BE – Class A & B

AY 2023-24

ISR ORAL QB

Question Bank:

Unit 1: Introduction to information Retrieval

1. Need for automatic text analysis

Ans : Automatic text analysis is essential in various domains due to the increasing volume of textual data being generated daily. Manual analysis of this data is time-consuming and often impractical, especially in scenarios where real-time or near-real-time processing is required. Automatic text analysis tools and algorithms allow for the efficient processing, extraction, and interpretation of relevant information from large text datasets, enabling timely decision-making and providing valuable insights across various applications such as intelligence analysis, sentiment analysis, information retrieval, and more.

2. Explain the Conflation algorithm

Ans: Conflation algorithm is the method to find the document representative. It is based on the Luhn's idea. The algorithm consists of three parts as follows

i. Removal of high frequency words :

The removal of high frequency words i.e. 'stop" words or 'fluff words is one way of implementing Luhn's upper cut-off. This is done by comparing each word from the document to the list of high frequency words. High frequency words are those words which comes more number of times in the text.These words does not contain meaning or semantic of the text. For e.g. is, am, I, are, the etc. are some of words. The advantage of this process is not only the non-significant words are removed but also the size of the document can be reduced to 30 to 50%.

ii. Suffix stripping :

In this step, each word is handled from the output of first step. If any word is having the suffix, then the suffix gets removed and the word is converted in its original form. For e.g. If the word is "Killed" it will be converted into 'Kill’ . Unfortunately, context free removal leads to a significant error rate. For e.g. we may want UAL removed from FACTUAL but not from EQUAL.

iii. Detecting equivalent stems :

After suffix stripping, we will have the list of words. Only one occurrence of the word is kept in the list for e.g. if two words "processing" and "processed" get converted into 'process' only one occurrence of process will be part of the list. Each word is called as 'stem'.

3. What is indexing and, Probabilistic indexing?

Ans:

- Indexing:

  During "Indexing" documents are prepared for use by Information Retrieval system. This means preparing the raw document collection into an easily accessible representation of documents. This transformation from a document text into a representation of text is known as indexing the documents. Transforming a document into an indexed form involves the use of:

  • A library or set of regular expressions

  • Parsers

  • A library of stop words (a stop list)

  • Other miscellaneous filters

  Conflation algorithm is used for converting document into its representation i.e. indexing. Each element i.e. word in index language is referred as index term.

- Probabilistic indexing:

  Probabilistic indexing is a technique used in information retrieval to rank documents based on their relevance to a given query. It employs probabilistic models to estimate the likelihood of a document being relevant to a specific query. The most well-known example of this approach is the Okapi BM25 algorithm, which considers factors such as term frequency, document length, and inverse document frequency to calculate relevance scores.

4. What is the use of Clustering in information retrieval, Explain Rocchio's Algorithm?

Ans:

- Clustering in Information Retrieval:

  Clustering is a technique used to group similar documents together based on their content or features. In information retrieval, this can be valuable for organizing search results or for exploratory data analysis. It helps in identifying patterns and relationships within a dataset, making it easier for users to navigate and find relevant information.

- Rocchio’s Algorithm:

  Rocchio’s Algorithm is a relevance feedback method used in information retrieval. It’s based on the idea that user feedback (both positive and negative) can be used to refine search results. The algorithm adjusts the weights of query terms based on their relevance to the user’s information needs. Specifically, terms from relevant documents are given more weight, while terms from irrelevant documents are given less weight. This iterative process helps improve the precision of search results over time.

5. Explain Single Pass and Single link Algorithm

Ans:

- Single Pass Algorithm:

  Single Pass clustering is a method of grouping data points into clusters in a single pass through the dataset. The algorithm processes data sequentially and assigns each data point to a cluster as it encounters it.

  Single-pass algorithm process as follows :

  - The object descriptions are processed serially.
  - The first object becomes the cluster representative of the first cluster.
  - Each subsequent object is matched against all cluster representatives existing at its

processing time.

  - A given object is assigned to one cluster (or more if overlap is allowed) according to some condition on the matching function.
  - When an object is assigned to a cluster the representative for that cluster is recomputed.
  - It an object fails a certain test it becomes the cluster representative of a new cluster.

