1. What is Conflation Algorithm? What Is the Need for Conflation?

A **conflation algorithm** is used to reduce different word forms to a common base or root form. This is often done in **information retrieval (IR)** systems to improve search efficiency by grouping words with similar meanings.

Need for Conflation:

- **Reduces Redundancy**: By grouping similar words, it reduces redundant data and storage requirements.
- **Improves Recall**: Users are able to retrieve relevant documents even if they search with variations of the words contained in the documents.
- **Increases Efficiency**: Streamlines the search process by focusing on core terms rather than specific word forms.

2. What is Luhn's Idea?

Luhn's idea, formulated by **Hans Peter Luhn**, is based on the **frequency of words** in a document. He proposed that terms with certain frequencies are likely to be more meaningful and relevant to a document's content.

- **Upper Bound**: Extremely frequent words (e.g., "and," "the") are often too common and add little value to document relevance.
- **Lower Bound**: Rare words might be too specific or obscure to contribute meaningfully.

By setting an upper and lower threshold, Luhn's idea helps in identifying keywords that are more representative of a document's subject matter.

---

3. What are Different Methods of Stemming?

Stemming is the process of reducing words to their base or root form, mainly by removing suffixes. Common stemming methods include:

- **Porter's Algorithm**: A popular rule-based method that uses several stages to strip suffixes.
- **Lovins Stemmer**: Focuses on removing the longest suffix found in a word.
- **Paice/Husk Stemmer**: An iterative, rule-based approach for suffix removal.
- **Snowball Stemmer**: An improved version of Porter's stemmer, also rule-based, with better language support.

4. What is Diff Algorithm for Suffix Stripping?

A **diff algorithm** is typically used for comparing text by finding differences between two sequences. In the context of suffix stripping, it could compare words to their possible stems to identify differences, mainly by stripping suffixes based on specific rules.

---

5. What is Major in Porter's Algorithm?

In Porter's Algorithm, **"major"** refers to key steps or rules applied to remove specific suffixes in stages. Each stage deals with a class of suffixes (e.g., "-ing," "-ed," "-ly") and modifies the word form by removing these endings in a systematic manner to arrive at the root word.

6. What is Document Representative?

A **document representative** is a summarized version of a document's content that captures essential information. It typically includes important keywords, their frequencies, occurrences, and positions within the document.

7. What are the Contents/Formats of Document Representative?

The common contents or formats of a document representative are:

- **Keywords** (conflated or stemmed terms)
- **Frequency** (number of times a keyword appears)
- **Occurrences** (total number of occurrences in the document)
- **Position** (positions of the keywords within the text)

---

8. How Much % Does the Document Reduce After Conflation?

Conflation can reduce document size by approximately **40-60%**. This reduction depends on factors like the language, document type, and specific conflation methods used.

9. What is the Use of Finding Similarity?

Finding similarity between documents or between a query and documents helps in:

- **Improving Relevance**: By retrieving documents similar to the query or relevant documents.
- **Content Clustering**: Grouping similar documents for organization or summarization.

- **Recommendation Systems**: Suggesting documents or products that align with user interests.

10. What are Different Techniques Needed for Finding Similarity?

Common similarity measurement techniques include:

- **Cosine Similarity**: Measures the cosine of the angle between two document vectors.
- **Jaccard Similarity**: Based on the intersection over union of terms in two sets.
- **Euclidean Distance**: Measures the straight-line distance between document vectors.
- **TF-IDF**: Weighs term frequency against its inverse document frequency to highlight relevant terms.

11. What is the Rule for Removing "ing" in Porter's Algorithm?

In Porter's Algorithm, the rule for removing **"-ing"** applies if the word ends with "-ing" and meets certain criteria (e.g., having a vowel before the ending). This rule helps in reducing words like "running" or "playing" to "run" and "play."

---

13. Different Stages of Conflation

Conflation generally involves the following stages:

1. **Preprocessing**: Text normalization, case-folding, and tokenization.
2. **Stemming/Lemmatization**: Reducing words to their base or root forms.
3. **Filtering Stopwords**: Removing common but uninformative words.
4. **Representation**: Creating a structured form of the text, such as keywords and their frequencies.

Calculate Precision and Recall for the Following System

Given:

- Total records in the database: 34
- Retrieved records: 50
- Relevant records among retrieved: 30
- Irrelevant records among retrieved: 20

$$\text{Precision} = \frac{\text{Relevant Retrieved Documents}}{\text{Total Retrieved Documents}}$$
$$= \frac{30}{50}$$
$$= 0.6 \text{ or } 60\%$$

$$\text{Recall} = \frac{\text{Relevant Retrieved Documents}}{\text{Total Relevant Documents}}$$
Assuming all 34 records are relevant, Recall $= \frac{30}{34} \approx 0.88$ or 88%

---

14. How to Measure Performance of IR System

The performance of an IR system can be measured using:

- **Precision and Recall**: Measures relevance and completeness.
- **F-measure**: Harmonic mean of precision and recall.
- **Mean Average Precision (MAP)**: Average precision across all queries.
- **Normalized Discounted Cumulative Gain (NDCG)**: Considers position of relevant documents in ranked results.

