

Faculty Development Program-2015, (CSE: Information Retrieval)

Prof. (Dr.) K.R. Chowdhary, Director SETG

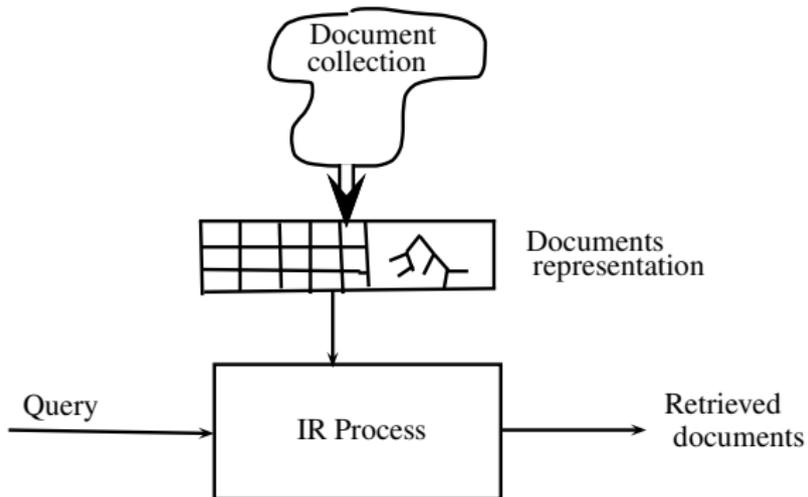
Email: kr.chowdhary@jietjodhpur.com

Webpage: <http://www.krchowdhary.com>

Jodhpur Institute of Engineering and Technology, SETG

July 14, 2015

Basic Model of IR



Search Engine

The screenshot shows a Recall search engine window. The search bar contains the text "information retrieval". Below the search bar, there are radio buttons for file types: All (selected), media, message, other, presentation, spreadsheet, and text. The search results are displayed as a list of documents. The first result is a PDF file named "0110114q_AAMAS07_0421_34d063151fd18a417db4daf1e7005279.dvi" with a size of 23% 16 KB. The second result is a PDF file named "TOIS2604-20.tex" with a size of 18% 79 KB. The third result is a PDF file named "p911.pdf" with a size of 16% 53 KB / 1 MB. The search results are displayed in a list format with a preview icon (a red triangle) next to each document. The search results are displayed in a list format with a preview icon (a red triangle) next to each document. The search results are displayed in a list format with a preview icon (a red triangle) next to each document.

Recall

Clear Search Query language: information retrieval

All media message other presentation spreadsheet text

Query results Documents 1-8 out of at least 1050 for [\(show query\)](#) [Next](#)

23% 16 KB [Preview](#) [Open](#)
0110114q_AAMAS07_0421_34d063151fd18a417db4daf1e7005279.dvi
application/pdf 2007-04-29 18:38:12 +0530 file://home/kr/home2/Intelliget-agents/AAMAS07_0421_34d063151fd18a417db4daf1e7005279.pdf
system we now distinguish retriever robots that access the ... robots and k 1 retrievers the problem description as ... are in charge of retrieving cases as new problems ... we call these robots retrievers there must be at least one retriever 1 ≤ k ≤ ... next retrieve the robot retrieves the case based on ... the field and it informs the retrieved case id ...

18% 79 KB [Preview](#) [Open](#) **TOIS2604-20.tex**
application/pdf 2009-01-23 10:08:22 +0530 file://home/kr/home2/acm/Tian on Info Sys/Improved Search Engines and Navigation preference in personal Information.pdf
information management acm trans inform syst 26 4 article ... navigation or search when retrieving their files and to ... percentage of the recent retrievals which people reported were ... percentage of total file retrievals using other navigational methods ... user both organizes and retrieves the information users are ... the same person who retrieves it later on it ... and smith i 2000 informing the design of an ...

16% 53 KB / 1 MB [Preview](#) [Open](#) **p911.pdf**
application/pdf 2012-07-09 18:02:54 +0530 file://home/kr/home2/Documents/informn-retr/docs-sigr-12/docs/p911.pdf

Result count (est.): 1050

Taxonomy of IR Models

Three classic models in IR are:

- **Boolean:** Document and query are sets of Index terms.
- **Vector Space:** Query and documents are vectors in t-dimensional space.
- **Probabilistic:** representation are based on probability theory.

