RGPVNOTES.IN

Program : **B.E**

Subject Name: **Modern Information Retrieval**

Subject Code: **CS-7004**

Semester: **7th**

Department of Computer Science & Engineering

Subject Name: Modern Information Retrieval **Subject Code:** CS7004

UNIT – 2

Syllabus

Classic Information Retrieval Techniques: Boolean Model, Vector Model, Probabilistic Model, comparison of classical models. Introduction to Alternative Algebraic Models: Latent Semantic Indexing Model etc.

A Formal Characterization of IR Models

Definition: An information retrieval model is a quadruple [D, Q, F, R (qi, dj)] where

1. D is a set of logical views (or representations) for the documents in the collection.
2. Q is a set of logical views (or representations) for the user information needs. Such representations are called queries.
3. F is a framework for modeling document representations, queries, and their relationships.
4. R (qi, dj) is a ranking function which associates a real number with a query qi ε Q and a document dj ε D. Such ranking defines an ordering among the documents with regards to the query qi.

To build a model, we think first of representations for the documents and for the user information need. Given these representations, we then conceive the framework in which they can be modeled. This framework should also provide the intuition for constructing a ranking function. For instance, for the classic Boolean model, the framework is composed of sets of documents and the standard operations on sets. For the classic vector model, the framework is composed of a t-dimensional vector space and standard linear algebra operations on vectors. For the classic probabilistic model, the framework is composed of sets, standard operations, and bayes' theorem.

Classic Information Retrieval

In this section we briefly present the three classic models in information retrieval namely, the Boolean, the vector and the probabilistic models.

Basic Concepts

The classic model in IR involves the following:

1. Each document is described by a set of representative keywords called index terms. An index term is simply a document word whose semantics help in remembering the main theme of the documents. Thus, index terms are used to index and summarize the document contents.
2. In general, index terms are mainly nouns because nouns have meaning by themselves and thus, their semantics is easier to identify and to grasp.
3. Adjectives, adverbs, and connectives are less useful as index terms because they work mainly as complements.
4. How to decide which index terms are more important? Think about domain-specific, application- specific terms. A word that appears in all documents is completely useless. But a word that appears many times in some of the documents but only a few in others may be useful!
5. Thus we use weights for each index term of a document. This weight quantifies the importance of the index term for describing the document semantic contents.

IR Models:

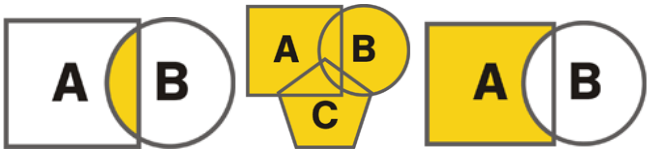
1. Boolean Model

The Boolean model is the first model of information retrieval and probably also the most criticized model. The

Department of Computer Science & Engineering

Subject Name: Modern Information Retrieval **Subject Code:** CS7004

Boolean model is the first model of information retrieval and probably also the most criticized model. The model can be explained by thinking of a query term as an unambiguous definition of a set of documents. For instance, the query term economic defines the set of all documents that are indexed with the term economical. Using the operators of George Boole's mathematical logic, query terms and their corresponding sets of documents can be combined to form new sets of documents. The Boolean model allows for the use of operators of Boolean algebra, AND, OR and NOT, for query formulation, but has one major disadvantage: a Boolean system is not able to rank the returned list of documents. In the Boolean model, a document is associated with a set of keywords. Queries are also expressions of keywords separated by AND, OR, or NOT/BUT. The retrieval function in this model treats a document as either relevant or irrelevant. In below figure, the retrieved sets are visualized by the shaded areas.



2. **Vector Space Model** Gerard Salton and his colleagues suggested a model based on Luhn's similarity criterion that has a stronger theoretical motivation (Salton and McGill 1983). They considered the index representations and the query as vectors embedded in a high dimensional Euclidean space, where each term is assigned a separate dimension. The vector space model can best be characterized by its attempt to rank documents by the similarity between the query and each document [10].In the Vector Space Model(VSM), documents and query are represent as a Vector, and the angle between the two vectors is computed using the similarity cosine function. Similarity Cosine function can be defined as: Where, , = · ·‖ = ∑ ,, ∑ , ∑ , (1) Documents and queries are represented as vectors. = ,, ,, … , , = ,, ,, … , , Vector Space Model has been introducing a term weight scheme known as if-idf weighting. These weights have a term frequency (tf) factor measuring the frequency of occurrence of the terms in the document or query texts and an inverse document frequency (idf) factor measuring the inverse of the number of documents that contain a query or document term.

