情報検索システム特論

Advanced Information Retrieval Systems

第3回 Lecture #3

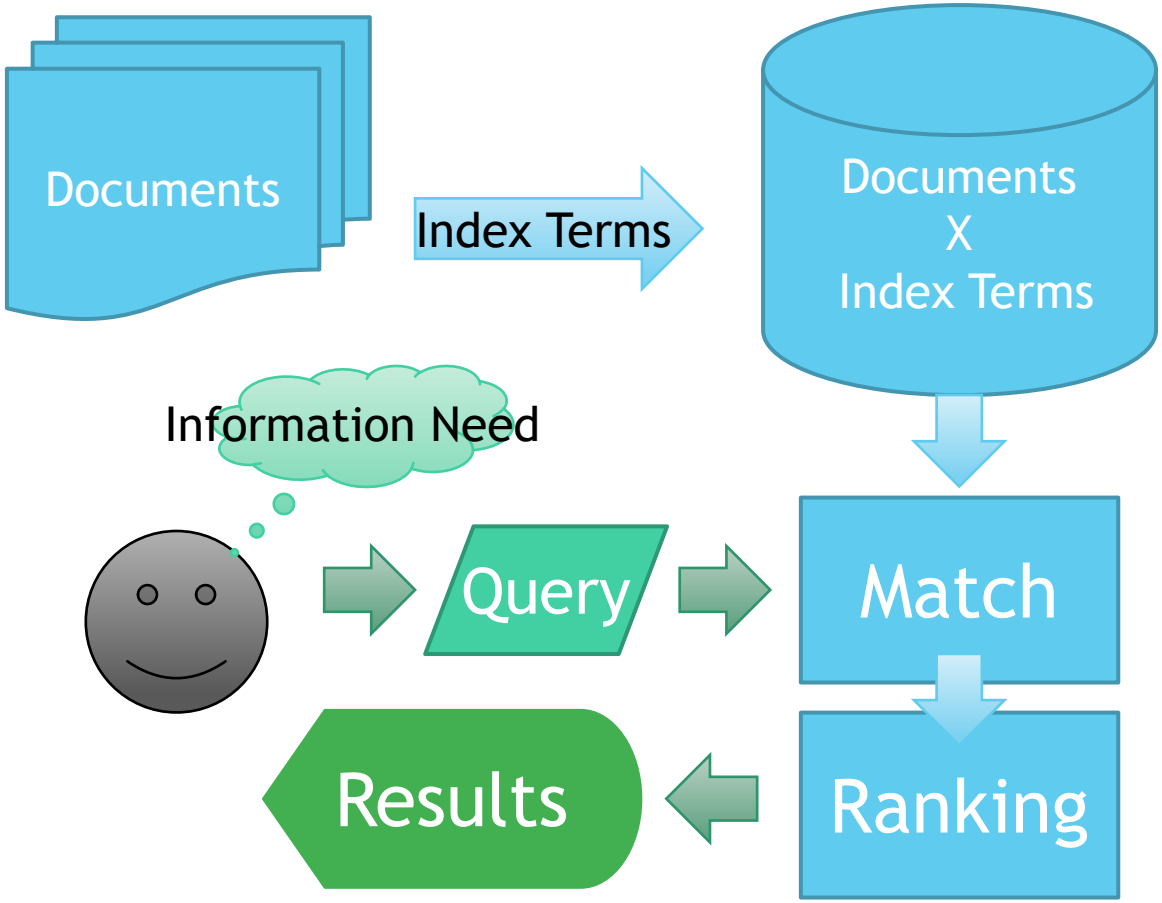
2023-04-24

湯川 高志 Takashi Yukawa

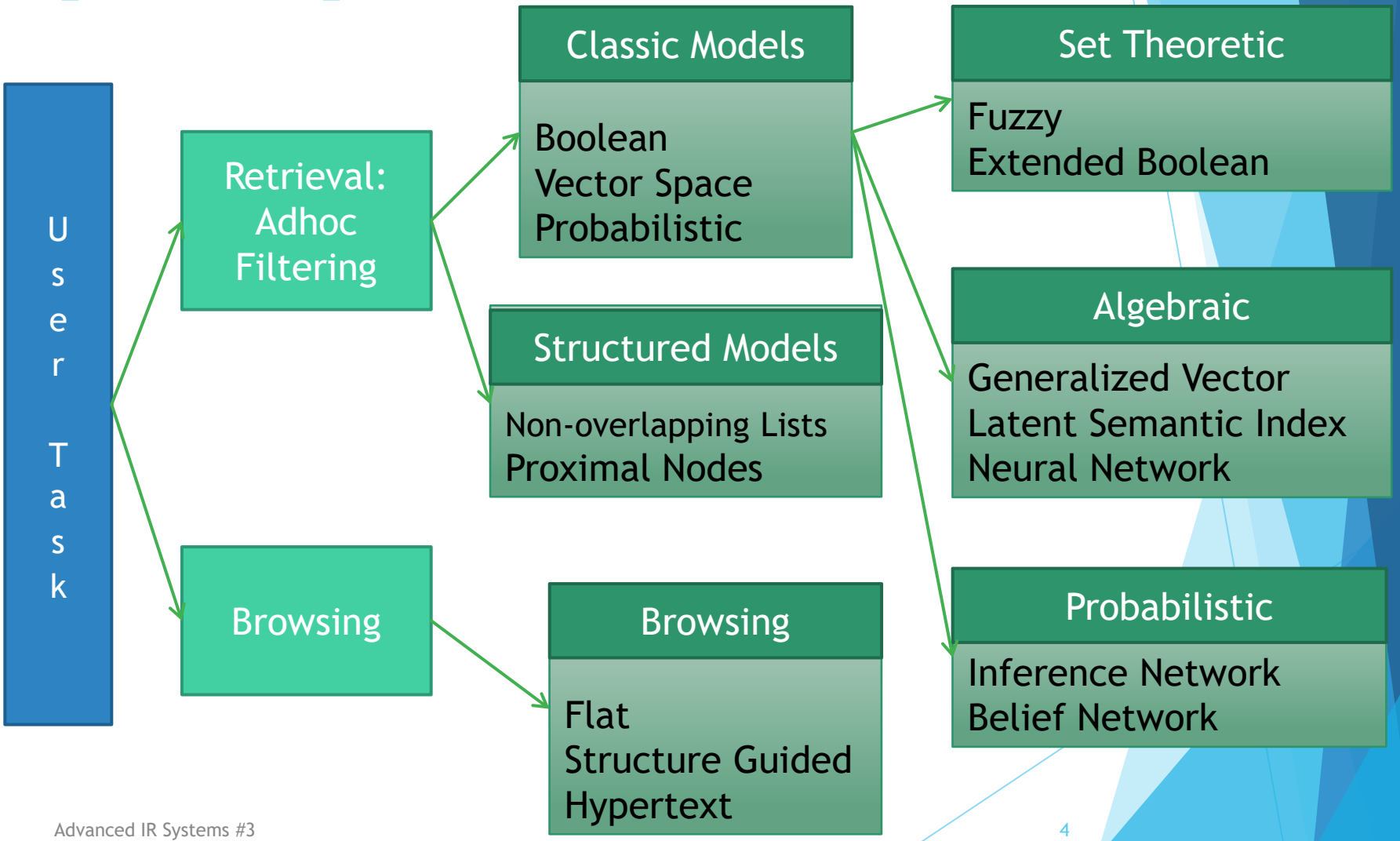
Modeling

(continued)

[Review] IR model overview



[Review] IR models



Classic IR Models

(continued)

[review] Basic Concepts

- Index terms: $k_1, k_2, \ldots, k_t$
- A document: d_j
- A weight associated with (k_i, d_j), which quantifies the importance of k_i for describing the contents of d_j : $w_{i,j}$ (>0)
  - If a term k_i does not occur within d_j : $w_{i,j}$=0
- A weighted vector associated with d_j :

$$\vec{d}_j = (w_{1,j}, w_{2,j}, \ldots, w_{t,j})$$

- a reference to the weight w_{ij} :

$$g_i(\vec{d}_j) = w_{i,j}$$

[review] Classic IR Models

- Boolean Model
- Vector Space Model
- Probabilistic Model

[review] The Vector Space Model (1)

