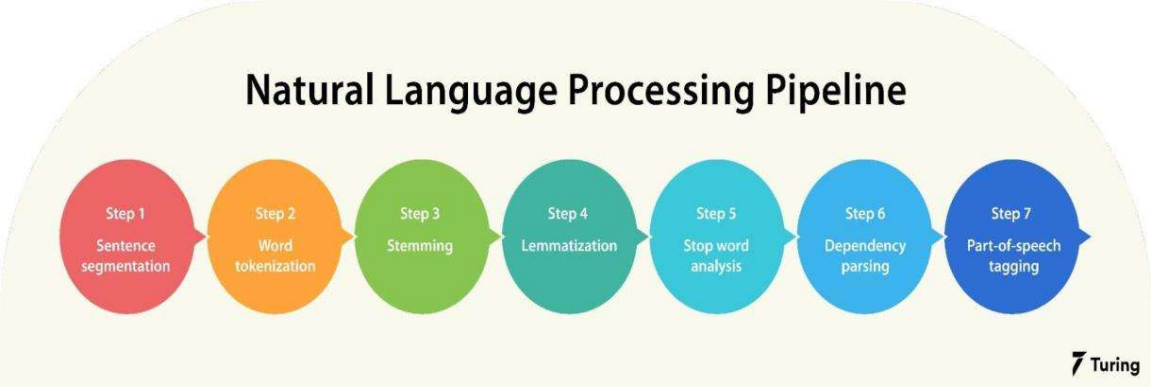
Information Retrieval for IAT1

1. **What are the components of IR? How to AI applied in IR systems?**

Information Retrieval (IR) systems are designed to help users locate relevant information within large datasets, such as documents, images, or videos.

Components of Information Retrieval (IR)



A. **Document Processing**: The first stage in IR where raw data (e.g., text documents) are processed. This includes:
B. **Tokenization**: Splitting text into words or tokens.
C. **Stemming/Lemmatization**: Reducing words to their root forms.
D. **Stop Words Removal**: Eliminating common words (e.g., "is," "the") to focus on significant terms.
E. **Indexing**: After processing, documents are indexed to make retrieval efficient. Indexing uses structures like an inverted index, where terms are mapped to document identifiers, enabling faster searches.
F. **Query Processing**: User queries are analysed and transformed to match the indexing structure. This involves:
G. **Query Tokenization and Expansion**: Expanding queries with synonyms or related terms for broader coverage.
H. **Relevance Feedback**: Adjusting queries based on initial search results to improve relevance.
I. **Ranking and Retrieval**: Algorithms rank the documents based on relevance to the query using scoring mechanisms, such as TF-IDF or BM25. Highly relevant documents are retrieved first.
J. **Evaluation**: This component checks the performance of the IR system using metrics like precision, recall, and F1-score to ensure results meet quality standards.

Application of AI in IR Systems

AI, especially machine learning (ML) and natural language processing (NLP), enhances IR in the following ways:

A. **Relevance Ranking**: ML algorithms learn from user behaviour to rank documents that are likely to be more relevant. Algorithms, such as neural networks, can process contextual information, improving ranking beyond simple keyword matching.
B. **Query Understanding and Expansion**: NLP models help in interpreting user intent more accurately, recognizing synonyms, or expanding queries with related terms.
C. **Personalization**: AI models adapt search results based on user preferences and past behaviours, offering personalized recommendations to improve user satisfaction.
D. **Semantic Search**: With advancements in AI, IR systems now use word embeddings (e.g., Word2Vec, BERT) to understand the semantic meaning of queries and documents, enabling the system to find documents based on meaning rather than exact wording.

E. **Content Summarization**: AI helps summarize large documents, making it easier for users to review results quickly without reading full texts, especially useful in fields like research and news.
F. In summary, AI in IR systems help make searches more accurate, context-aware, and personalized by leveraging advanced algorithms to understand both content and user intent better.

2. Explain Structured Model Explain Model based on Proximal Nodes

In information retrieval, models that rely on "proximal nodes" aim to improve search accuracy by considering the structural relationship and proximity between terms or nodes within a document. Let's break down these concepts:

1. Structured Model

A structured model organizes documents and queries in a structured format, often using nodes that represent different parts of the document, like sections, paragraphs, or terms. This structure allows the model to understand the context in which terms appear, providing a more nuanced approach than traditional keyword-based searches. By leveraging the document structure, it can identify the relevance of different document sections to a query, improving retrieval precision.

- **Hierarchical Structure**: Documents are represented in a tree or graph structure where each node holds information relevant to a specific aspect of the document. This could include the title, heading, subheading, or even the sentence level, helping to capture the context around query terms.
- **Term Dependencies**: Rather than considering terms independently, structured models look at the dependencies and relationships between terms within this structure. This helps understand the meaning conveyed by term placement, co-occurrences, and syntactic structure.

2. Proximal Nodes in Information Retrieval

Proximal nodes refer to nodes that are close to each other in this structured document representation, and they play a significant role in retrieval. By focusing on proximity, the model emphasizes nodes that are closer together, assuming these likely represent more related information.

- **Proximity-Based Scoring**: When a user searches for a phrase or a set of terms, the model assesses the proximity of terms within the document structure. Terms that appear close to each other or within the same node (e.g., a paragraph or a sentence) are often deemed more relevant than terms scattered across the document.
- **Semantic Context**: Proximity also helps preserve semantic context. For instance, in a tourism website, if terms like "eco-friendly" and "lodging" are proximate, the model might infer a focus on sustainable accommodations. This allows for a more sophisticated matching of search intent with document content.