- Single Link Algorithm:

  Single Link clustering is a hierarchical clustering method that starts by considering each data point as a separate cluster and then iteratively merges the two clusters that are closest to each other based on a chosen distance metric. It continues this process until a predefined stopping criterion is met, resulting in a hierarchical tree or dendrogram. The "distance" between clusters is often defined as the minimum distance between any two points in the different clusters (hence the name "single link").

Unit 2: Indexing and Searching Techniques

1. What are B Structures used in Inverted Files

Ans: B-Structures are a type of data structure used in conjunction with inverted files to facilitate efficient retrieval of information. They help organize and manage the inverted file index efficiently. These structures, which can include B-Trees or similar balanced search trees, are used to store pointers to the actual inverted lists or postings for each term. This enables faster access to the relevant documents containing the searched terms, contributing to improved search performance in information retrieval systems.

2. What is Inverted Files

Ans: Inverted Files, also known as inverted indexes, are data structures used in information retrieval systems to facilitate quick search and retrieval of documents containing specific terms or keywords. Unlike forward indexes that map documents to the terms they contain, inverted files store an index of terms along with a list of documents where each term appears. This allows for efficient retrieval of documents based on user queries, making them a fundamental component of search engines and other information retrieval systems.

3. Explain B-trees

Ans: B-trees are self-balancing tree data structures that are commonly used in database systems and file systems for efficient storage and retrieval of data. They are characterized by their ability to maintain a balanced structure even after insertions and deletions, which ensures relatively uniform access times. In a B-tree, each node can have multiple children and keys, and the keys are stored in sorted order within each node. This property allows for efficient searching, insertion, and deletion operations, making B-trees a crucial data structure for managing large datasets.

4. Explain Tries

Ans: Tries, also known as digital trees or prefix trees, are tree data structures used for efficient retrieval of strings or sequences of characters. Each node in a trie represents a single character, and paths from the root to the leaves form words or sequences. Tries are particularly useful for tasks like string matching, autocomplete, and searching for words with common prefixes.

5. What is Suffix Tree and Suffix Array

Ans :

- Suffix Tree:

  A suffix tree is a data structure used for efficient string matching and substring search. It stores all the suffixes of a given string in a way that allows for fast searching operations. Suffix trees are commonly used in applications like bioinformatics for tasks such as DNA sequence analysis.
- Suffix Array:

  A suffix array is an array that contains the starting positions of all suffixes of a given string, sorted in lexicographic order. Suffix arrays can be used in place of suffix trees in many applications, as they require less memory and can be more space-efficient. They are particularly useful in scenarios where memory is a critical consideration.

6. Explain Signature Files

Ans: Signature files are index structured based on hashing and are word oriented. They posses a low overhead at the cost of forcing a sequential search over the index. Search complexity is linear but constant. It is not suitable for very large texts.
Signature file used hash function which maps words to bit masks of B bits. It divides text in blocks of 3 words each. Then it assigns a bit mask of size B to each text block of size b. The mask is obtained by performing bitwise ORing the signatures of all the words in the text block. So signature file is the sequence of bits masks of all blocks with a pointer to each block. If the word is present in a text block, then all the bits set in its signature are also set in the bit set in its signature are also set in the bit mask of the text block. Whenever a bit is set in the mask of the query word and not in the mask of the text block then the word is not present in the text block

7. Explain any one Searching Techniques

Ans: Boolean search is a type of search method used in information retrieval systems. It allows users to combine keywords with operators like "AND," "OR," and "NOT" to refine their search queries and retrieve more specific and relevant results.

- AND: Retrieves documents that contain all of the specified terms. For example, a search for "cats AND dogs" will return documents that mention both cats and dogs.
- OR: Retrieves documents that contain at least one of the specified terms. For example, a search for "cats OR dogs" will return documents that mention either cats or dogs or both.
- NOT: Excludes documents that contain a specific term. For example, a search for "cats NOT dogs" will return documents that mention cats but not dogs.

8. What is Cluster-Based Retrieval

Ans: Cluster-based retrieval is a method used in information retrieval where similar documents or items are grouped into clusters. When a user searches for information, the search engine retrieves a representative document from each cluster, reducing the search space and improving efficiency. It's a technique to make search faster and more accurate by organizing and retrieving information from clusters of related data.

