Module III

CHAPTER 3

Query Processing and Operations

Syllabus

Query Languages : Keyword based Querying, Pattern Matching, Structural Queries, Query Protocols; Query Operations: User relevance feedback, Multimedia IR models: Data Modeling

Self-learning Topics: Proximity Queries and Wildcard Queries

3.1 WHAT IS QUERY PROCESSING ?

GQ. Write the types of queries.

- Query Processing is the activity performed in extracting data from the database.
- In query processing, it takes various steps for fetching the data from the database.
- The steps involved are: Parsing, Translation and Optimization.
- The queries applied of structured and unstructured data stored in databases combined with information retrieval techniques can lead to faster and efficient processing of data.
- When a database is queried, it generates results using one of the multiple available plans.

3.1.1 Query Languages

(1) Keyword based Querying

(2) Pattern Matching

(3) Structural Queries

3.2 KEYWORD-BASED QUERYING

GQ. Write a short note on Keyword-based Querying.

- It is the simplest and most widely used kind of IR queries.
- It requires the user to simply enter phrase combinations to retrieve documents.
- The majority of times, people look for similar documents online using keywords.
- A logical AND operator creates an implied connection between the query keyword terms.
- When searching for "information retrieval," for example, the first retrieved result will be documents that contain both the phrases "information" and "retrieval".
- Additionally, the majority of systems also retrieve documents that merely contain the words "information" or "retrieval" in them.
- Before delivering the filtered query key-words to the IR engine, some systems preprocess the data by removing the most frequent words (stopwords), such as a, the, of, and so on. The order of these terms in the query is typically ignored by IR systems.
- Keyword searches are supported by all retrieval models.

3.2.1 Types of Keyword-Based Querying

GQ. What are the different types of Keyword-based Querying?

(a) Single Word Queries

(b) Context Queries
- Phrase
- Proximity

(c) Boolean Queries

OR, AND, BUT

(d) Natural Language

(e) Wildcard Queries

(a) Single Word Queries

- A query is formulated by a word.
- A document is formulated by long sequences of words.
- A word is a sequence of letters surrounded by separators, for example, a word 'on-line' .
- The division of the text into words is not arbitrary.
- Word queries return a list of documents that contain at least one of the query words.
- The level of similarity between the returned documents and the query determines their ranking.
- Term frequency and inverse document frequency are commonly used to support ranking.

(b) Context Queries

- Search words in a given context, that is, near other words
- Words that are near together suggest a higher possibility of relevance than words that are far apart.

Types :

(1) Phrase (2) Proximity

Phrase

- It is a sequence of single-word queries.
- An occurrence of the phrase is a sequence of words, for example, "enhance retrieval".
- The phrase is generally enclosed within double quotes.
- Each retrieved document must contain at least one instance of the exact phrase.

Proximity

- Proximity refers to search that accounts for how close within a record multiple items should be to each other.
- It is a more relaxed version of phrase query.
- Here, a sequence of single words or phrases, and a maximum allowed distance between them are specified.

- For example, "enhance retrieval" should occur within 4 words will match '...enhance the power of retrieval...'.
- The word or phrases may or may not be required to appear in the same order as in the query.

(c) Boolean Queries

- Boolean queries give a syntax composed of atoms that retrieve documents, and of Boolean operators which work on their operands.
- Some IR systems allow using the AND, OR, NOT, ( ), +, and – Boolean operators in combinations of keyword formulations.
- For example, translation AND syntax OR syntactic as shown in Fig. 3.2.1.

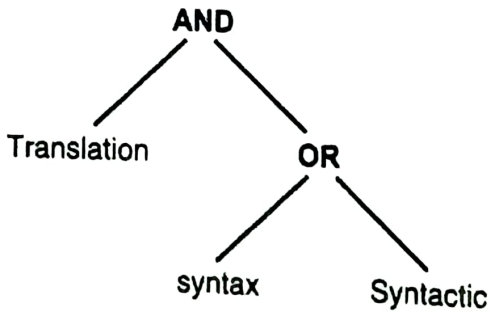

Fig. 3.2.1 : A query syntax tree

- AND requires that both terms be found.
- OR lets either term be found.
- NOT means any record containing the second term will be excluded.
- '( )' means the Boolean operators can be nested using parentheses.
- '+' is equivalent to AND, requiring the term; the '+' should be placed directly in front of the search term.
- '–' is equivalent to AND NOT and means to exclude the term; the '–' should be placed directly in front of the search term not wanted.
- Complex Boolean queries can be built out of these operators and their combinations, and they are evaluated according to the classical rules of Boolean algebra.
- No ranking is possible, because a document either satisfies such a query (is "relevant") or does not satisfy it (is "nonrelevant").
- A document is retrieved for a Boolean query if the query is logically true as an exact match in the document.

☞ **Fuzzy Boolean**

Retrieve documents appearing in some operands (The AND may require it to appear in more operands than the OR)

▶ **(d) Natural Language**

- It is generalization of "fuzzy Boolean".
- A query is an enumeration of words and context queries.
- All the documents matching a portion of the user query are retrieved.
- Few natural language search engines that aim to understand the structure and meaning of queries written in natural language text, generally as question or narrative.
- The system tries to formulate answers for these queries from retrieved results.
- Semantic models can provide support for this query type.