3. Application in Retrieval Models

Proximal node-based models are particularly useful in structured query language retrieval, where documents are parsed into nodes, and each node is scored based on term relevance and proximity. Two approaches commonly used here include:

- **Markov Random Fields (MRF)**: MRFs use graph structures to capture dependencies between terms. They can model proximal relationships by assigning higher weights to nodes where terms appear closer, enhancing the retrieval of contextually relevant information.
- **BM25 Proximity Variants**: Some variants of BM25, a popular retrieval model, consider term proximity and structure. For instance, BM25F extends BM25 by assigning weights to different document fields (nodes) to prioritize specific content areas.

3. What are the areas of AI for information retrieval? What are the Fundamental assumptions for probabilistic principle?

AI Areas in Information Retrieval (IR)

AI makes information retrieval smarter by enhancing how systems understand and match user queries. Here are some ways AI does this:

i. **Natural Language Processing (NLP)**:

- NLP helps computers understand human language, so they get better at finding what we mean, not just the exact words we type.
- Examples: NLP powers chatbots and question-answering systems to give direct answers instead of just showing documents.

ii. **Machine Learning (ML)**:
- ML models learn from data to improve search relevance and personalize results based on what users liked before.
- Examples: Systems learn to rank documents better (e.g., showing popular articles first) and make recommendations that fit each user.

iii. **Reinforcement Learning**:
- This area allows systems to learn by trial and error, refining search results based on what users click on or spend time reading.
- Example: Search results may adjust to become more relevant over time based on user actions.

iv. **Knowledge Graphs**:
- Knowledge graphs help search systems understand relationships between things, like places or events.
- Example: Knowing that Paris is a city in France, the system can answer questions like "What's the capital of France?" with more accuracy.

v. **Probabilistic Modelling**:
- Probabilistic methods help predict which documents are likely to be relevant based on previous user data.
- Example: If a system knows that people who search "healthy recipes" often click on "vegan" or "low-carb," it will show these results higher.

vi. **Computer Vision**:
- Computer vision helps retrieve multimedia content based on what's in images or videos, not just text.
- Example: Searching for images of "beaches at sunset" retrieves relevant images without relying solely on captions.

Fundamental Assumptions of the Probabilistic Principle in IR

The probabilistic model in IR works on a few core ideas to predict if a document is relevant to a query. Here's how it's simplified:

I. **Binary Relevance**:
- Each document is either relevant or not relevant, with no in-between. This makes it easier for the system to decide what to show.

II. **Probability Ranking Principle**:
- Documents are ranked by how likely they are to be relevant. The most relevant ones appear first.

III. **Relevance Independence**:
- The relevance of one document doesn't depend on the relevance of others. Each document is considered on its own.

IV. **Term Independence**:
- Each term (word) in the query is treated independently. If you search "best pizza," it looks at "best" and "pizza" separately rather than combining them.

V. **Learning from Feedback**:
- The system gets better by learning from past searches and clicks. As users interact, it updates to give more accurate results over time.

4. **Give the functions of information retrieval system. How can you find similarity between doc and query in probabilistic principle Using Bayes' rule?**

Functions of an Information Retrieval System (IR System)

An IR system's main goal is to find and rank documents relevant to a user's query. Its key functions include:

1. **Document Representation**:
   - Organizes and indexes documents by breaking down text into words, phrases, or other features. This makes it faster and easier to search for terms.
2. **Query Processing**:
   - Analyses the user's input to understand what information they are seeking. This may include breaking down the query into keywords, correcting spelling errors, or expanding terms to match similar ones (e.g., "car" and "automobile").
3. **Matching and Similarity Measurement**:
   - Finds documents that match the query by calculating similarity between the query and documents in the index. This involves ranking the documents based on relevance.
4. **Retrieval and Ranking**:
   - Once documents are found, the system ranks them based on relevance scores so the most relevant ones appear at the top.
5. **Feedback and Refinement**:
   - Some systems use user feedback, like clicks or time spent on a page, to improve future searches and refine results. The system learns from user behaviour to enhance accuracy over time.
6. **Results Display and Presentation**:
   - Displays search results in a user-friendly way, often with document snippets or previews, making it easier for users to determine which documents meet their needs.

Finding Similarity Between Document and Query Using the Probabilistic Principle with Bayes' Rule

In the probabilistic model, we use Bayes' Rule to estimate the probability that a document is relevant to a query. Here's a breakdown of how this works:

1. **Define Relevance Probability**:
   - The goal is to find the probability $P(R|d,q)$ that a document d is relevant R given the query q.
2. **Applying Bayes' Rule**:
   - Using Bayes' Rule, we can express this probability as:
   $$P(R|d,q) = \frac{P(d|R,q) \cdot P(R|q)}{P(d|q)}$$
   - Where:
     - $P(R|q)$ is the prior probability of relevance given the query.
     - $P(d|R,q)$ is the probability of observing the document d given that it is relevant to q.
     - $P(d|q)$ is the probability of observing the document given the query.
3. **Estimating Probabilities**:
   - **Prior Probability** $P(R|q)$: Often assumed constant for all documents, representing the general probability of any document being relevant to the query.
   - **Document Likelihood** $P(d|R,q)$: Often estimated based on term frequency, which represents how likely terms in the query appear in relevant documents.
   - **Normalization** $P(d|q)$: Ensures probabilities sum to 1 but is often ignored during ranking because it is constant across all documents.
4. **Ranking Documents**:
   - Since $P(d|q)$ is constant, we only need to focus on $P(d|R,q) \cdot P(R|q)$. The documents are ranked based on the likelihood of query terms appearing in each document, given relevance.