9. What do you mean by Query Language

Ans: A query language in information storage and retrieval is a specialized language or set of commands that allows users to request and retrieve specific information from a database or information system. It provides a structured way for users to express their information needs. Query languages are designed to interact with databases, search engines, or information repositories, and they can be used to filter, search, and manipulate data based on user-defined criteria.

10. Explain all types of queries in short

Ans: There are several types of queries used in information retrieval and database systems. Here's a brief explanation of some common types:

1. Boolean Queries: These queries use Boolean operators (AND, OR, NOT) to combine keywords, allowing users to create precise queries by specifying logical relationships between terms.

2. Keyword Queries: These are basic queries where users enter one or more keywords to search for documents or records containing those exact terms.

3. Phrase Queries: Users specify a sequence of words enclosed in quotation marks, and the system retrieves documents containing that exact phrase.

4. Wildcard Queries: Wildcards (e.g., *, ?) are used to match parts of words, enabling users to find variations of a term, like "comput*" to find "computer" and "computing."

5. Fuzzy Queries: Fuzzy search accounts for spelling mistakes or variations, returning results with similar terms to the query, useful when exact spelling is uncertain.

6. Proximity Queries: These queries find documents where specified keywords or phrases appear within a certain proximity of each other, indicating a relationship.

7. Range Queries: Used to find records within a specific range of values (e.g., dates, numbers) such as "date between 2022-01-01 and 2022-12-31."

8. Full-Text Queries: Searches for documents or records based on the entire text content, making them useful for more extensive text-based searches.

9. Structured Queries: Typically used with databases, these queries follow a structured format (e.g., SQL) to retrieve data based on specific criteria, often involving joins, sorting, and filtering.

10.Natural Language Queries: Users can enter queries in everyday language, and the system attempts to understand the user's intent and retrieve relevant information.

11. Explain different Models in IR

Ans: Information Retrieval (IR) models are used to represent and rank documents based on their relevance to a user's query. Here's a brief explanation of different IR models:

1. Boolean Model: In this model, documents are represented as sets of terms, and queries are expressed using Boolean operators (AND, OR, NOT). It's a precise but limited model, mainly used in database search systems.

2. Vector Space Model (VSM): Documents and queries are represented as vectors in a multi-

dimensional space, with each dimension corresponding to a unique term. The cosine similarity between vectors is used to rank documents by relevance to a query.

3. Probabilistic Model: These models, such as the Okapi BM25, use probabilistic approaches to estimate the likelihood of a document being relevant to a query. They take into account term frequency and document length to rank documents.

4. Language Model: Language models treat documents and queries as probability distributions over terms. They estimate the likelihood of generating a query from a document and use this probability to rank documents.

5. Term Frequency-Inverse Document Frequency (TF-IDF): TF-IDF assigns weights to terms based on their frequency in a document and their inverse frequency in the document collection. It helps identify the importance of terms in documents.

6. Latent Semantic Indexing (LSI): LSI applies singular value decomposition to reduce the dimensionality of the term-document matrix, capturing latent semantic relationships between terms and documents to improve retrieval.

7. Lemur/Indri Model: This is a retrieval model used in the Lemur toolkit and the Indri search engine. It combines elements of language modeling and probability theory to estimate document relevance.

8. Divergence from Randomness (DFR): DFR models calculate the divergence between a term's actual distribution in a document and the expected distribution under randomness. This helps in identifying terms that are significant for retrieval.

9. Machine Learning Models: Various machine learning techniques, including neural networks, are used to learn and optimize retrieval models. Learning to Rank (LTR) approaches train models to predict document relevance based on features derived from queries and documents.

10. Neural IR Models: Recent advancements in IR involve neural network-based models, such as BERT and Transformer-based models, which have shown remarkable performance improvements in capturing the semantics and context of queries and documents.

12. Advantages of Vector Model

Ans: Advantages of the Vector Space Model (VSM) in information retrieval:

1. Flexible representation.
2. Scalability for large datasets.
3. Effective term weighting (e.g., TF-IDF).
4. Cosine similarity for ranking.
5. Support for partial matching and ranked retrieval.
6. Potential for term and query expansion.
7. Compatibility with machine learning.
8. Versatility with different data types.
9. Interpretability.
10. Proven effectiveness in various applications.