15. Alternative Measures for Evaluation of IR System

Alternative measures include:

- **Mean Reciprocal Rank (MRR)**: Measures ranking quality based on the position of the first relevant document.
- **ROC-AUC (Receiver Operating Characteristic - Area Under Curve)**: Measures trade-off between sensitivity and specificity.
- **Error Rate**: Number of misclassified results.

---

16. Upper Bound and Lower Bound in Luhn's Idea

In Luhn's frequency-based idea:

- **Upper Bound**: A threshold where very frequent terms are disregarded as they're too common to be useful (e.g., "the," "is").
- **Lower Bound**: A threshold below which terms are too rare or specific to carry general meaning. Only terms within the bounds are likely to be relevant.

Assignment 2

---

20. What are the Differences Between Clustering and Classification?

Feature	Clustering	Classification
Purpose	Groups data without predefined labels	Assigns data to predefined labels
Output	Unlabeled groups (clusters)	Labeled categories
Supervision	Unsupervised	Supervised
Applications	Document grouping, topic modeling	Spam detection, sentiment analysis

17. What is Clustering?

- **Clustering** is an unsupervised learning technique that groups similar data points together based on defined criteria. In Information Retrieval (IR), clustering helps organize documents or data into meaningful groups, making it easier to analyze.

18. What is the Need of a Centroid?

- A **centroid** represents the center of a cluster in some clustering algorithms (like K-means). It helps define the cluster by serving as an average point, making it easier to measure the distance between data points and the cluster.

19. What are the Differences Between Clustering and Classification?

- **Clustering**: Unsupervised; groups data without predefined labels.
- **Classification**: Supervised; assigns predefined labels to data based on training.
- Clustering finds inherent structure, while classification predicts labels based on known patterns.

20. What are the Different Methods of Clustering? Which Method Have You Used?

- **Methods**:
    - **K-means Clustering**: Partitions data into K clusters based on centroids.
    - **Hierarchical Clustering**: Builds a hierarchy of clusters using dendrograms.
    - **Density-Based Clustering**: Forms clusters based on dense regions of data points.
    - **Agglomerative/Divisive Clustering**: Bottom-up/top-down approaches for hierarchical clustering.
- **Method Used**: Specify if you've used a particular clustering method, e.g., **K-means** is commonly used for its simplicity and efficiency.

21. What are Different Similarity Functions/Measures?

- **Euclidean Distance**: Measures straight-line distance between points.
- **Cosine Similarity**: Measures angle between vectors, useful for text data.
- **Jaccard Similarity**: Measures overlap between sets.
- **Manhattan Distance**: Sum of absolute differences, useful in grid-based spaces.
- **Pearson Correlation**: Measures linear correlation between data points.

22. Write an Example of Clustering.

- Example: **Document Clustering** groups similar documents (like news articles) based on shared topics. A clustering algorithm could group articles into clusters like sports, politics, and technology.

23. What is Cluster Hypothesis?

- The **Cluster Hypothesis** states that documents in the same cluster tend to be relevant to the same queries, so grouping them together enhances retrieval effectiveness.

24. What is a Dendrogram?

- A **dendrogram** is a tree-like diagram used in hierarchical clustering to represent the merging or splitting of clusters. It helps visualize the relationships and hierarchy among clusters.

25. What are Different Search Strategies?

- **Linear Search**: Checks each item one by one.
- **Binary Search**: Efficient search in sorted data by dividing in halves.

- **Depth-First Search (DFS)**: Explores as deep as possible before backtracking.
- **Breadth-First Search (BFS)**: Explores all neighbors level-by-level.
- **Best-First Search**: Uses heuristic information to find an optimal path.

26. What is the Difference Between Single Pass Algorithm and Single Link Algorithm?

- **Single Pass Algorithm**: Processes each item once and assigns it to a cluster. Often used for quick, non-hierarchical clustering.
- **Single Link Algorithm**: In hierarchical clustering, merges clusters based on the closest pair of points, suitable for building dendrograms.

27. What is Measure of Association?

- **Measure of Association** quantifies the strength or similarity between two items. In clustering, it's used to determine how closely related data points are, often through similarity or distance measures.