3. **Probabilistic Model** Whereas Maron and Kuhns introduced ranking by the probability of relevance; it was Stephen Robertson who turned the idea into a principle. He formulated the probability ranking principle, which he attributed to William Cooper, as follows (Robertson 1977). The most important characteristic of the probabilistic model is its attempt to rank documents by their probability of relevance given a query. Documents and queries are represented by binary vectors ~d and ~q, each vector element indicating whether a document attribute or term occurs in the document or query, or not. Instead of probabilities, the probabilistic model uses odds O(R), where O(R) = P(R)/1 − P(R), R means "document is relevant" and ¯R means "document is not relevant". Probability theory can be used to compute a measure of relevance between a query and a document.
   - Simple Term Weights.
   - Non binary independent model.
   - Language model

3.1 Simple Term Weights: The use of term weights is based on the Probability Ranking Principle (PRP), which assumes that optimal effectiveness occurs when documents are ranked based on an estimate of the probability of their relevance to a query The key is to assign probabilities to components of the query and then use each of these as evidence in computing the final probability that a document is relevant to the query. The terms in the query are assigned weights which correspond to the probability that a particular term, in a match with a given query, will retrieve a relevant document. The weights for each term in the query are

combined to obtain a final measure of relevance. Most of the papers in this area incorporate probability theory and describe the validity of independence assumptions, so a brief review of probability theory is in order.

3.2 Non-Binary Independence Model: The non-binary independence model term frequency and document length, somewhat naturally, into the calculation of term weights. Once the term weights are computed, the vector space model is used to compute an inner product for obtaining a final similarity coefficient. The simple term weight approach estimates a term's weight based on whether or not the term appears in a relevant document. Instead of estimating the probability that a given term will identify a relevant document, the probability that a term which appears if times will appear in a relevant document is estimated.

3.3 Language Models: A statistical language model is a probabilistic mechanism for "generating" a piece of text. It thus defines a distribution over all the possible word sequences. The simplest language model is the unigram language model, which is essentially a word distribution. More complex language models might use more context information (e.g., word history) in predicting the next word if the speaker were to utter the words in a document, what is the likelihood they would then say the words in the query.

4. **Inference Network Model** In this model, document retrieval is modeled as an inference process in an inference network Most techniques used by IR systems can be implemented under this model. In the simplest implementation of this model, a document instantiates a term with certain strength, and the credit from multiple terms is accumulated given a query to compute the equivalent of a numeric score for the document. From an operational perspective, the strength of instantiation of a term for a document can be considered as the weight of the term in the document, and document ranking in the simplest form of this model becomes similar to ranking in the vector space model and the probabilistic models described above. The strength of instantiation of a term for a document is not defined by the model, and any formulation can be used.

Comparison of classical models

IR Models (IR mod)/ attributes (A)	Boolean Model	Vector space Model	Probabilistic Model	Latent semantic Indexing
Concept (A)	Based on set theory and Boolean algebra	Based on the concept of vectors	Based on probability ranking principle	It is an extension of vector space model
Representation (A)	Documents are represented by the index terms extracted from documents, and queries are Boolean expressions on terms.	Represented in the form of weighted-term vectors. Cosine measure is used to find the similarities	Documents and queries are represented in binary vectors	Documents are represented in the form of term-document matrix.
Information Type (A)	Docs not consider semantic information	It consider semantic information	Considers the semantic information	Considers the semantic information
Word occurrence (A)	Number of occurrence arc not	Tells about the number of occurrence	Occurrence based on the probability	Based on term-document matrix

	mentioned		relevance	
Output(A)	Exact match of the output to the query	Best match of the query	It gives best match of output	Best match of the query
Advantages(A)	Easy to implement	Simple model, weights are not in binary	Theoretical adequacy: ranks by probabilities	Synonymy and polysemy
Disadvantages(A)	Does not rank documents, retrieves too many or too few	Suffers from synonymy and polysemy. It theoretically assumes that terms are statistically independent	Binary weights ignore frequencies and independence assumption.	Not clear about similarity between words

Table 1: Comparison of Information Model

Introduction to Alternative Algebraic Models:

Latent semantic analysis (LSA) is a technique in natural language processing, in particular distributional semantics, of analyzing relationships between a set of documents and the terms they contain by producing a set of concepts related to the documents and terms. LSA assumes that words that are close in meaning will occur in similar pieces of text (the distributional hypothesis). A matrix containing word counts per paragraph (rows represent unique words and columns represent each paragraph) is constructed from a large piece of text and a mathematical technique called singular value decomposition (SVD) is used to reduce the number of rows while preserving the similarity structure among columns. Paragraphs are then compared by taking the cosine of the angle between the two vectors (or the dot product between the normalizations of the two vectors) formed by any two columns. Values close to 1 represent very similar paragraphs while values close to 0 represent very dissimilar paragraphs.