⏭ 3.3 PATTERN MATCHING

GQ. Define Pattern Matching.
GQ. Write a short note on Pattern Matching.

- Data retrieval : allow the retrieval of pieces of text that have some property (match a pattern)
- A pattern is a set of syntactic features that must occur in a text segment.

3.3.1 Types of Pattern Based Querying

GQ. What are the different types of Pattern Matching based Querying?

(a) Words

- Basic pattern
- A string which must be a word in the text

(b) Prefixes

- A string which must form the beginning of the text word
- For example, 'inter' in words 'interactive, international', etc

(c) Suffixes

- A string which must form the termination of the text word
- For example, 'dom' in words 'freedom, kingdom', etc

(d) Substrings

- A string which can appear within a text word
- For example, 'pal' in words 'palm, pals, principal, palace, municipality', etc.

(e) Ranges

- Matches any word lying between a pair of strings in lexicographical order (alphabetical order)
- For example, 'held' and 'hold' retrieve word such as 'hoax' and 'hissing'

(f) Allowing errors

- A word together with an error threshold
- Retrieve all text words which all 'similar' to the given word
- The pattern or text may have error typing, spelling or from optical character recognition.
- Models which can be used for information retrieval are

☞ **Edit distance**

- the minimum number of character insertions, deletions, and replacements needed to make two strings equal
- for example, 'flower' and 'flo wer' (edit distance 1)

☞ **Maximum allowed edit distance**

- query specifies the maximum number of allowed errors for a word to match the pattern
- extended to search substring and not only words

(g) Regular expressions

General pattern built up by simple string and following operators :

- union: if e1 and e2 are regular expressions, then (e1|e2) matches what e1 or e2 matches

- concatenation: if e1 and e2 are regular expressions, the occurrences of (e1e2) are formed by the occurrences of e1 immediately followed by those of e2
- repetition: if e is a regular expression, then (e*) matches a sequence of zero or more contiguous occurrence of e
- 'pro(blem|tein) (s|e)(0|1|2)*'-> 'problem2' and 'proteins'

3.4 STRUCTURAL QUERIES

GQ. What are structural Queries? Give different structures.

GQ. Explain the following data structures giving suitable examples
(a) Fixed (b) Hypertext (c) Hierarchical

- Allow user to query the text on their structure
- Mixing contents and structure in queries
  - **Contents :** words, phrases, or patterns
  - **Structural constraints :** containment, proximity, or other restrictions on structural elements
- Three main structures
  (a) Fixed structure
  (b) Hypertext structure
  (c) Hierarchical structure

3.4.1 Fixed Structure

- Fixed structure for text retrieval such as Form which is shown in Fig. 3.4.1
- **For example :** Mail archive
- Each mail has a sender, a receiver, a date a subject and a body field as a fixed structure.
- Easy to search a mail based on a date, a receiver, a subject and so on
- **Other examples :** Log file (Document : a fixed set of fields)

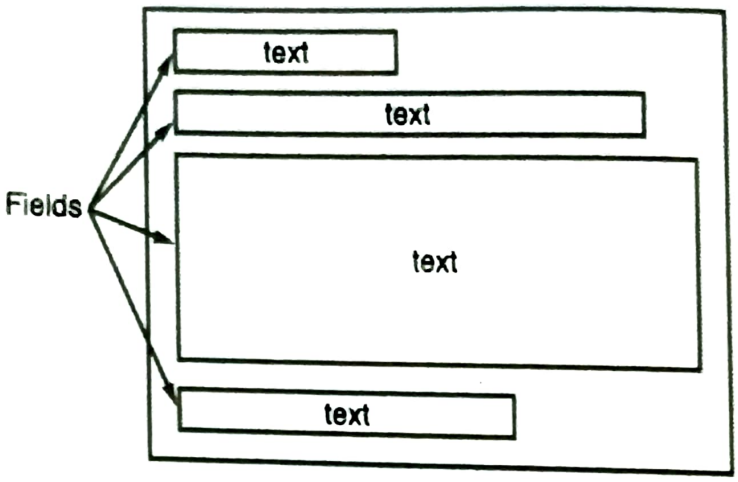

Fig. 3.4.1 : Form-like fixed structure

3.4.2 Hypertext Structure

- Search by content and structure
- A hypertext is a directed graph where nodes hold some text (text contents). The links represent connections between nodes or between positions inside nodes (structural connectivity).
- The user had to manually traverse the hypertext nodes following links to search what he wanted.
- Fig. 3.4.2 represents a hypertext structure with nodes and links

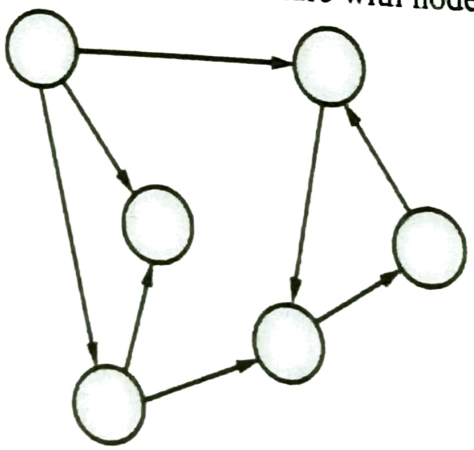

Fig. 3.4.2 : Hypertext structure

- Hypertext : Web Glimpse
- Web Glimpse combine browsing and searching on the Web.
- It supports traditional navigation and enables searching for content nearby the current node.