- Use of binary weights is too limiting
- Non-binary weights provide consideration for partial matches

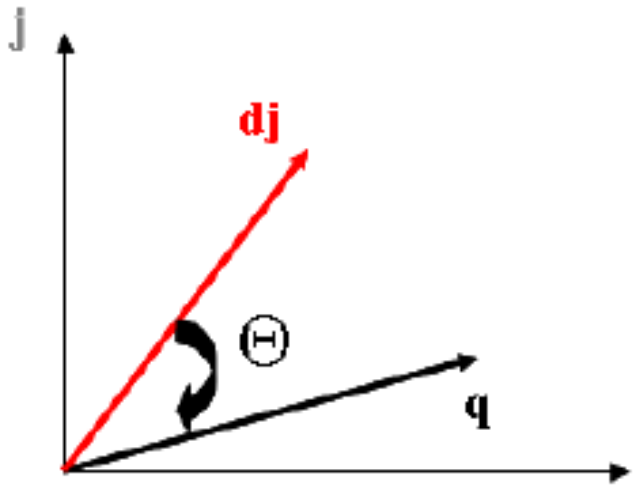
- Term weights are used to compute a degree of similarity between a query and each document
- Ranked set of documents provides for better matching

[review] The Vector Space Model (2)

- Define:
  - $w_{ij} > 0$ whenever $k_i \in d_j$
  - $w_{iq} > 0$ associated with the pair (k_i, q)
  - $\overrightarrow{d_j} = (w_{1j}, w_{2j}, \cdots w_{tj})$
  - $\overrightarrow{d_q} = (w_{1q}, w_{2q}, \cdots w_{tq})$
  - To each term k_i is associated a unitary vector $\vec{\imath}$
  - The unitary vectors $\vec{\imath}$ and $\vec{\jmath}$ are assumed to be orthonormal
    - i.e., index terms are assumed to occur independently within the documents

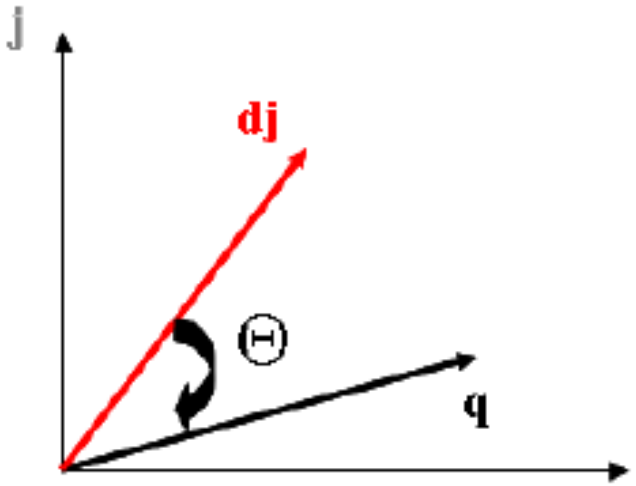
[review] The Vector Space Model (3)

- The t unitary vectors $\vec{\imath}$ form an orthonormal basis for a t-dimensional space
- In this space, queries and documents are represented as weighted vectors



[review] The Vector Space Model (4)

- Similarity
  - $\text{sim}(d_j, q) = \cos(\theta)$
  - $= \frac{\vec{d_j} \cdot \vec{q}}{|\vec{d_j}| \times |\vec{q}|}$
  - $= \frac{\sum_{i=1}^{t} w_{ij} \times w_{iq}}{\sqrt{\sum_{i=1}^{t} {w_{ij}}^2} \times \sqrt{\sum_{i=1}^{t} {w_{iq}}^2}}$



- $0 \leq \text{sim}(d_j, q) \leq 1$
- A document is retrieved even if it matches the query terms only partially

[review] How to compute the weights (1)