13. What is Pattern Matching

Ans: Pattern matching is the process of finding specific sequences or structures within data or text. It's used in various fields for tasks like searching for words in documents, validating data formats, and analyzing code syntax. Regular expressions are a common tool for flexible pattern matching.

14. Explain Probabilistic Model

Ans: A probabilistic model in information retrieval estimates the likelihood of a document's relevance to a query. It uses factors like term frequency, inverse document frequency, and document length to calculate document scores. One well-known example is the Okapi BM25 model, which is effective in ranking documents for search engines and other retrieval tasks. These models handle uncertainty and variability in document content and user queries.

UNIT 3 Evaluation & Visualization of Information

1. Define precision, recall, MRR and F-score.

Ans: 1. Precision: Precision measures the proportion of relevant items among the retrieved items. It is calculated as (true positives) / (true positives + false positives) and indicates how accurate the retrieved results are.

2. Recall: Recall measures the proportion of relevant items that were successfully retrieved. It is calculated as (true positives) / (true positives + false negatives) and indicates how well the system retrieves all relevant items.

3. Mean Reciprocal Rank (MRR): MRR is a metric used in information retrieval to evaluate the effectiveness of search engines. It calculates the average of the reciprocal ranks of the first relevant item for each query, providing a measure of how quickly relevant results are found.

4. F-score: The F-score is a combination of precision and recall, providing a balanced measure of a system's performance. It is calculated as 2 * (precision * recall) / (precision + recall) and is useful when there's a need to balance between precision and recall in an evaluation.

2. What are user oriented measures?

Ans: User-oriented measures in information retrieval assess the quality of search results from the perspective of the user. They include metrics like precision, recall, click-through rate, and user satisfaction surveys to ensure that the retrieved information meets user needs and expectations, leading to a more user-friendly and effective search experience.

3. What is feature map?

Ans: In Information Storage and Retrieval (ISR), a feature map typically refers to a representation of a document or data in a high-dimensional space where each dimension corresponds to a specific feature or attribute. These features can be derived from the content of the document, such as the frequency of words or the presence of certain terms. Feature maps are used to model and compare documents for retrieval purposes, helping in the process of ranking and selecting relevant documents based on their feature representations.

4. Explain Query Specification

Ans: Query specification in Information Storage and Retrieval (ISR) is the process of defining the criteria or requirements for retrieving information from a database or document collection. It involves specifying the keywords, attributes, or other parameters that the user or system will use to search for relevant documents or data. Query specification is a critical step in the retrieval process and determines the success of finding the desired information.

5. How Interface is support for search process?

Ans: Interfaces support the search process by providing tools for query input, suggestions, result presentation, sorting, filtering, and user feedback. They enhance the user's ability to find relevant information efficiently and effectively.

Unit 4:Distributed and Multimedia IR

1. What is Distributed IR

Ans: Distributed Information Retrieval (Distributed IR) is a field of information retrieval where search and retrieval tasks are distributed across multiple computers or nodes in a network to improve performance and scalability. It involves techniques for efficiently searching and retrieving information from distributed data sources.

2. What is the need for Distributed IR

Ans: The need for Distributed Information Retrieval (Distributed IR) arises from the growing volume of data and the need for efficient and scalable ways to search and retrieve information from distributed data sources. It allows for faster and more reliable information retrieval by distributing the search process across multiple nodes, improving performance, and handling large-scale data.

3. What is Collection Partitioning

Ans: Collection partitioning is the process of dividing a large information collection or database into smaller, more manageable segments. It's often used in distributed information retrieval systems to improve efficiency and distribute the workload among multiple nodes or servers for faster and more effective searching and retrieval.

4. What is Source Selection

Ans: Source selection in the context of information retrieval and distributed systems refers to the process of choosing the most relevant data sources or repositories to search for information based on the user's query or information needs. It involves selecting the appropriate data sources, databases, or servers to retrieve information from, optimizing the retrieval process, and ensuring the retrieval system's efficiency and accuracy.