3.4.3 Hierarchical Structure

- Intermediate level of flexibility
- Lies between fixed structure and hypertext structure
- Represent the recursive decomposition of the text
- Fig. 3.4.3 represent a schematic view of Hierarchical structure

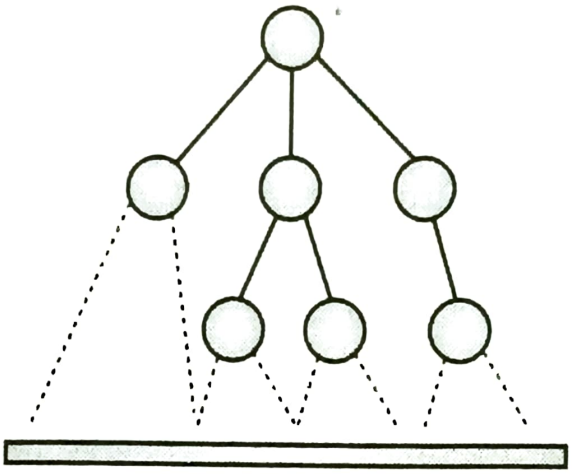

Fig. 3.4.3 : Hierarchical structure

- Fig. 3.4.4 as shown below gives an example of hierarchical structure with the page of book, its semantic view and a parsed query to retrieve the figure

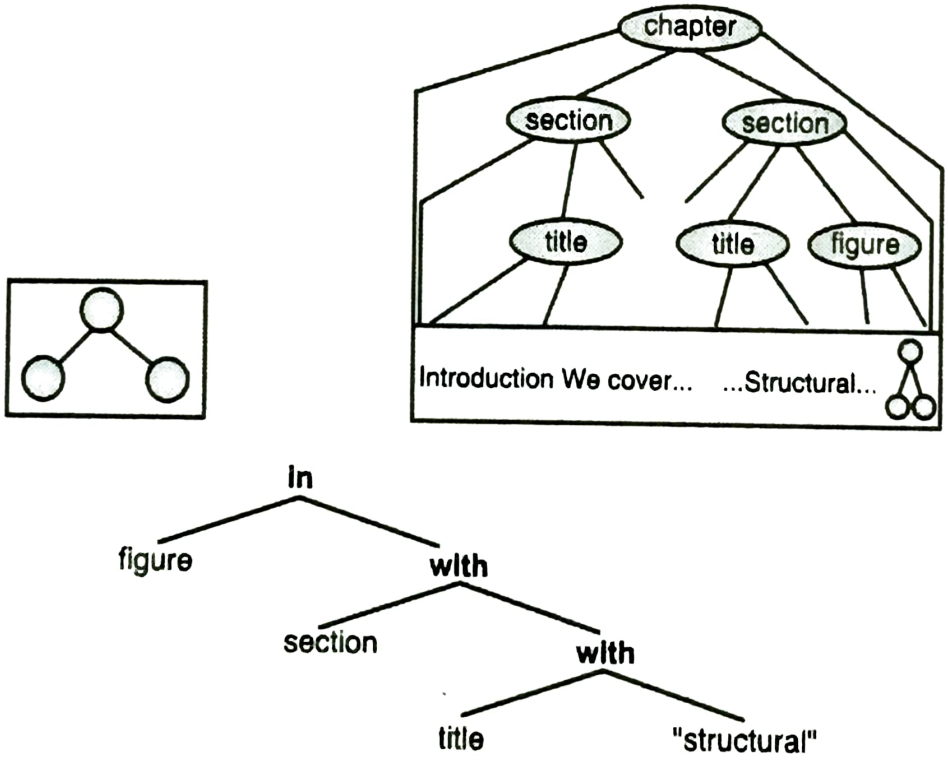

Fig. 3.4.4 : An example-hierarchical structure

3.5 SAMPLE OF HIERARCHICAL MODELS

GQ. Discuss different sample of hierarchical models.
GQ. Explain PAT data structure with suitable examples.

3.5.1 PAT Expressions

- Built on the same index as the text index
- Structure is presumptively indicated in the text by tags
- Structure is defined in terms of initial and final tags
- Region is defined by each pair of initial and final tags
- The model allow for the areas of a region to overlap or nest
- PAT is a text searching system
- Developed at University of Waterloo
- PAT interprets text as a set of suffix strings
- For example, indexing every word in this sentence yields the 12 strings :
    - For example, indexing every word in this sentence yields the 12 strings
    - example, indexing every word in this sentence yields the 12 strings
    - indexing every word in this sentence yields the 12 strings
    - every word in this sentence yields the 12 strings
    - word in this sentence yields the 12 strings
    - in this sentence yields the 12 strings
    - this sentence yields the 12 strings
    - sentence yields the 12 strings
    - yields the 12 strings
    - the 12 strings
    - 12 strings
    - strings

3.5.2 Overlapped Lists

- The model considers the use of an inverted list to index words as well as regions.

- The model allows to perform set union and to combine regions.
- The model allows for the areas of a region to overlap, but not to nest.
- A 'followed by' operator adds the extra restrictions requiring that the first region come before the second area.
- An 'n words' operator creates the region containing all text's sequences of n words.
- It is not clear, whether overlapping is good or not for capturing the structural properties.