- How should we compute the weights w_{ij} and w_{iq} ?
- A good weight must take into account two effects:
  - quantification of intra-document contents (similarity)
    - *tf* factor, the term frequency within a document
  - quantification of inter-documents separation (dissimilarity)
    - *idf* factor, the inverse document frequency
  - $w_{ij} = tf_{ij} \times idf_i$

[review] How to compute the weights (2)

- Let, $k_i \in d_j$
  - the total number of documents in the collection: N
  - the number of documents which contain k_i: n_i
  - raw frequency of k_i within d_j: $\text{freq}_{i,j}$
- A normalized tf factor
  - $$tf_{i,j} = \frac{freq_{i,j}}{max_l freq_{l,j}}$$
  - where the maximum is computed over all terms which occur within the document d_j

[review] How to compute the weights (3)

- The *idf* factor

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

- the log is used to make the values of *tf* and *idf* comparable. It can also be interpreted as the amount of information associated with the term k_i.

Vector Space Model: Example

Documents

- d_1:
  Information retrieval (IR) in computing and information science is the process of obtaining information system resources that are relevant to an information need from a collection of those resources. Retrieval.
- d_2:
  The World Wide Web (WWW), commonly known as the Web, is the world's dominant software platform.[1] It is an information space where documents and other web resources can be accessed using a web browser and (more recently) web-based applications.
- d_3:
  The quick brown fox jumps over the lazy dog

Document -> Set of Index Terms

- d_1:
  information retrieval compute information science process obtain information system resource relevant information need collection resource retrieval
- d_2:
  world wide web common know web world dominant software platform information space document web resource access use web browser recent web application
- d_3:
  quick brown fox jump lazy dog

Put ID Numbers to the Index Terms

Information:1
retrieval:2
compute:3
science:4
process:5
obtain:6
system:7
resource:8
relevant:9
need:10
collection:11
world:12
wide:13
web:14
common:15
know:16
dominant:17
software:18
platform:19
space:20
document:21
access:22
use:23
browser:24
recent:25
application:26
quick:27
brown:28
fox:29
jump:30
lazy:31
dog:32

Document x ID Numbers of Index Terms

- d_1:
  information=1 retrieval=2 compute=3 information=1 science=4 process=5 obtain=6 information=1 system=7 resource=8 relevant=9 information=1 need=10 collection=11 resource=8 retrieval=2
  {1,2,3,1,4,5,6,1,7,8,9,1,10,11,8,2}
- d_2:
  world=12 wide=13 web=14 common=15 know=16 web=14 world=12 dominant=17 software=18 platform=19 information=1 space=20 document=21 web=14 resource=8 access=22 use=23 web=14 browser=24 recent=25 web=14 application=26
  {12,13,14,15,16,14,12,17,18,19,1,20,21,14,8,22,23,14,24,25,14,24,25,14,26}
- d_3:
  quick=27 brown=28 fox=29 jump=30 lazy=31 dog=32
  {27,28,29,30,31,32}

Document x Term Frequency

max

max

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
d_1	4	2	1	1	1	1	1	2	1	1	1	0	0	0	0	0
d_2	1	0	0	0	0	0	0	1	0	0	0	2	1	6	1	1
d_3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
d_1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
d_2	1	1	1	1	1	1	1	2	2	1	0	0	0	0	0	0
d_3	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1

max

Document x TF Value

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
d_1	1	0.5	0.25	0.25	0.25	0.25	0.25	0.5	0.25	0.25	0.25	0	0	0	0	0
d_2	0.17	0	0	0	0	0	0	0.17	0	0	0	0.33	0.17	1	0.17	0.17
d_3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
d_1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
d_2	0.17	0.17	0.17	0.17	0.17	0.17	0.17	0.33	0.33	0.17	0	0	0	0	0	0
d_3	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1

IDF Values

$N = 3$

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
n_i	2	1	1	1	1	1	1	2	1	1	1	1	1	1	1	1
$\frac{N}{n_i}$	1.5	3	3	3	3	3	3	1.5	3	3	3	3	3	3	3	3
idf	0.18	0.48	0.48	0.48	0.48	0.48	0.48	0.18	0.48	0.48	0.48	0.48	0.48	0.48	0.48	0.48