LSI is a proximity model, a mathematical technique for spatially grouping similar objects together. This is achieved by first a) identifying keyword co-occurrences that exist across a set of documents, then b) organizing them into a multi-dimensional term-by document matrix, and finally c) algebraically decomposing the matrix to reduce the matrix dimensionality. As the dimensional space is reduced, related documents draw closer to one another. The relative distances between these points in the reduced vector space have been shown to represent semantic similarity between documents, and can be used as a basis for the final stage of d) fulfilling information queries. The keywords input by a user are mapped onto the same reduced vector space, and the information retrieval systems present documents to the user that are located in the same relative neighbourhood. These retrieved resources are semantically related to the user's search criteria, even if they do not share the exact same keywords. Testing has demonstrated that these semantically related documents are also related conceptually, thereby satisfying the user's information need. This is the fundamental appeal and promise of LSI; that it circumvents the need for direct word to-word mapping between user and system, and replaces onerous, manual indexing techniques with an automated approach.

The accuracy of LSI model performance in experimental studies – up to 30% better retrieval performance over traditional lexical matching techniques (Dumais, 1991) - is impressive. At a time when electronic textual resources

are burgeoning beyond the capacity of manual indexing efforts, LSI's promise of accurate, automated content representation holds great appeal. Despite 15 years of active research into ways to improve the latent semantic indexing model, though, field applications of LSI remains limited to specialized resource collections of limited size.

Benefits of LSI

1. LSI helps overcome synonymy by increasing recall, one of the most problematic constraints of Boolean keyword queries and vector space models. Synonymy is often the cause of mismatches in the vocabulary used by the authors of documents and the users of information retrieval systems. As a result, Boolean or keyword queries often return irrelevant results and miss information that is relevant.
2. LSI is also used to perform automated document categorization. In fact, several experiments have demonstrated that there are a number of correlations between the way LSI and humans process and categorize text. Document categorization is the assignment of documents to one or more predefined categories based on their similarity to the conceptual content of the categories. LSI uses example documents to establish the conceptual basis for each category. During categorization processing, the concepts contained in the documents being categorized are compared to the concepts contained in the example items, and a category (or categories) is assigned to the documents based on the similarities between the concepts they contain and the concepts that are contained in the example documents.
3. Dynamic clustering based on the conceptual content of documents can also be accomplished using LSI. Clustering is a way to group documents based on their conceptual similarity to each other without using example documents to establish the conceptual basis for each cluster. This is very useful when dealing with an unknown collection of unstructured text.
4. Because it uses a strictly mathematical approach, LSI is inherently independent of language. This enables LSI to elicit the semantic content of information written in any language without requiring the use of auxiliary structures, such as dictionaries and thesauri. LSI can also perform cross-linguistic concept searching and example-based categorization. For example, queries can be made in one language, such as English, and conceptually similar results will be returned even if they are composed of an entirely different language or of multiple languages.
5. LSI is not restricted to working only with words. It can also process arbitrary character strings. Any object that can be expressed as text can be represented in an LSI vector space. For example, tests with MEDLINE abstracts have shown that LSI is able to effectively classify genes based on conceptual modeling of the biological information contained in the titles and abstracts of the MEDLINE citations.
6. LSI automatically adapts to new and changing terminology, and has been shown to be very tolerant of noise (i.e., misspelled words, typographical errors, unreadable characters, etc.). This is especially important for applications using text derived from Optical Character Recognition (OCR) and speech-to-text conversion. LSI also deals effectively with sparse, ambiguous, and contradictory data.
7. Text does not need to be in sentence form for LSI to be effective. It can work with lists, free-form notes, email, Web-based content, etc. As long as a collection of text contains multiple terms, LSI can be used to identify patterns in the relationships between the important terms and concepts contained in the text.

LSI has proven to be a useful solution to a number of conceptual matching problems. The technique has been shown to capture key relationship information, including causal, goal-oriented, and taxonomic information

Applications:

1. Information discovery
2. Automated document classification (eDiscovery, Government/Intelligence community, Publishing)
3. Text summarization (eDiscovery, Publishing)
4. Relationship discovery (Government, Intelligence community, Social Networking)
5. Automatic generation of link charts of individuals and organizations (Government, Intelligence community)
6. Matching technical papers and grants with reviewers
7. Online customer support
8. Understanding software source codes(Software Engineering)
9. Filtering spam (System Administration)
10. Information visualization
11. Stock returns prediction

RGPVNOTES.IN

We hope you find these notes useful.

You can get previous year question papers at https://qp.rgpvnotes.in .

If you have any queries or you want to submit your study notes please write us at rgpvnotes.in@gmail.com

LIKE & FOLLOW US ON FACEBOOK
facebook.com/rgpvnotes.in