3.5.3 Lists of References

- Model makes the definition and querying of structured text uniform
- The structure of the document is fixed and hierarchical
- All possible regions are defined at indexing time
- Overlap and nest are not allowed
- All elements must be of the same type, e.g. only sections, or only paragraphs.
- Answer to the query is seen as list of 'references'
- A reference is a pointer to a region of the database.

3.5.4 Proximal Nodes

- This model tries to find a good compromise between expressiveness and efficiency.
- It does not define a specific language, but a model in which it is shown that a number of useful operators can be included achieving good efficiency.
- The structure of the document is fixed and hierarchical.
- The model allows nested elements but no overlaps.

3.5.5 Tree Matching

- The model relies on tree inclusion.

- Interprets the structure of both the text database and the query as a tree to determine the embedding of the query into the database respecting the hierarchical relationships between the query's nodes.
- The leaves of the query can be not only structural elements but also text patterns, meaning that the ancestor of the leaf must contain that pattern.

3.6 QUERY PROTOCOLS

GQ. Write a short note on Query Protocols.

(a) Z39.50

- Approved by American National Standards Institute (ANSI) and National Information Standards Organization (NISO) in 1995
- Can be implemented on any platform
- Query bibliographical information using a standard interface between the client and the host database manager
- With query language, the protocol also specifies a way in which client and server establish a session, communicate and exchange information, etc.
- Z39.50 protocol is part of WAIS
- Z39.50 Brief history
- Work on the Z39.50 protocol began in the 1970s and led to successive versions in 1988, 1992, 1995 and 2003
  - Z39.50-1988(version 1)
  - Z39.50-1992(version 2)
  - Z39.50-1995(version 3)
  - Z39.50-2003(version 4)

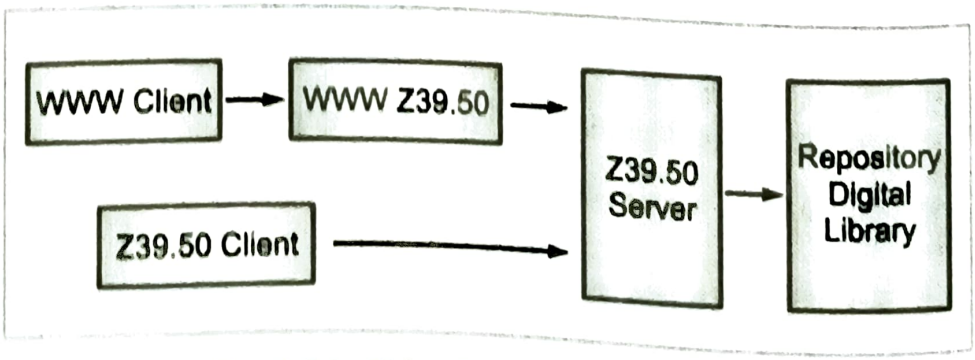

Fig. 3.6.1 : Using Z39.50 over the WWW

(b) WAIS

- Wide Area Information Service
- Beginning in the 1990s
- Network publishing protocol
- Query databases through the Internet

(c) CCL

- Common Command Language
- NISO proposal based on Z39.50
- Defines 19 commands
- More popular in Europe
- Based on the classical Boolean model

(d) CD-RDx

- Compact Disk Read only Data exchange
- Uses client server architecture on most platforms
- Client is generic
- Server is designed and provided by the CD-ROM publisher
- Allows fixed length fields, images and audio
- Supported by CIA, NASA and GSA

(e) SFQL

- Structured Full-text Query Language
- Based on SQL
- Uses client server architecture
- Adopted as a standard by aerospace community
- Documents are rows in relational tables which are tagged using SGML
- Answer format has header and message area

3.7 TRENDS AND RESEARCH ISSUES

- Table 3.7.1 shows the different basic queries allowed in the different models.

- Probabilistic and Bayesian Belief Network (BBN) models can also incorporates set operations.

Table 3.7.1 : Relationship between types of queries and models

Model	Queries allowed
Boolean	Queries allowed
Vector	Words
Probabilistic	Words
Bayesian Belief Network	Words

3.8 QUERY LANGUAGE TAXONOMY

- Fig. 3.8.1 represents the types of operations covered so far and how they can be structured.