	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
n_i	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
$\frac{N}{n_i}$	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
idf	0.48	0.48	0.48	0.48	0.48	0.48	0.48	0.48	0.48	0.48	0.48	0.48	0.48	0.48	0.48	0.48

Document x TF-IDF Value

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
d_1	0.18	0.24	0.12	0.12	0.12	0.12	0.12	0.09	0.12	0.12	0.12	0	0	0	0	0
d_2	0.03	0	0	0	0	0	0	0.03	0	0	0	0.16	0.08	0.48	0.08	0.08
d_3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
d_1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
d_2	0.08	0.08	0.08	0.08	0.08	0.08	0.08	0.16	0.16	0.08	0	0	0	0	0	0
d_3	0	0	0	0	0	0	0	0	0	0	0.48	0.48	0.48	0.48	0.48	0.48

$$\left|\overrightarrow{d_1}\right| = 0.46 \qquad \left|\overrightarrow{d_2}\right| = 0.62 \qquad \left|\overrightarrow{d_3}\right| = 1.18$$

Query

- Q:
  information processing
- Index terms: information process
- ID numbers: information=1 process=5

$$Q = \{1,5\}$$

- Query x Term Frequency

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
q	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0

	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
q	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Query (cont'd)

- Query x TF Value

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
q	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
q	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

- Query x TF-IDF Value

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
q	0.18	0	0	0	0.48	0	0	0	0	0	0	0	0	0	0	0
	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
q	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

$$|\vec{q}| = 0.51$$

Similarity

- $sim(q, d_1) = \frac{w_{1,1} \times w_{1,q} + w_{5,1} \times w_{5,q}}{|\overrightarrow{d_1}| \times |\vec{q}|} = \frac{0.18 \times 0.18 + 0.12 \times 0.48}{0.46 \times 0.51} = \frac{0.09}{0.23} = 0.39$
- $sim(q, d_2) = \frac{w_{1,2} \times w_{1,q} + w_{5,2} \times w_{5,q}}{|\overrightarrow{d_2}| \times |\vec{q}|} = \frac{0.03 \times 0.18 + 0 \times 0.48}{0.62 \times 0.51} = \frac{0.0054}{0.32} = 0.02$
- $sim(q, d_3) = \frac{w_{1,3} \times w_{1,q} + w_{5,3} \times w_{5,q}}{|\overrightarrow{d_3}| \times |\vec{q}|} = \frac{0. \times 0.18 + 0 \times 0.48}{1.18 \times 0.51} = \frac{0}{0.6} = 0$

Classic IR Models

- Boolean Model
- Vector Space Model
- Probabilistic Model

Probabilistic Model

- Objective: to capture the IR problem using a probabilistic framework
- Given a user query, there is an ideal answer set
- Querying as specification of the properties of this ideal answer set (clustering)
- But, what are these properties?
- Guess at the beginning what they could be (i.e., guess initial description of ideal answer set)
- Improve by iteration

Principle

- A user query q and a document d_j
  - The probabilistic model tries to estimate the probability that the user will find the document d_j interesting (i.e., relevant)
  - The model assumes that this probability of relevance depends on the query and the document representations only
- Ideal answer set: R
  - It should maximize the probability of relevance
  - Documents in the set R are predicted to be relevant.

Computing Similarity (1)

- Definition of similarity
  - $\text{sim}(d_j, q) = \frac{P(R|\overrightarrow{d_j})}{P(\bar{R}|\overrightarrow{d_j})}$
    - $P(R|\overrightarrow{d_j})$: probability of document d_j being an element of the set of relevant documents (R)
    - $P(\bar{R}|\overrightarrow{d_j})$: probability of document d_j being an element of the set of non-relevant documents ($\bar{R}$)

Computing Similarity (2)