5. Query Processing in DIR

Ans: Query processing in Distributed Information Retrieval (DIR) involves the steps taken to handle user queries that are distributed across multiple data sources or servers. It includes query decomposition, source selection, query routing, and result merging. The goal is to efficiently process user queries and retrieve relevant information from distributed sources to provide a unified response to the user**.**

6. Issue in Distributed IR

1.Consistency and Relevance
2. Scalability
3. Heterogeneity
4. Network Latency
5. Fault Tolerance
6. Query Routing
7. Load Balancing
8. Security and Privacy
9. Result Merging
10. Resource Discovery

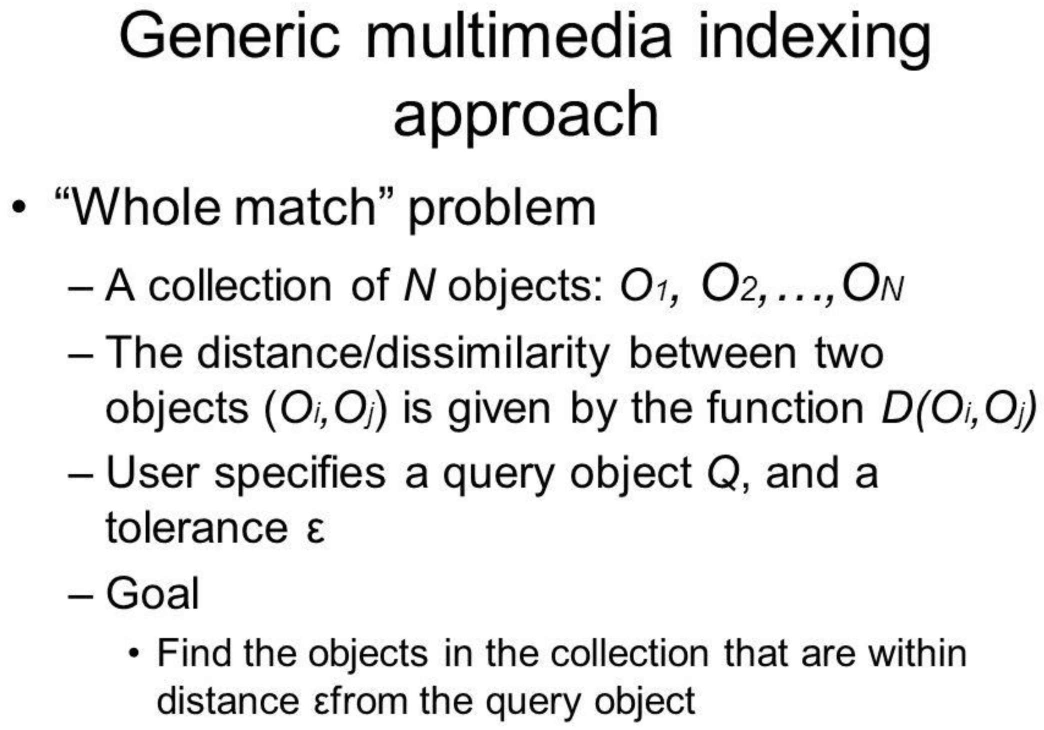
Distributed IR systems face challenges in scalability, consistency, interoperability, evaluation, and specific issues such as indexing, ranking, caching, fault tolerance, latency, staleness, and bias.

7. What is Data Modelling.

Data modelling is the process of creating a structured representation of data to define its structure, relationships, and constraints in a database or information system. It involves designing a blueprint that describes how data elements are organized and how they relate to one another, providing a foundation for efficient data storage, retrieval, and analysis. Data modelling helps ensure data accuracy, consistency, and the ability to meet specific information requirements.

In an Information Retrieval System (IRS), data modeling is limited to the classification of objects. The process of data modeling typically involves several steps, including: Requirements gathering, Conceptual design, Logical design, Physical design, Implementation

8. Explain GEMINI Algorithms.



Generic multimedia indexing approach

- "Whole match" problem
  - A collection of *N* objects: $O_1, O_2, \ldots, O_N$
  - The distance/dissimilarity between two objects (O_i, O_j) is given by the function $D(O_i, O_j)$
  - User specifies a query object Q, and a tolerance ε
  - Goal
    - Find the objects in the collection that are within distance εfrom the query object

9. Multimedia data support in commercial DBMS.

1.Varchar2-4000 bytes

2.RAW and LONG RAW - used to store graphics, sounds ,unstructured data

3.LOB- used to store Large unstructured data objects up to 4GB in size.

4.BLOBs- used to store Unstructured Binary Large Objects.