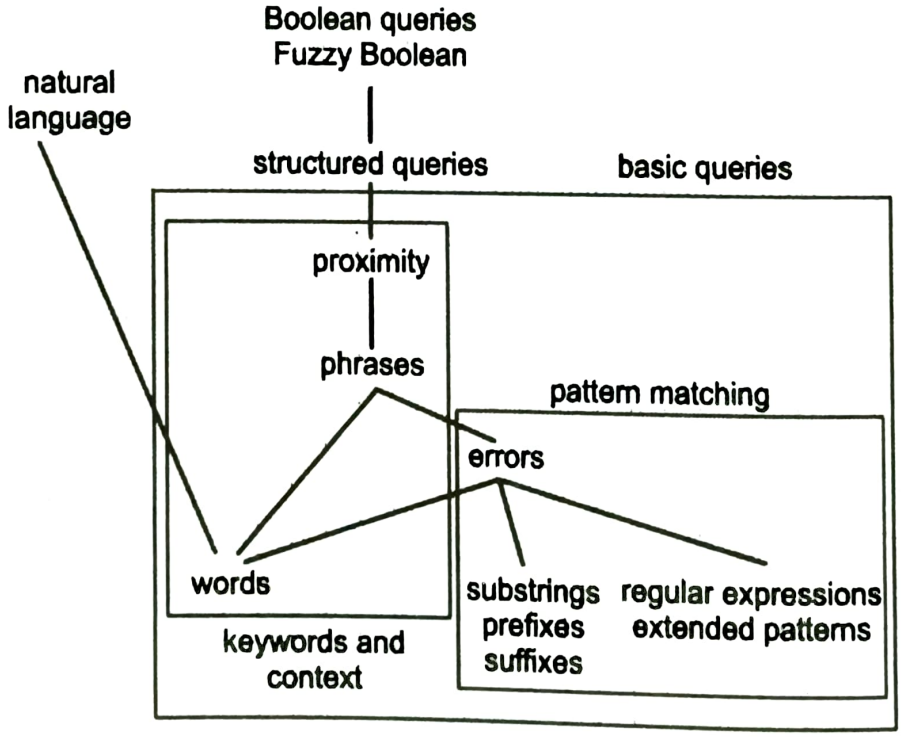

Fig. 3.8.1 : Query Language Taxonomy

3.9 QUERY OPERATIONS

- It is difficult to formulate queries which are well designed for retrieval purposes.
- Improving the initial query formulation through query expansion and term reweighting

- Approaches based on :
  - feedback information from the user
  - information derived from the set of documents initially retrieved (called the local set of documents).
  - global information derived from the document collection

3.10 USER RELEVANCE FEEDBACK

GQ. Define Relevance feedback model.

GQ. Give short notes for User Relevance Feedback. OR Give brief notes about user Relevance feedback method and how it is used in query expansion

GQ. What are the two basic approaches in User Relevance Feedback for query processing?

- User receives a list of searched documents and, after reviewing, marks the relevant documents
- A selection of key terms or expressions that are attached to the document and identified by the user as relevant
- **Definition : Relevance Feedback Model**

  After initial retrieval, results are presented, allow the user to provide feedback on the relevance of one or more of the retrieved documents. Use this feedback information to reformulate the query and produce new results based on reformulated query. Thus allows more interactive multi pass process.
- **Two basic operations :**
  - **Query expansion :** addition of new terms from relevant document (Expand queries with the vector model)
  - **Term reweighting :** modification of term weights based on the user relevance judgement
- The usage of user relevance feedback to :
  (a) expand queries with the vector model
  (b) reweight query terms with the probabilistic model
  (c) reweight query terms with a variant of the probabilistic model

3.11 VECTOR MODEL

GQ. How do you calculate the term weighting in document and Query term weight in Vector Model?

Define :

- **Weight :**

  Let the k_i be a generic index term in the set $K = \{ k_1, \ldots, k_t \}$

  A weight $w_{i,j} > 0$ is associated with each index term k_i of a document d_j

- **document index term vector :**

  the document d_j is associated with an index term vector d_j represented by

$$\vec{d_j} = (w_{1,j}, w_{2,j}, \ldots, w_{t,j}) \quad \ldots(3.11.1)$$

- **the term weighting :**

$$w_{i,j} = f_{i,j} \times \log \frac{N}{n_i} \quad \ldots(3.11.2)$$

- **the normalized frequency :**

$$f_{i,j} = \frac{freq_{i,j}}{\max_i freq_{i,j}} \quad \ldots(3.11.3)$$

  $freq_{i,j}$ be the raw frequency of k_i in the document d_j

- **inverse document frequency for k_i :**

$$idf_i = \log \frac{N}{n_i} \quad \ldots(3.11.4)$$

- **the query term weight :**

$$w_{i,q} = \left(0.5 + \frac{0.5\, freq_{i,q}}{\max_l freq_{l,q}} \right) \times \log \frac{N}{n_i} \quad \ldots(3.11.5)$$

- **query vector :**

  query vector q is defined as

  D_r : set of relevant documents identified by the user

  D_n : set of non-relevant documents among the retrieved documents

  C_r : set of relevant documents among all documents in the collection

  α, β, γ : tuning constants

3.11.1 Query Expansion and Term Reweighting for the Vector Model

GQ. What are the three classic and similar ways to calculate the modified query qm?

- Ideal case C_r : the complete set C_r of relevant documents to a given query q
  - the best query vector is presented by

$$\vec{q}_{opt} = \frac{1}{|C_r|} \sum_{\forall \vec{d}_j \in C_r} \vec{d}_j - \frac{1}{N - |C_r|} \sum_{\forall \vec{d}_j \in C_r} \vec{d}_j \quad \ldots(3.11.6)$$

- The relevant documents C_r are not known a priori, should be looking for them
- 3 classic and similar way to calculate the modified query are
  - Standard_Rochio:

$$\vec{q}_m = \alpha \vec{q} + \frac{\beta}{|D_r|} \sum_{\forall \vec{d}_j \in C_r} \vec{d}_j - \frac{\gamma}{|D_n|} \sum_{\forall \vec{d}_j \notin C_r} \vec{d}_j \quad \ldots(3.11.7)$$

  - Ide_Regular :

$$\vec{q}_m = \alpha \vec{q} + \beta \sum_{\forall \vec{d}_j \in D_r} \vec{d}_j - \gamma \sum_{\forall \vec{d}_j \notin D_n} \vec{d}_j \quad \ldots(3.11.8)$$

  - Ide_Dec_Hi :

$$\vec{q}_m = \alpha \vec{q} + \beta \sum_{\forall \vec{d}_j \in D_r} \vec{d}_j - \gamma \max_{non-relevant} \left(\vec{d}_j \right) \quad \ldots(3.11.9)$$

- The D_r and D_n are the document sets which the user judged
- The Rochio formulation is basically a direct adaptation of Equation (3.11.6) in which the terms of the original query are added in.
- **Advantages :** Simplicity and good result
- **Disadvantages :** No optimality criterion is adopted

3.12 TERM REWEIGHTING FOR THE PROBABILISTIC MODEL

GQ. How do you calculate the term reweighting in Probabilistic Model?