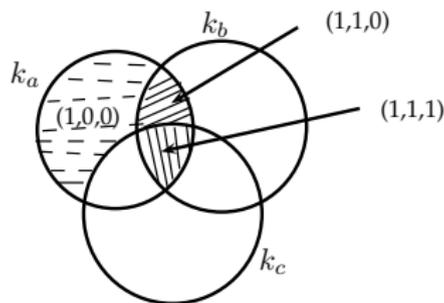
Formal Characterization of models

- IR Model : $[\mathbf{D}, \mathbf{Q}, \mathcal{F}, R(q_i, d_j)]$
- \mathbf{D} : logical view/representation of documents
- \mathbf{Q} : logical view/representation of query
- \mathcal{F} : framework for representation of queries, documents, and their relationship
- $R(q_i, d_j)$: a ranking function (a real number), $q_i \in \mathbf{Q}$, $d_j \in \mathbf{D}$

- Document is transformed to index terms
- Nouns are index terms (others less useful)
- More frequent keywords as index terms
- Index terms are assigned weights
- k_i (index term), d_j is document, then $w_{i,j} \geq 0$ is weight for pair (k_i, d_j) .
- Let $K = \{k_1, k_2, \dots, k_t\}$ is set of index terms. Weight $w_{i,j} \geq 0$ associated with each term k_i and document d_j . For $k_i \notin d_j$, $w_{i,j} = 0$.
- d_j has associated index term Vector $\vec{d}_j = (w_{1,j}, \dots, w_{t,j})$
- Let $g_i(\vec{d}_j) = w_{i,j}$, is a function that returns weight associated with each term. For the sake of simplicity, we assume that term weights in a sentence are independent. However, in a true sense they are not, say in *computer network*, the term “computer attracts the existence of ” network“, and vice-versa.

- It is based on theory of Boolean algebra, simple, intuitive.
- Consider that index terms are present/absence. $w_{i,j} \in \{0,1\}$.
- Query q 's terms are linked by *and*, *or*, *not*. q is either *CNF* or *DNF*.
- $q = k_a \wedge (k_b \vee \neg k_c)$ can be written in DNF as $\vec{q}_{dnf} = [(1,0,0) \vee (1,1,0) \vee (1,1,1)]$. Each component (e.g., $(1,1,0)$) is binary weighted

vector associated with tuple (k_a, k_b, k_c) .



- drawback: retrieval strategy is binary decision

- For $w_{i,j} \in \{0,1\}$, \vec{q}_{dnf} as query vector, let \vec{q}_{cc} be any of disjunctive components of \vec{q}_{dnf} .
- Similarity of d_j to q is:

$$sim(d_j, q) = \begin{cases} 1 & \text{if } \exists \vec{q}_{cc} | (\vec{q}_{cc} \in \vec{q}_{dnf}) \wedge (\forall k_i, g_i(\vec{d}_j) = g_i(\vec{q}_{cc})) \\ 0 & \text{otherwise.} \end{cases}$$

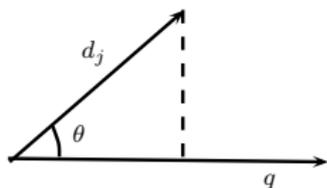
- if $sim(d_j, q) = 1$ then d_j is relevant to q , else not.
- no notion of *partial match*
- e.g., $\vec{d}_j = (0,1,0)$, so d_j includes index term k_b , but not relevant to query $q = k_a \wedge (k_b \vee \neg k_c)$.
- Index term weighting brings *vector model*.

- Considers the documents that match partially
- Non-binary weights to index terms in queries and documents
- Documents' Similarity is ordered in descending order
- $w_{i,j}$ for (k_i, d_j) is positive and non-binary.
- Let $w_{i,j}$ is weight for pair (k_i, q) . $\vec{q} = (w_{1,q}, \dots, w_{t,q})$,

and t is index term count.
Vector $\vec{d}_j = (w_{i,j}, \dots, w_{t,j})$.

- Cosine of θ adopted as $sim(d_j, q)$
- Vector model evaluates degree of similarity between document d_j and query q as a correlation between \vec{d}_j and \vec{q} .
- This correlation is θ , angle between vectors.