5.CLOBs- used to store Character Large Object data.

10. What is MULTOS.

- MULTOS-Multimedia Office Server.
- It is a multimedia doc server with advanced doc retrieval capabilities.
- It is based on
  - Client/Server architecture.
  - Three types of doc servers are supported
    - Current server
    - Dynamic server
    - Archive server

These servers differ in storage capacity and doc retrieval speed.

- MULTOS data model allows grouping of documents into classes of docs having similar content and structure.
- Each document is described by a
  - logical structure,
  - layout structure,
  - conceptual structure

Unit 5: Web Searching

1. Give characteristics of search engine

Crawling and Indexing: They discover and store web content.

Relevance Ranking: They determine the order of search results.

User Interface: They provide a search interface for users.

Structured Data: They offer organized information in results.

Spam Detection: They combat web spam to maintain quality.

Personalization: Results are tailored to user preferences.

Multimedia Support: They handle various media types.

Global Accessibility: Available worldwide.

Mobile-Friendly: Optimized for mobile devices.

Ad Integration: Include advertising for revenue.

Continuous Improvement: Algorithms evolve over time.

Privacy and Security: Some prioritize user privacy.

2. Explain Ranking of documents.

Document ranking is the process of ordering a set of documents according to their relevance to a given query. It is a fundamental task in information retrieval (IR) and is used in a wide range of applications, such as search engines, recommender systems, and question answering systems.

There are many different ways to rank documents. Some common approaches include:

Term frequency-inverse document frequency (TF-IDF): TF-IDF is a simple but effective ranking algorithm that assigns a weight to each term in a document based on its frequency in the document and its rarity in the document collection. The weights are then used to calculate a similarity score between the document and the query. Documents with higher similarity scores are ranked higher.

Vector space model (VSM): VSM represents documents and queries as vectors in a high-dimensional space. The similarity between a document and a query is then calculated using a vector similarity metric, such as cosine similarity. Documents with higher similarity scores are ranked higher.

Learning to rank (LTR): LTR algorithms use machine learning to learn how to rank documents effectively. LTR algorithms are trained on a set of labeled examples, where each example consists of a query and a set of ranked documents. The LTR algorithm learns to predict the relevance of each document to the query. Once the LTR algorithm is trained, it can be used to rank new documents for new queries.

The choice of ranking algorithm depends on a number of factors, such as the type of documents being ranked, the type of queries being handled, and the desired performance characteristics.

Here are some of the key considerations in document ranking:

Relevance: The ranking algorithm should be able to rank documents in order of their relevance to the query. This means that the most relevant documents should be ranked at the top of the list.

Precision: Precision is the fraction of retrieved documents that are relevant to the query. The ranking algorithm should strive to achieve high precision, so that users are not presented with irrelevant results.

Recall: Recall is the fraction of relevant documents in the collection that are retrieved. The ranking algorithm should strive to achieve high recall, so that users can find all of the relevant information they need.

Efficiency: The ranking algorithm should be efficient enough to handle large collections of documents and queries.

It presents retrieved documents in an order of their estimated degrees of relevance to query.

3. What is hyperlink

a hyperlink (or link) is an item like a word or button that points to another location. When you click on a link, the link will take you to the target of the link, which may be a webpage, document or other online content. Websites use hyperlinks as a way to navigate online content.

4. What is Web scraping

Web scraping is an automatic method to obtain large amounts of data from websites. Most of this data is unstructured data in an HTML format which is then converted into structured data in a spreadsheet or a database so that it can be used in various applications. There are many different ways to perform web scraping to obtain data from websites. These include using online services, particular API's or even creating your code for web scraping from scratch. Many large websites, like Google, Twitter, Facebook, Stack Overflow, etc.

5. What is Beautiful Soup and HTML parsing

Beautiful Soup is a Python package for parsing HTML and XML documents. It creates a parse tree for parsed web pages based on specific criteria that can be used to extract, navigate, search, and modify data from HTML, which is mostly used for web scraping.

HTML parsing is the process of taking raw HTML code, reading it, and generating a DOM tree object structure from it.