28. What are the Different Methods of Classification?

- **Decision Trees**: Splits data based on attribute values to classify items.
- **Naive Bayes**: Uses probabilities for classification based on Bayes' theorem.
- **K-Nearest Neighbors (KNN)**: Classifies based on the majority class of nearest neighbors.
- **Support Vector Machines (SVM)**: Finds the optimal boundary for classification.
- **Neural Networks**: Uses layers of nodes to classify complex patterns.

29. What is the Use of Clustering in IR?

- Clustering in IR helps **organize documents by topic**, enables **relevant document grouping**, enhances search results, and improves retrieval efficiency by grouping similar items.

30. What is Minimum Spanning Tree? Where is it Used?

- A **Minimum Spanning Tree (MST)** is a subset of edges in a weighted graph that connects all vertices with the minimum possible total edge weight. It's used in clustering, network design, and pathfinding tasks to minimize distances or costs.

32. What is Indexing?

- **Indexing** is the process of creating a structured representation of data to make retrieval faster and more efficient. In Information Retrieval (IR), it involves organizing data (such as words in documents) for quick search access.

33. What is the Use of Indexing?

- Indexing is used to **speed up search operations** by reducing the amount of data the system needs to examine, allowing for faster access to relevant information.

34. What is the Need for Indexing?

- Indexing is essential to improve the efficiency of data retrieval, especially in large datasets or databases, by organizing data in a way that makes searching faster and more manageable.

35. What Data Structures Are Used in Indexing?

- Common data structures used in indexing include:
    - **Inverted Index**: Maps terms to the documents that contain them.
    - **B-trees and B+ trees**: Used for ordered indexing in databases.
    - **Hash Tables**: For quick lookup of indexed terms.
    - **Tries**: Used in full-text search engines for efficient word indexing.

36. What Are Different Types of Files?

- Types of files often mentioned in IR contexts:
    - **Text Files**: Stores raw data in text form.
    - **Binary Files**: Stores data in binary format for more efficient access.
    - **Index Files**: Specialized files that store index data for quick look-up.
    - **Inverted Files**: Store mappings of terms to documents (inverted index).
    - **Dictionary Files**: Contain unique terms or keywords.

37. How Will You Search in Indexing?

- Searching in indexing typically involves:
    - **Dictionary Lookup**: Find the term in the dictionary.
    - **Access Posting List**: Retrieve the posting list, which points to documents containing the term.
    - **Rank Results**: Rank the documents based on relevance, which can be determined by term frequency, inverse document frequency, etc.

38. What is a Dictionary?

- In IR, a **dictionary** is a collection of all unique terms in the dataset or corpus. Each term usually has a pointer to a posting list, helping in efficient search operations.

39. What is a Posting List?

- A **posting list** is a list associated with each term in an index, containing identifiers of documents in which the term appears, often with additional data like term frequency.

40. Implement the Assignment Till the End!

- This could imply completing the process of implementing an indexing or search algorithm as part of the assignment requirements. Ensure all steps, from data input to index creation and search execution, are finalized and verified for correctness.

41. What is Index Term Weighting?

- **Index term weighting** is assigning importance to terms in a document, usually based on factors like **Term Frequency (TF)** and **Inverse Document Frequency (IDF)**. This helps prioritize relevant terms for more accurate search results.

42. What is Probabilistic Index?

- A **probabilistic index** is an indexing approach in which terms and documents are assigned probabilities reflecting the likelihood that a document is relevant to a query containing a specific term.

43. Why is the Index Called an Inverted Index?

- It's called an **inverted index** because it inverts the typical document-to-term mapping, instead linking each term directly to the documents in which it appears, allowing for faster keyword-based searches.

44. Difference in All 3 IR Models

- **Boolean Model**: Uses boolean logic to match documents exactly based on query terms.
- **Vector Space Model**: Represents documents and queries as vectors; uses cosine similarity for relevance scoring.
- **Probabilistic Model**: Estimates the probability that a document is relevant based on query terms, ranking results by likelihood.

45. Explain the Probabilistic Model

- The **Probabilistic Model** in IR is based on estimating the probability that a document is relevant to a given query. It assigns weights to terms based on observed relevance, with the goal of ranking documents by their relevance probability.

Assignment 4

46. What is Crawling?

- **Crawling** is the process of automatically browsing the web to collect information from various websites. A **web crawler** (or spider) navigates from page to page via links, indexing content for search engines.

47. When Does Google Use a Web Crawler?

- Google uses web crawlers, like **Googlebot**, to discover and index new and updated web content. Crawling helps Google keep its search index current and relevant for users.

48. Difference Between Browsing and Crawling?

- **Browsing**: Done manually by a user to view content on a website.
- **Crawling**: Done automatically by a program (crawler) to systematically index and analyze content on multiple websites.