Vectors: \vec{d}_j, \vec{q} .



$$\begin{aligned} \text{sim}(d_j, q) &= \frac{\vec{d}_j \cdot \vec{q}}{|\vec{d}_j| \times |\vec{q}|} \\ &= \frac{\sum_{i=1}^t w_{i,j} \times w_{i,q}}{\sqrt{\sum_{i=1}^t w_{i,j}^2} \times \sqrt{\sum_{i=1}^t w_{i,q}^2}} \end{aligned}$$

- where, $|\vec{d}_j|$ and $|\vec{q}|$ are the norms of document and query vectors. The $|\vec{q}|$ does not effect ranking as it is same for all docs.
- The factor $|\vec{d}_j|$ provides normalization.
- vector model ranks the docs in order of their similarity to query, i.e., as per $\text{sim}(d_j, q)$.
- A threshold is used to reject those below that.

- Given collection set C of objects, and description of set A , classify $x \in C$ to $R(x, A)$, and $\neg R(x, A)$, here R is relation. (This is clustering) (vague !).
 - Example: C is all cars, and A is Maruti-Alto.
 - Example: C is all cancer patients, and $A = \{\text{terminal, advanced, metastatic, diagnosed, healthy}\}$. Then A divides C into five clusters.
- For $C =$ all docs, and $A =$ features of some docs, what $x \in C$ is $x \in A_i$ (for $i = 1, n$) is clustering.
 - A is documents features.
 - Term weights? it is based on two factors: 1) intra-clustering similarity, is based on term frequency (tf) of term k_i , in d_j (how well the term describes the doc.), 2) inter-cluster similarity, inverse of the freq. of k_i among documents (idf).

- Let Docs = N , the k_i term exists in n_i numbers. $freq_{i,j}$ is freq (counts) of k_i in the d_j , the normalized freq of k_i in d_j ,

$$f_{i,j} = \log \frac{freq_{i,j}}{\max_l freq_{l,j}}$$

,
where, maximum is computed over all terms in doc d_j . If $k_i \notin d_j$, then $f_{i,j} = 0$.

- Let idf is *inverse document frequency* for k_i ,

$$idf_i = \log \frac{N}{n_i}$$

- Best known weighted scheme is: $tf \times idf$

Adv: of vector:

- term weighting improves retrieval performance
- partial matching of q and d_j , allows retrieval of those not matching fully
- disadv**: index terms are assumed mutually independent

Probabilistic model

- Given d_j and q , model will find probability that d_j is relevant to q .
- Certainly, $R \subseteq D$ is relevant to q ; the R is ideal answer set
- Here, $w_{i,j} \in \{0, 1\}$, $q \subseteq \cup\{k_i\}$
- $R \subseteq D$ is set of relevant docs and \bar{R} is non-relevant.
- Let $P(R|\vec{d}_j)$ is prob. that d_j is relevant to q , and Let $P(\bar{R}|\vec{d}_j)$ is prob. that d_j is non-relevant to q
- Similarity of d_j to q ,

$$\text{sim}(d_j, q) = \frac{P(R|\vec{d}_j)}{P(\bar{R}|\vec{d}_j)}$$

- Using Bayes rule:
$$P(A|B) = \frac{P(B|A)P(A)}{P(B)},$$

$$\text{sim}(d_j, q) = \frac{P(\vec{d}_j|R)P(R)}{P(\vec{d}_j|\bar{R})P(\bar{R})}$$

where, $P(\vec{d}_j|R)$ is probability randomly selecting doc. given that it is relevant. $P(R)$ is prob. that selected doc is relevant.

- Since $P(R)$ and $P(\bar{R})$ are same for all docs.

$$\text{sim}(d_j, q) \sim \frac{P(\vec{d}_j|R)}{P(\vec{d}_j|\bar{R})}$$

- Assuming independence of index terms:

$$\text{sim}(d_j, q) \sim \frac{(\prod_{g_i(\vec{d}_j)=1} P(k_i|R)) \times (\prod_{g_i(\vec{d}_j)=0} P(\bar{k}_i|R))}{(\prod_{g_i(\vec{d}_j)=1} P(k_i|\bar{R})) \times (\prod_{g_i(\vec{d}_j)=0} P(\bar{k}_i|\bar{R}))}$$

- where $P(k_i|R)$ is prob. that k_i exists in a doc randomly selected from R , and \bar{k}_i means does not exist.