Unit 6:Advanced Information Retrieval

1. **What are the Challenges in XML Retrieval.**

   Determining the unit of retrieval: In traditional IR, the unit of retrieval is typically the entire document. However, in XML retrieval, it is possible to retrieve individual elements or subelements of a document. This can be useful for users who are looking for specific information within a document. However, it can also be challenging to determine which elements or subelements are most relevant to a given query.

   Handling nested elements: XML documents can contain nested elements. This means that one element can contain other elements. This can make it difficult to index and rank XML documents effectively. For example, if a query matches an element that is nested within another element, should the entire document be retrieved or just the nested element?

   Handling schema variations: XML documents can be created using different schemas. This means that the elements and attributes in an XML document can vary from document to document. This can make it difficult to develop a general-purpose XML retrieval system.

   Handling user queries: Users may not be familiar with the schema of an XML document. This can make it difficult for them to formulate effective queries. For example, a user may know the name of a specific element, but they may not know the name of the parent element.

2. **Difference between Text centric and Data-Centric**

   Data-centric XML: used for messaging between enterprise applications

   ■Mainly a recasting of relational data

   ■ Numerical and non-text data dominate

   Mostly stored in the databases

   Text-centric XML: used for annotating content Rich in text

   ■Demands good integration of text retrieval functionality

   ■ Queries are user information needs. E.g., give me the Section (element) of the document that tells me how to change a brake light

3. **What is Content-based filtering.**

   Content-based filtering is a recommender system that predicts what a user may like based on their activity. It uses keywords and attributes from a database to match with a user profile.

4. **Explain Semantic Web**

   The Semantic Web is a vision about an extension of the existing World Wide Web, which provides software programs with machine-interpretable metadata of the published information and data. In other words, we add further data descriptors to otherwise existing content and data on the Web.

5. **Explain term Ontology**

   Ontology, at its simplest, is the study of existence. But it is much more than that, too. Ontology is also the study of how we determine if things exist or not, as well as the classification of existence. It attempts to take things that are abstract and establish that they are, in fact, real.

6. **What is Vector Space Model**

   The Vector Space Model is an algebraic model used for Information Retrieval. It represent natural language document in a formal manner by the use of vectors in a multi-dimensional space, and allows decisions to be made as to which documents are similar to each other and to the queries fired.

7. **What is Recommendation System.**

   Recommender systems, closely related to information retrieval systems, however work without a query. Instead, the recommender system attempts to identify the most relevant piece of information solely based on an implicitly expressed information need and intent—i.e., the user profile.

8. **Reason for using RS**

Recommender systems are highly useful as they help users discover products and services they might otherwise have not found on their own.

Recommender systems are trained to understand the preferences, previous decisions, and characteristics of people and products using data gathered about their interactions. These include impressions, clicks, likes, and purchases. Because of their capability to predict consumer interests and desires on a highly personalized level, recommender systems are a favorite with content and product providers. They can drive consumers to just about any product or service that interests them, from books to videos to health classes to clothing.

9. **What is Collaborative Filtering.**

   Collaborative filtering is also known as social filtering. Collaborative filtering uses algorithms to filter data from user reviews to make personalized recommendations for users with similar preferences. Collaborative filtering is also used to select content and advertising for individuals on social media.

10. **What is Item-based Collaborative Filtering.**

    Item-based collaborative filtering in information retrieval is a method of recommending items to users based on the items that other users have rated or interacted with. It works by first creating a similarity matrix that captures the similarity between all pairs of items. This similarity matrix can be created using a variety of techniques, such as cosine similarity or Pearson correlation.

11. **What is User-based Collaborative Filtering.**

    User-based collaborative filtering is a recommendation algorithm that works by finding other users with similar preferences to the active user and then recommending items that those users have liked. It is a type of collaborative filtering, which is a machine learning technique that uses the preferences of a group of users to make predictions about the preferences of an individual user.

12. **Advantage and Drawback of Content based filtering**

    Advantages

    - The model doesn't need any data about other users, since the recommendations are specific to this user. This makes it easier to scale to a large number of users.
    - The model can capture the specific interests of a user, and can recommend niche items that very few other users are interested in.

Disadvantages

- Since the feature representation of the items are hand-engineered to some extent, this technique requires a lot of domain knowledge. Therefore, the model can only be as good as the hand-engineered features.
- The model can only make recommendations based on existing interests of the user. In other words, the model has limited ability to expand on the users' existing interests.