49. Use of Scheduler and Crawler

- **Scheduler**: Determines when and which pages a crawler should visit next, helping optimize the crawling process.
- **Crawler**: Collects data from web pages according to the scheduler's instructions, ensuring the web is explored systematically.

50. What is Browsing?

- Browsing is the act of **manually navigating** the web to view content on websites. It's interactive and user-driven.

51. Why Do We Use a Scheduler in a Crawler?

- A **scheduler** helps manage the crawling process by prioritizing pages, avoiding overloading websites, and ensuring efficient and timely indexing of relevant content.

52. What is a Multithreaded Downloader in a Crawler?

- A **multithreaded downloader** allows a crawler to download content from multiple pages concurrently, improving the speed and efficiency of the crawling process.

53. What is the Role of a Crawler?

- The role of a crawler is to **discover and index** web pages, gathering information to make them searchable and accessible in a search engine.

54. What is a URL?

- A **URL (Uniform Resource Locator)** is the unique address of a web page, allowing browsers and crawlers to locate and access content on the internet.

55. What is a Meta Crawler?

- A **meta crawler** is a type of crawler that aggregates results from multiple search engines, rather than directly crawling websites, to provide combined search results.

56. What is a Meta Searcher?

- A **meta searcher** sends queries to multiple search engines, combines the results, and presents them to the user, often providing more comprehensive coverage than a single search engine.

57. What Are Methods of Crawling?

- **Incremental Crawling**: Only new or updated pages are crawled.
- **Focused Crawling**: Only specific, relevant pages are crawled based on set criteria.
- **Depth-Limited Crawling**: Limits the number of links followed from a starting point.
- **Breadth-First Crawling (BFS)**: Pages are crawled level-by-level.
- **Depth-First Crawling (DFS)**: Follows links deeply before moving to the next page.

58. How Are BFS and DFS Used in a Crawler?

- **BFS (Breadth-First Search)**: Used to crawl pages at the same link depth before moving deeper. Good for wide and shallow crawling.
- **DFS (Depth-First Search)**: Crawls as deeply as possible along each link before backtracking. Suitable for deep crawling in specific areas.

Assignment 5

59. What is the Difference Between 2D Image and 3D Image?

- **2D Image**: Has only two dimensions (width and height) and represents flat images, like photos or drawings, without depth.
- **3D Image**: Includes an additional dimension (depth) and provides a sense of volume and spatial perspective, giving a more realistic representation (e.g., 3D models or stereoscopic images).

60. How is an Image Read?

- An image is read by loading it into memory using an image processing library, which interprets the file format and converts it into a matrix of pixel values that can be processed by the computer.

61. What Are the Contents of an Image?

- The contents of an image include **pixels** representing colors or grayscale intensities. These pixels can also be analyzed for features such as **edges, textures, and shapes** to identify meaningful structures within the image.

62. What Do Pixels of a Color Image Represent?

- Each pixel in a color image represents a combination of **three color channels**: Red, Green, and Blue (RGB). Each channel typically has a value between 0 and 255, indicating the intensity of that color.

63. What is Feature Extraction of an Image?

- Feature extraction is the process of identifying and isolating **key attributes** or **patterns** in an image, such as edges, textures, or shapes, that help in understanding the image's content for tasks like classification or retrieval.

64. Which Properties Are Used for Feature Extraction?

- Properties commonly used include **color, texture, shape, and edges**. Each of these can give unique insights into the visual content of the image.

65. What Are Image Contents?

- Image contents refer to the **visual elements** in an image, like **color information, shapes, edges, textures**, and spatial arrangements of pixels.

66. How to Identify Content in an Image?

- Content in an image is identified by analyzing **visual features** like color patterns, edge detection, shape analysis, and texture properties, often using algorithms that detect patterns in the pixel data.

67. How is a Color Image Represented?

- A color image is represented as a **3D matrix** with dimensions for width, height, and color channels (RGB). Each pixel in the image has three values corresponding to the intensity of Red, Green, and Blue colors.

68. What is the Size of a Color Image Matrix?

- The size of a color image matrix is **height x width x 3**, where "3" represents the RGB channels for each pixel. For example, a 100x100 color image would have a matrix of size 100 x 100 x 3.

69. (Each Pixel is Represented by 3 Values Ranging from 0 to 255)

- Yes, in a color image, each pixel is represented by **three values** (one for each color channel: Red, Green, and Blue), with each value ranging from 0 to 255.

70. How is a Pixel Represented in a Grayscale Image?

- In a grayscale image, each pixel is represented by a **single value** indicating intensity, typically ranging from 0 (black) to 255 (white), with intermediate values representing shades of gray.

