Feb 10 and 12, 2014

CS533 - Information Retrieval Systems

Class Notes – Prepared by Gülsüm Ece Bıçakcı

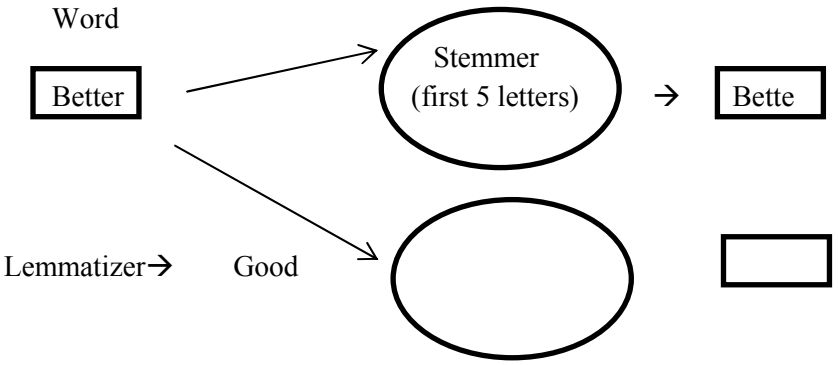
$$D = \begin{bmatrix} 2\ 0\ 5 \\ 1\ 2\ 0 \\ 1\ 3\ 2 \end{bmatrix}$$

Binary form → keep terms that appears at least twice

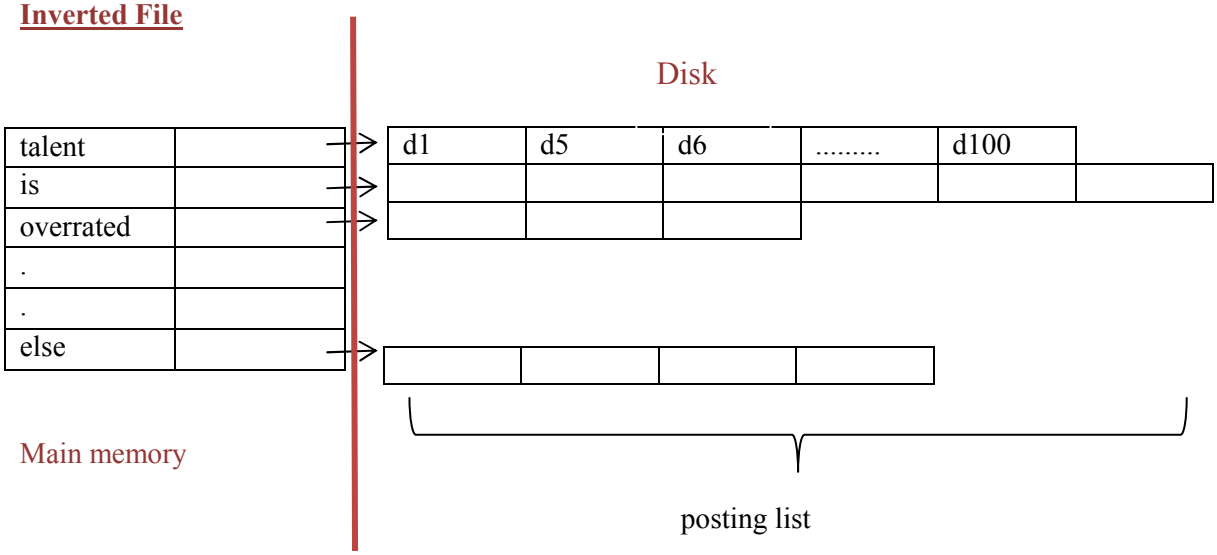
$$D = \begin{bmatrix} 1\ 0\ 1 \\ 0\ 1\ 0 \\ 0\ 1\ 1 \end{bmatrix}$$

For words, we may use **normalization**→ rather than using words as they are use their stems.

