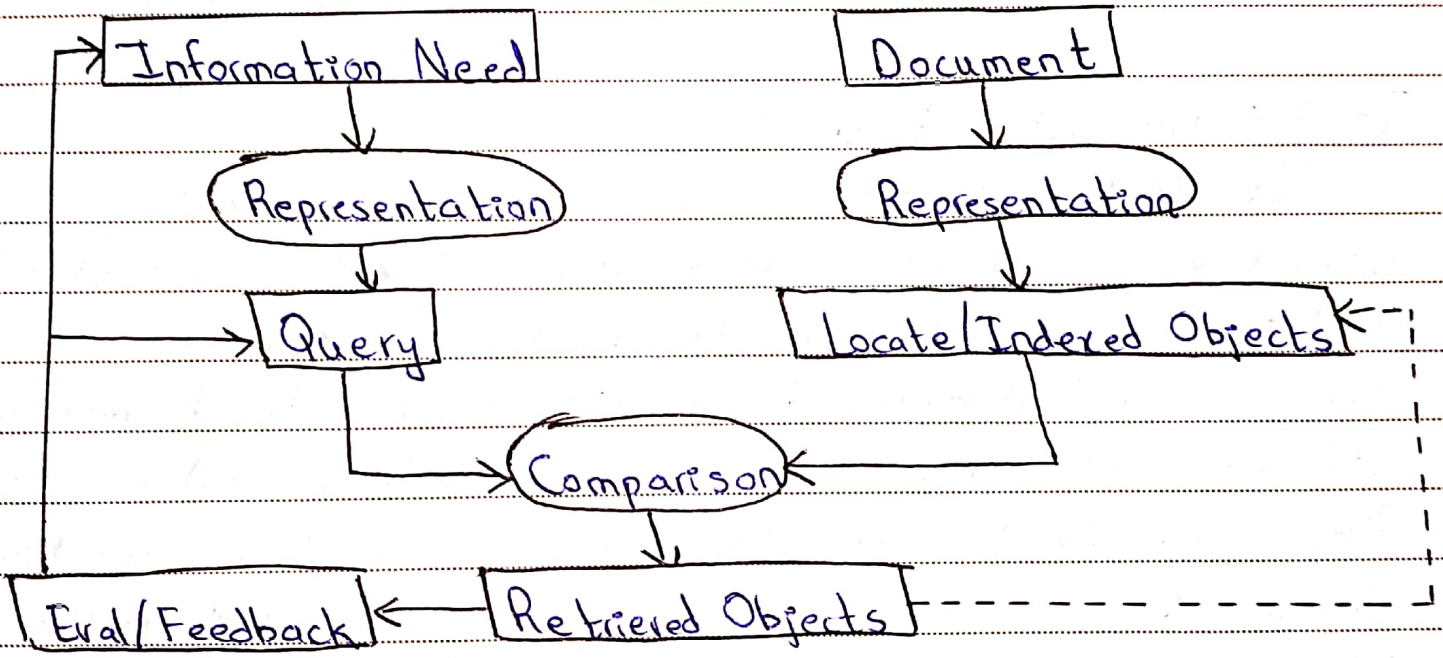
Dated: _____

★ Routines VIS Adhoc queries =>

Routines are predefined re-usable queries that are used to perform specific tasks stored in a DB.

Adhoc queries are one time, on-the-fly user generated queries that are executed as needed. They allow users to quickly and easily retrieve information from DB without the need for programming or routines.

★ Basic IR process =>



★ An IR system returns 8 relevant docs, 10 irrelevant.
Total = 20 relevant docs in collection.

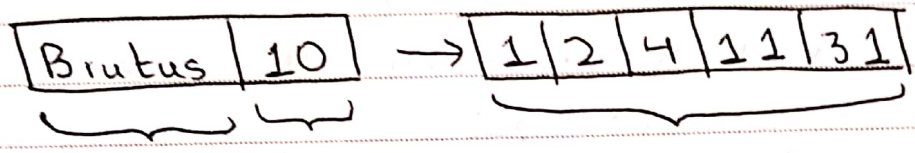
Precision = 8/18 = 0.44

Recall = 8/20 = 0.4

Dated: _____

BACK TO PRESENT TIMELINE

* **Inverted Index** => Actual index of the vocab that occurs in a collection. It consists of two parts:
  1) Lexicon / Dictionary / Vocabulary
  2) Posting / Posting List



term frequency posting list

↳ How are they made?

Consider two documents

Doc1 => My name is Wahaj Javed Alam.

Doc2 => I am Wahaj.

Step1 => List the terms and their docID.