71. What Other Features Can Be Extracted from a 2D Image?

- Other features that can be extracted include **shapes, edges, corners, textures, and histograms**. These help in understanding patterns and details within the image.

72. What is Content-Based Image Retrieval (CBIR)?

- CBIR is a technique used to search and retrieve images based on their **visual content**, such as color, texture, and shape, rather than relying on text-based metadata. It’s commonly used in applications where visual similarity is crucial.

73. What is Multimedia-Based Image Retrieval?

Multimedia-Based Image Retrieval (MBIR) refers to techniques used to search and retrieve images from a database based on their visual content, such as color, texture, shape, and other features. Unlike text-based search, where images are tagged with descriptive keywords, MBIR directly analyzes image properties to retrieve relevant results.

Some common types of MBIR include:

- **Content-Based Image Retrieval (CBIR)**: Retrieves images based on visual features extracted from the image itself rather than relying on metadata or keywords. For example, a user can search for images with similar color patterns, textures, or shapes.
- **Feature Extraction**: Involves extracting specific features from an image, such as color histograms, edge patterns, or spatial information, which are used to compare and match with other images in the database.
- **Relevance Feedback**: Allows users to refine search results iteratively by marking images as relevant or irrelevant, helping the system learn and improve retrieval accuracy.

Applications of MBIR include:

- Medical imaging, where specific patterns or abnormalities need to be identified.
- Digital libraries and museums for managing and searching large collections of images.
- Surveillance and security for recognizing objects or people.
- E-commerce platforms for visual product search.

---

74. How is an Image Represented in Memory?

In computer memory, an image is represented as a matrix of pixels, with each pixel storing color information. The way an image is stored in memory depends on its **format** (e.g., grayscale, RGB, etc.) and **bit depth** (e.g., 8-bit, 24-bit).

- **Grayscale Image**: Stored as a 2D array of pixels, where each pixel has a single intensity value ranging from 0 to 255 (for 8-bit images), representing shades of gray.
- **RGB Image**: Stored as a 3D array or a series of 2D arrays (channels), where each pixel has three values for Red, Green, and Blue channels, allowing a combination of colors. Each color channel value typically ranges from 0 to 255 in an 8-bit image.

    - **Example**: An RGB image of 100x100 pixels will have 3 matrices of size 100x100 for the Red, Green, and Blue channels.
- **Binary Image**: Stored as a binary matrix, where each pixel is either 0 (black) or 1 (white), commonly used in document scanning and simple image processing tasks.
- **Bit Depth**: The bit depth of an image defines how many bits are used to represent the color or intensity of each pixel.
    - **8-bit**: 256 shades (2^8 values).
    - **24-bit**: 16.7 million colors (8 bits per RGB channel).
- **Compression**: Images may be compressed in memory to save space, using formats like JPEG (lossy compression) or PNG (lossless compression), which apply algorithms to reduce file size while preserving visual quality.

Assignment 6

75. What is a web search engine?

A web search engine is an online tool that searches the web for information based on user queries. It retrieves data from billions of web pages and provides ranked results based on relevance to the query. Popular examples include Google, Bing, and Yahoo.

76. What is lexicon in Google search engine?

In Google's search engine, a *lexicon* refers to a dictionary or list of terms (keywords) and associated metadata that the search engine uses to index and retrieve documents. It includes common terms, phrases, and variations to improve the accuracy and relevance of search results.

---

77. What is the difference between a search engine and IR (Information Retrieval)?

- **Search Engine**: Primarily focused on retrieving information from the web. It uses web crawlers to index content from web pages, applying ranking algorithms to display the most relevant pages first.
- **Information Retrieval (IR)**: Refers to retrieving information from a structured or unstructured data collection, which could include documents, images, and databases, not limited to the web. IR methods are applied in both search engines and other types of databases.

78. Which algorithm is used by Google for page ranking?

Google uses the **PageRank algorithm**, developed by Larry Page and Sergey Brin, which ranks web pages based on their link structure. PageRank evaluates the importance of a page based on the number and quality of inbound links, adjusted by a damping factor to account for random browsing behavior.

---

79. What is the difference between Typical IR and Web Search as an IR?

- **Typical IR**: Focuses on smaller, controlled datasets (e.g., scientific papers, legal documents) where content is often well-structured and static.
- **Web Search IR**: Operates on a massive, constantly changing dataset with heterogeneous data (text, images, video). Web search must handle high variability, scalability, and frequent updates in web content.

80. Lucene Architecture

Apache Lucene is an open-source IR library for full-text indexing and searching. Its architecture includes:

1. **Indexing Components**: Analyze documents and convert them into an index.
2. **Searching Components**: Parse and execute queries to retrieve results from the index.
3. **Ranking Components**: Use scoring algorithms to rank the documents based on relevance.