- Similarity : the correlation between the vectors d_j and this correlation can be quantified as :

$$sim(d_j, q) = \frac{\vec{d_j} \bullet \vec{q}}{|\vec{d_j}| \times |\vec{q}|} \quad ...(3.12.1)$$

- The probabilistic model according to the probabilistic ranking principle.
  - **$P(k_i | R)$:** The probability of observing the term k_i in the set R of relevant document
  - **$P(ki | \bar{R})$** : the probability of observing the term k_i in the set $\bar{R}$ of non-relevant document
- The similarity of a document dj to a query q can be expressed as

$$sim(d_j, q) \propto \sum w_{i,q}\, w_{i,j} \left(\log \frac{P(k_i | R)}{1 - P(k_i | R)} + \log \frac{P(k_i | \bar{R})}{1 - P(k_i | \bar{R})} \right) \quad ...(3.12.2)$$

- For the initial search
  - estimated above equation by following assumptions

$$P(k_i | R) = 0.5$$

$$P(k_i | \bar{R}) = \frac{n_i}{N}$$

n_i is the number of documents which contain the index term k_i get

- For the feedback search
  - The $P(k_i | R)$ and $P(k_i | R)$ can be approximated as:

$$P(k_i | \bar{R}) = \frac{n_i - |D_{r,i}|}{N - |D_r|} \quad ...(3.12.3)$$

$$P(k_i | R) = \frac{|D_{r,i}|}{|D_r|} \quad ...(3.12.4)$$

the D_r is the set of relevant documents according to the user judgement

the $D_{r,i}$ is the subset of D_r composed of the documents contain the term

k_i

- The similarity of d_j to q :

$$\text{sim}(d_j, q) = \sum_{i=1}^{t} w_{i,q}\, w_{i,j} \log \left(\frac{|D_{r,i}|}{|D_r| - |D_{r,i}|} \Big/ \frac{n_i - |D_{r,i}|}{N - |D_r| - (n_i - |D_{r,i}|)} \right)$$

...(3.12.5)

There is no query expansion occurs in the procedure

- Adjustment factor
  - Because of $|D_r|$ and $|D_{r,i}|$ are certain small, take a 0.5 adjustment factor added to the $P(k_i | R)$ and $P(k_i | R)$

$$P(k_i | R) = \frac{|D_{r,i}| + 0.5}{|D_r| + 1} \quad ...(3.12.6)$$

$$P(k_i | \bar{R}) = \frac{n_i - |D_{r,i}| + 0.5}{N - |D_r| + 1} \quad ...(3.12.7)$$

  - Alternative adjustment factor n_i / N

$$P(k_i | R) = \frac{|D_{r,i}| + \frac{n_i}{N}}{|D_r| + 1} \quad ...(3.12.8)$$

$$P(k_i | \bar{R}) = \frac{n_i - |D_{r,i}| + \frac{n_i}{N}}{N - |D_r| + 1} \quad ...(3.12.9)$$

☞ Advantages

(1) Feedback process is directly related to the derivation of new weights for query terms and that the term reweighting is optimal under the assumptions of term independence and binary document indexing.

☞ Disadvantages

(1) Document term weights are not taken into account during the feedback loop.

(2) Weights of terms in the previous query formulations are also disregarded.

(3) No query expansion is used.

3.13 A VARIANT OF PROBABILISTIC TERM REWEIGHTING

GQ. Discuss variant of probabilistic term reweighting?

- 1983, Croft extended above weighting scheme by suggesting distinct initial search methods and by adapting the probabilistic formula to include within-document frequency weights
- The variant of probabilistic term reweighting :

$$sim(d_j, q) \propto \sum_{i=1}^{t} w_{i,q}\, w_{i,j}\, F_{i,j,q} \qquad ...(3.13.1)$$

  - the $F_{i,j,q}$ is a factor which depends on the triple $[k_i, d_j, q]$.
  - using distinct formulations for the initial search and feedback searches
- **Initial search :**

$$F_{i,j,q} = (C = id f_i)\, \bar{f}_{i,j} \qquad ...(3.13.2)$$

$$\bar{f}_{i,j} = K + (1 + K)\frac{f_{i,j}}{\max(f_{i,j})} \qquad ...(3.13.3)$$

The $f_{i,j}$ is a normalized within-document frequency C and K should be adjusted according to the collection

- **Feedback searches :**

$$F_{i,j,q} = \left(C + \log \frac{P(k_i \mid R)}{1 - P(k_i \mid R)_i} + \log \frac{1 - P(k_i \mid \bar{R})}{P(k_i \mid \bar{R})} \right) \bar{f}_{i,j} \qquad ...(3.13.4)$$

☞ Advantages

(1) Consider within document frequency.