My	1	Wahaj	1	I	2
name	1	Javed	1	am	2
is	1	Alam	1	Wahaj	2

Step2 => Sort the terms

Alam	1	is	1	name	1
am	2	Javed	1	Wahaj	1
I	2	My	1	Wahaj	2

Step3 => TADAA! Posting List time

Terms & Connectors → words & features → Transpose = Doc Term

Dated: _____

★ Term Document Matrix [possibly 2D Array] [DS]

Term	Play 1	Play 2	Play 3
Term 1	1	0	1
Term 2	0	0	0
Term 3	1	1	1

(Documents → Play 1, Play 2, Play 3; Term → Term 1, Term 2, Term 3; Matrix)

Since, we need Term1 and Term2 we AND them and not needed Term3 so !AND Term3.

Brutus && Caesar && !Calpurnia.

★ Problems ⇒ Sparse matrix → can be really huge for memory → we cannot machine read each doc → not noticing the order in each doc [only focus is features]

→ might make very simple or very complex queries.

→ shakespear has 44,000 words so matrix would become a 44K X 56 matrix. assuming there are 56 docs.

→ exact matching {forms of words} [morphs]

→ Complex query formulation {what is needed VS isn't}

→ user knows the features. → we can machine read each doc

→ user can formulate logical queries.

IR EVAL ASSUMPTIONS

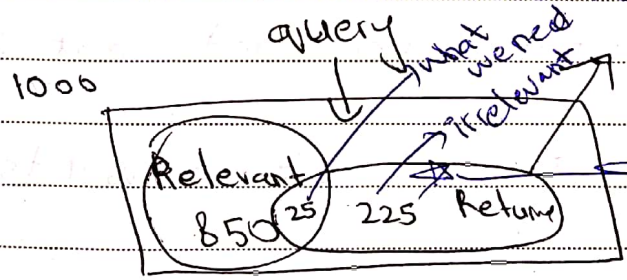
★ Effectiveness ⇒ Evaluation matrix

→ effectiveness is the quality of its search results.

→ Precision ⇒ What fraction of returned results are relevant to the need. {Term based}

{Doc based}

→ Recall ⇒ What fraction of relevant document in {static data} the collection were returned by the system.



Problems ⇒ can determine the two without knowing total docs.

→ Precision & Recall can be manipulated as needed.

★ Processing Boolean Queries =>

```
INTERSECT (p1, p2):   [Posting List Algorithm]
1   answer ← ()
2   while p1 && p2 :
3       if docID(p1) = docID(p2)
4           add (answer, docID(p1))
5           p1 ← next(p1)
6           p2 ← next(p2)
7       else if docID(p1) < docID(p2)
8           p1 ← next(p1)
9       else    p2 ← next(p2)
10  return answer
```

↳ How intersection works?

1) Find terms in dictionary
2) Retrieve posting lists
3) Intersect

★ Query Optimization => Selecting how to organize the work of answering a query so that the least total amount of work needs to be done by the system.

```
INTERSECT (t1, ..., tn):
1   terms ← sortByFrequency (t1, ..., tn)
2   result ← postings (first(terms))
3   terms ← rest(terms)
4   while terms AND result:        → previous one
5       result ← INTERSECT (result, postings(first(terms)))
6       terms ← rest(terms)
7   return result
```

=> $q_{11} = 1000$, $q_{12} = 500$, $q_{13} = 4$, $q_{14} = 4000$

optimized query = $q_{13} \wedge q_{12} \wedge q_{11} \wedge q_{14}$

Dated: _____

* Boolean Model (contd) =>

↳ Advantages =>

↳ Simplicity => easy to use and understand.

↳ Flexibility => allows user to search multiple terms and use operators to define criteria for search, thus conducting complex queries.

↳ Precision => only returns documents that exactly match the search criteria.

↳ Relevance => users can specify what they are looking for and exclude irrelevant results.

↳ Disadvantages =>

↳ Lack of Context => Searches do not consider context of terms.

↳ Limited Ranking => no ranking of results, so it is difficult to determine relevant documents.

↳ Complex Queries => May become too complex and difficult to manage on extremely restrictive queries or extremely simple.

↳ Inflexibility => no partial matches or synonyms. This can lead to a large number of false negatives. As a consequence, it might return too few or too many documents.

* Westlaw => Online legal research platform. It is a DB-driven IR system.

↳ AND/OR/NOT => simple AND {contract AND dispute}

↳ citations => '/s' parallel citation

↳ Proximity Operators => contract w/5 dispute [words should appear within 5 words]

↳ Wildcard searches => allows to search terms with a specific pattern.

{contra*] → contract, contractory