---

81. Lucene block diagram and functioning of various blocks

In Lucene's architecture:

1. **Document Processing**: Text is tokenized and transformed into terms.
2. **Inverted Index Creation**: An inverted index is created where each term points to documents containing it.
3. **Query Parsing**: User queries are parsed and translated into search instructions.
4. **Scoring**: Documents are scored and ranked based on relevance to the query.

---

82. What is a web search engine?

A web search engine is a tool that collects, indexes, and retrieves content from the internet. It uses algorithms to rank and display relevant results for user queries based on content relevance and authority of the pages.

---

83. What is lexicon in Google search engine?

The lexicon in Google's search engine contains a list of terms or keywords from indexed pages. It aids the engine in efficiently finding relevant pages by matching query terms with indexed terms.

84. What is page ranking?

Page ranking is the process of ordering web pages in search results based on their relevance and importance. In Google's PageRank algorithm, a web page's importance is calculated based on the number and quality of links pointing to it, with adjustments from a damping factor.

---

85. Discuss challenges involved in web searching

1. **Scalability**: Indexing billions of web pages and handling large-scale queries.
2. **Relevance and Quality**: Ensuring results are accurate, relevant, and of high quality.
3. **Spam Detection**: Avoiding low-quality or manipulative sites from ranking high.
4. **User Intent Understanding**: Deciphering the true intent behind ambiguous or complex queries.
5. **Privacy Concerns**: Balancing search personalization with user privacy.
6. **Data Freshness**: Regularly updating the index to reflect new or modified web content.
7. **Multimedia Content Retrieval**: Handling non-text data like images, audio, and video.

These challenges require advanced algorithms, regular updates, and robust infrastructure to ensure efficient and effective web search.

Assignment 7

86. What is the Semantic Web and recommender system?

- **Semantic Web**: The Semantic Web is an extension of the existing World Wide Web that enables data to be shared and reused across applications,

enterprises, and communities. It adds a layer of meaning to data, allowing for more intelligent data processing. Semantic Web uses technologies like RDF, OWL, and SPARQL to enable machines to understand and process web content in a human-like way.

- **Recommender System**: A recommender system is a tool used to provide users with personalized suggestions based on their preferences and behaviors. These systems are widely used in e-commerce, streaming services, and content platforms to suggest products, movies, or articles that may be of interest.

87. Components of a Recommender System:

1. **User Profile**: Stores information about the user's preferences and past behaviors.
2. **Content Database**: Contains items (products, movies, etc.) to be recommended.
3. **Filtering Method**: Uses algorithms like collaborative filtering, content-based filtering, or hybrid approaches to generate recommendations.
4. **Recommendation Engine**: Matches users to items and generates personalized suggestions.

---

88. Designing a recommender system: What components can be used?

1. **User Data Collection**: Collects data on user interactions (clicks, purchases, ratings).
2. **Item Data Collection**: Stores detailed information on items (descriptions, features, categories).
3. **Feature Extraction**: Identifies important characteristics of items for matching purposes.
4. **Filtering Algorithms**:
    - **Collaborative Filtering**: Uses user-item interactions to make recommendations.
    - **Content-Based Filtering**: Recommends items with similar attributes to items the user previously liked.
    - **Hybrid Methods**: Combines collaborative and content-based methods.
5. **Ranking and Scoring**: Assigns scores to recommendations based on relevance.
6. **Feedback Loop**: Continuously improves recommendations using user feedback.

---

89. Difference between IR and Recommender System

- **Information Retrieval (IR)**: Aims to retrieve documents or information relevant to a user's query, focusing on the accuracy and relevance of the results. Typically relies on search algorithms and query matching.
- **Recommender System**: Predicts and suggests items of interest to users based on their preferences and behaviors. It focuses on personalization rather than direct query relevance.

---

90. What are the functions, features, and properties of a recommender system?

- **Functions**:
  - Provides personalized recommendations.
  - Enhances user experience by suggesting relevant items.
  - Helps users discover new content or products.
- **Features**:
  - Real-time recommendations based on user interactions.
  - User profile updating with new preferences.
  - Adaptability to various content types and platforms.
- **Properties**:
  - **Accuracy**: How well recommendations match user preferences.
  - **Diversity**: Provides a variety of suggestions to avoid repetition.
  - **Scalability**: Handles a large number of users and items.
  - **Robustness**: Continues to perform well despite sparse or noisy data.

91 . Draw a Typical Block Diagram of a Search Engine:

1. **Crawler**: Collects data from the web.
2. **Indexer**: Processes and organizes crawled data into an index.
3. **Query Processor**: Interprets user queries and sends them to the search engine.
4. **Ranking Algorithm**: Ranks the retrieved documents based on relevance.
5. **Results Presentation**: Displays ranked search results to the user.