(2) Adopts a normalized frequencies.

(3) Introduces constant C and K to provide greater flexibility.

☞ Disadvantages

(1) Constitutes more complex formulation.

(2) No query expansion.

3.14 MULTIMEDIA IR

GQ. Discuss the architecture of Multimedia IR system.
GQ. Give basic steps for data retrieval in Multimedia IR system.
GQ. Write a short note on data retrieval in Multimedia IR system.

- The architecture of a Multimedia IR system depends on two main factors
  (1) The peculiar characteristics of multimedia data
  (2) The kinds of operations to be performed on such data
- Multimedia IR system support variety of data and different kinds of media
  (1) Text, images (both still and moving), graphs, and sound
  (2) Mix of structured and unstructured data
  (3) Metadata
  (4) Semi-structured data
  (5) Data whose structure may not match, or only partially match, the structure prescribed by the data schema
  (6) The system must typically extract some features from the multimedia objects.

Data retrieval

- Exploiting data attributes and the content of multimedia objects
- Basic steps for data retrieval :

(1) **Query specification**

- Fuzzy predicates (Find all images similar to a car)
- Content-based predicates (Find all objects containing an apple)
- Object attributes (Find all red images)
- Structural predicates (Find all multimedia containing a video clip).

(2) **Query processing and optimization**

Query is parsed and compiled into an internal form

(3) **Query answer**

The retrieved answers are returned to the user in decreasing order of relevance

(4) Query iteration

- The query execution is iterated until the user is satisfied
- Combine DBMS and IR technology
  - **DBMS :** Data modeling capabilities
  - **IR system :** Similarity-based query capabilities

3.15 DATA MODELING

GQ. Explain the role of data modeling in Multimedia IR system.

Main tasks in data modeling are

(1) A data model should be defined by which the user can specify the data to be stored into the system
  - Support conventional and multimedia data types
  - Provide methods to analyze, retrieve, and query such data

(2) Provide a model for the internal representation of multimedia data

Object-oriented DBMS

- Provide rich data model
  - More suitable for modeling both multimedia data types and their semantic relationships
- Class in OODBMS is characterized by both attributes and set of operations
- Classes are also related to inheritance hierarchies hence multimedia class is a specialization of one or more super classes

Drawback of OODBMSs

- The performances of storage techniques, query processing, and transaction management is not comparable to that of relational DBMSs
- Highly non-standard
- Object-relational DBMS
- Extend the relational model
  - Represent complex data types
  - Maintain the performance and the simplicity of relational DBMSs

and related query languages

- Define abstract data types to allow one to define ad hoc data types for multimedia data.

- Multimedia data representation inside the system
  - Using attributes is not sufficient to describe data
  - Information extracted from objects to use during query processing
  - Multimedia object is represented as a set of features
  - Features can be assigned manually, automatically, or using a hybrid approach
  - Values of some specific features are assigned to an object by comparing the object with some previously classified objects
  - Feature extraction cannot be precise
  - A weight is usually assigned to each feature value representing the uncertainty of assigning such a value to that feature
  - For example, 80% sure that a shape is a square

3.16 SQL3

GQ. Explain the role of SQL3 in Multimedia IR system.
GQ. Write a short note on MULTOS Data Model.
GQ. Explain MULTOS Data Model with proper example.

- Support extensible type system
  - Provide constructs to define user-dependent abstract data types, in an object-oriented like manner
- Provides three types of Collection data types
  - Sets, multisets, and lists
  - The elements of a collection must have compatible types
- Provides a restricted form of object identifier that supports sharing and avoids data duplication.

  Example : MULTOS (MULTimedia Office Server)
- Multimedia document Server
  - Client/server
  - Support filing and retrieval of multimedia objects

- Documents are described by :
  - logical structure: title, into, chapter, ...
  - layout structure: pages, frames, ...
  - conceptual structure: allows content-based queries
  - Docs similar in conceptual structures are grouped into *conceptual types*
  - **Example :** Generic_Letter
  - Conceptual structure of the type Generic_Letter and Business_Product_Letter is shown in Fig. 3.16.1 and Fig. 3.16.2 respectively