- Let,
  - $P(R)$: probability that a document randomly selected from the entire collection is relevant
  - $P(\bar{R})$: probability that a document randomly selected from the entire collection is not relevant
  - $P(\overrightarrow{d_j}|R)$: probability of randomly selecting the document d_j from the set of relevant documents (R)
  - $P(\overrightarrow{d_j}|\bar{R})$: probability of randomly selecting the document d_j from the set of non-relevant documents ($\bar{R}$)
- $\text{sim}(d_j, q) = \frac{P(R|\overrightarrow{d_j})}{P(\bar{R}|\overrightarrow{d_j})} = \frac{P(\overrightarrow{d_j}|R)\times P(R)}{P(\overrightarrow{d_j}|\bar{R})\times P(\bar{R})} \sim \frac{P(\overrightarrow{d_j}|R)}{P(\overrightarrow{d_j}|\bar{R})}$

Computing Similarity (3)

- Assuming independence of index terms
  - $\text{sim}(d_j, q) \sim \frac{\left(\prod_{g_i(\overrightarrow{d_j})=1} P(k_i|R)\right) \times \left(\prod_{g_i(\overrightarrow{d_j})=0} P(\overline{k_i}|R)\right)}{\left(\prod_{g_i(\overrightarrow{d_j})=1} P(k_i|\bar{R})\right) \times \left(\prod_{g_i(\overrightarrow{d_j})=0} P(\overline{k_i}|\bar{R})\right)}$
    - $P(k_i|R)$: probability that the index term k_i is present in a document randomly selected from the set R of relevant documents
    - $P(\overline{k_i}|R)$: probability that the index term k_i is not present in a document randomly selected from the set R
    - The probabilities associated with the $\bar{R}$ have meanings which are analogous to the ones just described

Computing Similarity (4)

- Finally
  - $\text{sim}(d_j, q) \sim \sum_{i=1}^{t} w_{iq} \times w_{ij} \times \left(\log \frac{P(k_i|R)}{1-P(k_i|R)} + \log \frac{1-P(k_i|\bar{R})}{P(k_i|\bar{R})} \right)$
    - $P(\bar{k}_i|R) = 1 - P(k_i|R)$
    - $P(\bar{k}_i|\bar{R}) = 1 - P(k_i|\bar{R})$

Initial Ranking

- $\text{sim}(d_j, q) \sim \sum_{i=1}^{t} w_{iq} \times w_{ij} \times \left(\log \frac{P(k_i|R)}{1-P(k_i|R)} + \log \frac{1-P(k_i|\bar{R})}{P(k_i|\bar{R})} \right)$
  - Probabilities $P(k_i|R)$ and $P(k_i|\bar{R})$?
- Estimate based on assumptions
  - $P(k_i|R) = 0.5$
  - $P(k_i|\bar{R}) = \frac{n_i}{N}$
    - n_i: the number of documents that contain k_i
    - N : the number of documents in the document set
    - Use this initial guess to retrieve an initial ranking
    - Improve upon this initial ranking

Improving the Ranking (1)

- Let,
  - V: set of documents initially retrieved
  - V_i: subset of documents retrieved that contain k_i
- Reevaluate estimates:
  - $P(k_i|R) = \frac{V_i}{V}$
  - $P(k_i|\bar{R}) = \frac{n_i - V_i}{N - V}$
- Repeat recursively

Improving the Ranking (2)

- Another estimation:
  - to avoid problems with $V = 1, V_i = 0$
  - $P(k_i|R) = \frac{V_i + 0.5}{V + 1}$
  - $P(k_i|\bar{R}) = \frac{n_i - V_i + 0.5}{N - V + 1}$
- Also
  - $P(k_i|R) = \frac{V_i + \frac{n_i}{N}}{V + 1}$
  - $P(k_i|\bar{R}) = \frac{n_i - V_i + \frac{n_i}{N}}{N - V + 1}$

Advantages and Disadvantages

- Advantages:
  - Documents ranked in decreasing order of probability of relevance
- Disadvantages:
  - need to guess initial estimates for $P(k_i|R)$
  - method does not take into account *tf* and *idf* factors

Comparison of the Models

- Boolean model does not provide for partial matches and is considered to be the weakest classic model
- Croft showed that the probabilistic model outperforms the vector space model
- Salton and Buckley did a series of experiments that indicate that, in general, the vector space model outperforms the probabilistic model with general collections

That's it today

See you next week

今日はここまで

来週も授業あり！