---

92. Difference between Normal Computer Search and IR System

- **Normal Computer Search**:
  - Searches for exact matches in structured databases (like file names or directory structures).
  - Works with structured and organized data.
  - Usually faster and simpler.
- **Information Retrieval System**:

    - Finds relevant documents based on content, even with partial or inexact matches.
    - Works with unstructured or semi-structured data (like text).
    - Requires more complex algorithms for relevance and ranking.

General Questions

---

93. How do we evaluate an Information Retrieval (IR) system?

- Precision: Ratio of relevant documents retrieved to total retrieved documents.
- Recall: Ratio of relevant documents retrieved to total relevant documents in the database.
- F-measure: Harmonic mean of precision and recall to balance both.
- Mean Average Precision (MAP): Average of the precision values calculated after each relevant document is retrieved.
- Normalized Discounted Cumulative Gain (nDCG): Evaluates ranking quality by considering the position of relevant documents in search results.
- User satisfaction: Feedback or survey to assess user satisfaction with the IR system.

94. What is Relevance?

- Relevance refers to how well a document meets the user's query needs.
- It considers both topicality (content matching query) and situational relevance (user's specific needs and context).

95. What is Relevance Feedback?

- A technique in IR to improve query results by incorporating user feedback on the relevance of retrieved documents.
- The system adjusts the query based on user-selected relevant documents, refining search results.
- Methods:
    - Explicit Feedback: User marks relevant/non-relevant documents.
    - Implicit Feedback: System infers preferences based on user actions (e.g., time spent on documents).

96. What are different models for Information Retrieval?

- Boolean Model: Uses logical operators (AND, OR, NOT) to match documents with queries.

- Vector Space Model (VSM): Represents documents and queries as vectors in multi-dimensional space; calculates relevance via cosine similarity.
- Probabilistic Model: Predicts the probability of a document being relevant to a query.
- Latent Semantic Analysis (LSA): Uses matrix factorization to uncover relationships between words and concepts.
- Neural IR Models: Employ neural networks to learn semantic relationships in queries and documents.

97. What is an Information System?

- A structured set of components that collect, store, process, and deliver data or information.
- Functions to support decision-making, coordination, control, and analysis for an organization.
- Examples include databases, ERP systems, and content management systems.

98. What is Data Retrieval and Information Retrieval?

- Data Retrieval: Focuses on exact matching and finding structured data, like in databases (e.g., SQL queries).
- Information Retrieval: Focuses on retrieving unstructured or semi-structured data that is relevant to user queries (e.g., search engines).

99. Explain Exhaustively.

- In IR, exhaustive search refers to a complete search through all possible documents or data sources.
- It often implies covering all aspects or matching criteria to ensure comprehensive retrieval.

100. What is Distributed IR?

- Distributed Information Retrieval (IR) refers to systems that search across multiple, distributed databases or collections.
- Often involves merging and ranking results from different sources, managing latency, and dealing with heterogeneity in indexing.
- Useful in large networks or federated search environments (e.g., searching multiple library catalogs).

101. What is a Suffix Tree and Suffix Array?

- **Suffix Tree**: A compressed trie structure for all suffixes of a string; allows fast substring searches.

- **Suffix Array**: A sorted array of all suffixes of a string; more space-efficient than suffix trees but may require additional structures for fast searching.
- Used in text search and pattern matching.

102. What is a Signature File?

- An indexing method where documents are represented by compact binary signatures.
- Used for efficient searching; signatures indicate potential presence of words in documents (false positives may occur).

103. What are different algorithms for sequential search?

- Linear Search: Searches each element in a sequence until the target is found or the end is reached.
- Jump Search: Skips a fixed number of elements to reduce the number of comparisons in sorted arrays.
- Binary Search: Divides the search range by half at each step; applicable only in sorted arrays.
- Exponential Search: Finds range with potential matches using exponentially increasing steps and then applies binary search within this range.

104. What is a File in Google?

- Likely refers to Google File System (GFS), a distributed file system designed by Google.
- Key Features:
    - Stores large files with high redundancy.
    - Supports large-scale data processing applications.
    - Designed to handle frequent component failures and manage large numbers of reads/writes.

105. Benefits and Needs of Automatic Classification

- **Benefits**:
    - Saves time and resources compared to manual classification.
    - Improves consistency and accuracy in large datasets.
    - Enables efficient organization and retrieval of information.
- **Needs**:
    - Necessary for managing and searching large-scale information.
    - Supports automation in digital libraries, e-commerce, and search engines.