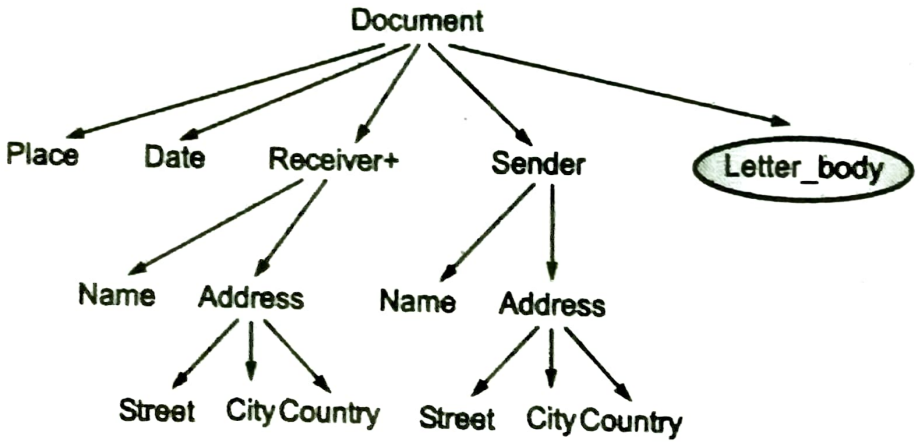

Fig. 3.16.1 : Conceptual structure of the type Generic_Letter

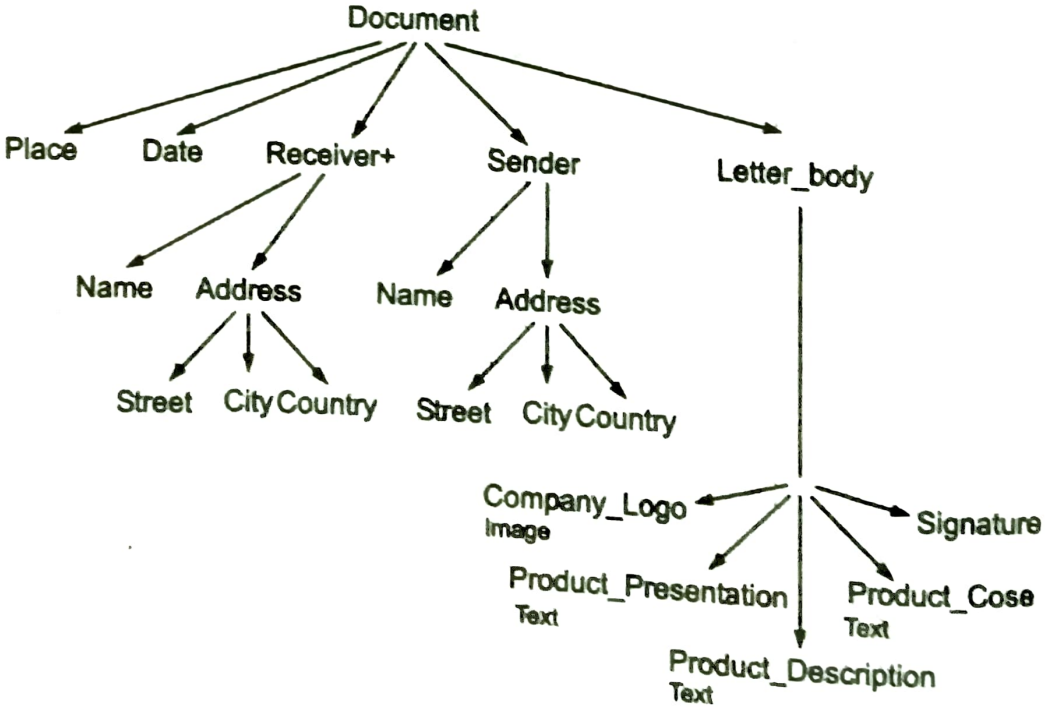

Fig. 3.16.2 : Complete conceptual structure of the type Business_Product_Letter

Image data in MULTOS

- Analysis
  - **low level :** detect objects and positions
  - **high level :** image interpretation
- Result of analysis
  - description of objects found and their classes
  - certainty values
- Indices are used for fast access to this info
  - Object index. Includes pointers to objects and certainty values
  - Cluster index, with fuzzy clusters of similar images

3.17 WILDCARD QUERIES

- It supports regular expressions and pattern matching-based searching in text.
- Retrieval models do not directly support for this query type.
- In IR systems, certain kinds of wildcard search support may be implemented.
- **Example :** Usually words ending with trailing characters (for example, 'data*' would retrieve *data, database, datapoint, dataset,* and so on).
- Providing support for wildcard searches in IR systems involves preprocessing over-head and is not considered worth the cost by many Web search engines today.

3.18 GENERAL QUESTIONS

Q. 1 What are the advantages and disadvantages of query processing?

☑ **Ans. :**

Advantages

(1) It is simple : The fact that the modified term weights are computed directly from the set of retrieved documents.

(2) It gives good results : Observed experimentally and are due to the fact that the modified query vector does reflect a portion of the intended query semantics.

Disadvantages

(1) No optimality

Q. 2 Explain the process for User Relevance Feedback method.

☑ **Ans. :**

- It is the most popular query formulation strategy.
- In a relevance feedback cycle, the user presented with a list of the retrieved documents .
- Then examine them, marks those which are relevant.
- Only to 10 (or 20) ranked documents are examined.
- Selecting important terms, or expression, attached to the documents
- Enhancing the important of these terms in a new query formulation
- The new query will be
  - (1) Moved towards the relevant documents
  - (2) Away from the non-relevant ones.

Q. 3 What are the Advantages of User Relevance Feedback method?

☑ **Ans. :**

(1) It shields the user from the details of the query reformulation process because all the user has to provide is a relevance judgement on documents.

(2) It breaks down the whole searching task into a sequence of small steps which are easier to grasp.

(3) It provides a controlled process designed to emphasize some terms (relevant ones) and de-emphasize others (non-relevant ones)

Q. 4 Discuss the Parameters used in calculating a weight for a document term or query term?

☑ **Ans. :**

(1) Term Frequency (tf) : Term Frequency is the number of times a term i appears in document j (tfij).

(2) Document Frequency (df) : Number of documents a term i appears in, (dfi).

(3) Inverse Document Frequency (idf) : A discriminating measure for a term i in collection, i.e., how discriminating term i is. (idf i) = log10(n / dfi), where n is the number of document.

Chapter Ends...

❑❑❑