106. What is the Semantic Web?

- An extension of the web where information is structured to enable machines to understand relationships and meanings.
- Uses standardized ontologies and vocabularies for linking and retrieving data meaningfully.
- Facilitates data sharing and interoperability across different systems.

107. Probabilistic Model of Information Retrieval

- A model that ranks documents based on the probability of relevance to a user query.
- Uses Bayes' theorem to calculate relevance scores.
- Formula:

$$P(\text{relevance}|\text{document, query}) = \frac{P(\text{document}|\text{relevance}) \times P(\text{relevance})}{P(\text{document})}$$

108. What is Ontology?

- A structured framework representing knowledge as a set of concepts within a domain and the relationships between them.
- Used in AI, the Semantic Web, and knowledge management to organize information.

109. What is Page Ranking? How does it work? Explain Google Page Ranking

- Page ranking is a method for ranking web pages based on importance and relevance.
- **How it Works**:
  - Considers the number and quality of links to a page as indicators of its importance.
- **Google PageRank**:
  - Assigns a score to pages based on link structure.
  - Formula:

$$PR(A) = (1 - d) + d \times (PR(B)/L(B) + PR(C)/L(C) + \dots)$$

  - d is a damping factor, typically set to 0.85.

110. How to Build an IR System for 100 Documents for Quick Retrieval

- Index the documents using keywords or terms.
- Use an inverted index to map keywords to document locations.
- Implement ranking algorithms (like TF-IDF) for relevance-based results.
- Store indexes in efficient data structures (like hash tables).

111. Arranging Clothes in a Shop for Efficient Retrieval

- Use **classification** to group clothes by type (e.g., shirts, pants), size, and color.
- Create labeled sections within the shop for easy navigation.
- Use customer preferences and trends to place popular items in accessible locations.

112. Difference between Windows Search and Google's Search

- **Windows Search**: Searches locally stored files and applications on a computer.
- **Google Search**: Searches the web using complex algorithms and indexing for a vast array of content.

113. Information for a College IR System

- **Information to Include**:
  - Student records, course details, attendance, and grades.
  - New announcements, events, and resources for students.

114. Should Data Retrieval Systems Be Replaced by IR Systems?

- **Argument**:
  - IR systems are better for unstructured data and allow for approximate matching.
  - Data retrieval systems are useful for precise, structured data needs.
- **Conclusion**: Use case-dependent; IR is better for information-rich environments.

115. Difference between Structured and Unstructured Data

- **Structured Data**: Organized in a fixed format (e.g., databases, spreadsheets).
- **Unstructured Data**: Not organized in a predefined way (e.g., emails, articles).

116. Where Are They Used? Examples

- **Structured Data**: Used in relational databases (e.g., employee records).
- **Unstructured Data**: Used in document search (e.g., emails, social media).

117. Tools for Data and Text Analysis

- R, WEKA, MapReduce, Apache Spark, Lexica, and others.

118. Examples of IR and Data Retrieval

- **Information Retrieval**: Search engines, recommendation systems.
- **Data Retrieval**: Database queries, ERP systems.

119. Measuring Term Relevance in Documents (TF-IDF)

- **Term Frequency (TF)**: Measures how often a term appears in a document.
- **Inverse Document Frequency (IDF)**: Reduces the weight of common terms across documents.
- **Formula**:

$$TF \times IDF = TF(t,d) \times \log\left(\frac{N}{DF(t)}\right)$$

120. Function of Collaborative Systems

- Enables group communication, sharing of resources, and collaborative work.
- **Types**:
  - Synchronous (e.g., video conferencing).
  - Asynchronous (e.g., discussion forums).
- **Examples**: Slack, Microsoft Teams, Google Docs.

121. K-Nearest Neighbors (KNN) Algorithm

- Used for classification and regression tasks.
- Classifies a point based on the majority class among its nearest neighbors.
- Example: Classifying an email as spam or not based on similar past emails.

122. Different Web Query Languages

- SPARQL: For querying RDF data in the Semantic Web.
- XQuery: For querying XML data.
- SQL: For relational databases.

123. K-Means Algorithm for Clustering

- Divides data into kkk clusters based on minimizing the distance between points and cluster centroids.
- Steps:
  1. Initialize kkk centroids.
  2. Assign each point to the nearest centroid.
  3. Recalculate centroids and repeat until stable.

124. Architecture for Content-Based Recommender System

- **Components**:
  - Item profile (content features).
  - User profile (preferences based on past interactions).
  - Matching algorithm (compares user profile to item profile).
  - Recommends items based on similarity to user preferences.

125. Example for Content-Based and Item-Based Recommendations

- **Content-Based**: Recommending movies similar to those a user has watched.
- **Item-Based**: Recommending items that users with similar preferences have liked.