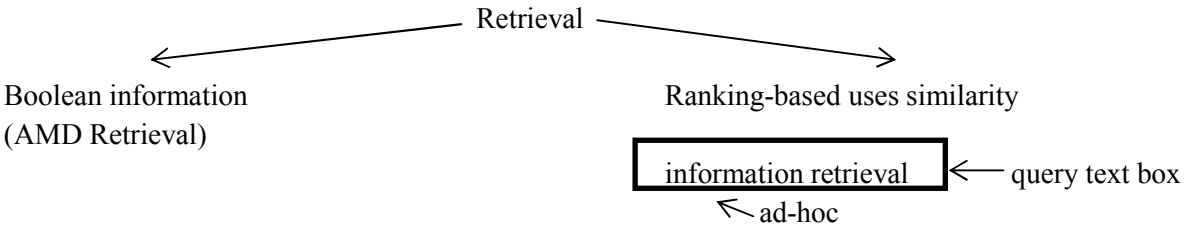
- Stem vs. Lemma



Inverted File

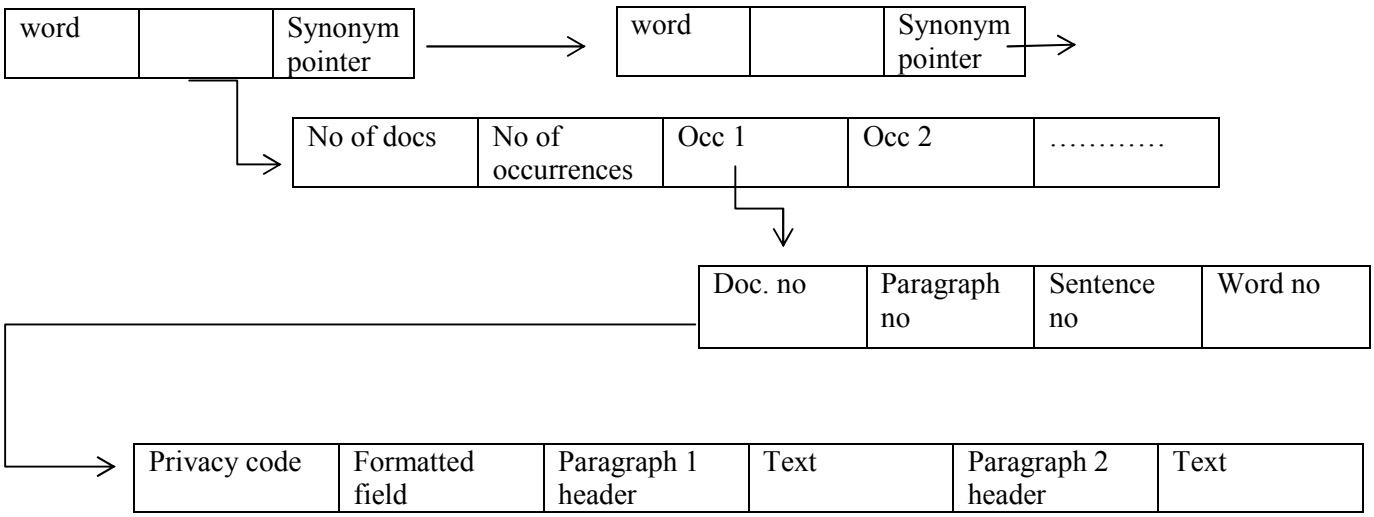


d_no	Term weight



Example: Search Engine
Classic

STAIRS: Storage and Information Retrieval System (it is a software by IBM)



Author name
Journal no
Page no

STAIRS has 2 programs

1. Utility for database creation and maintenance
2. Query and retrieval systems (AQUARIUS)

Modes of Operations

1. SEARCH mode: for textual retrieval
2. SELECT mode: for structured information retrieval

Search example:

HEART

HEART or DISEASE

HEART$

DISEASE$3

HEART and DISEASE: in the same sentence

HEART SAME DISEASE: in the same paragraph

Ranking

After doing SEARCH how to rank?

a: The frequencey of the term in the document

b: The frequencey of the term in the retrieval set

c: The number of documents in the retrieval set in which the term appears

Value of a term = (a x b) / c

Score of a document = ∑ value of all query terms which appear in this document

Example:

a: 16 appears a times in this document

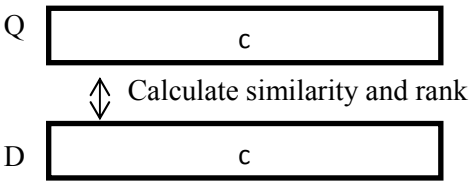
b : 1,247 appears b times in all of the retrieved docs

c: 152 appears c many of distinct documents

value = (16 x 1,247) / 152 = 131.26

Similarity Calculation

Vector Space Model



Measure	Binary	Formula
Inner Product	$\|X \cap Y\|$	$\sum_{i=1}^{n} Xi.Yi$
Dice Coefficient	$\frac{2\|X \cap Y\|}{\|X\| + \|Y\|}$	$\frac{2\sum_{i=1}^{n} Xi.Yi}{\sum_{i=1}^{n}(Xi)^2 + \sum_{i=1}^{n}(Yi)^2}$
Cosine Coefficient	$\frac{\|X \cap Y\|}{\|X\|^{1/2} x \|Y\|^{1/2}}$	$\frac{\sum_{i=1}^{n} Xi.Yi}{(\sum_{i=1}^{n}(Xi)^2 \; x \; \sum_{i=1}^{n}(Yi)^2)^{1/2}}$
Jaccard Coefficient	$\frac{\|X \cap Y\|}{\|X\| + \|Y\| - X \cap Y\|}$	$\frac{\sum_{i=1}^{n} Xi.Yi}{\sum_{i=1}^{n}(Xi)^2 + \sum_{i=1}^{n}(Yi)^2 - \sum_{i=1}^{n} Xi.Yi}$

Example:

X = (1 0 1 1 1) |x| = 4

Y = (1 1 0 1 0) |y| = 3

Inner Product = 2

Dice Coefficient = (2 x 2) / (4 + 3) = 4 / 7

Cosine Coefficient = 2 / (4^1/2 x 3^1/2)

X = (2 0 1 3 2)

Y = (1 0 2 1 5)

Dice Coefficient = 2 (2x1 + 1x2 + 3x1 + 2x5) / (4+1+9+4) + (1+4+1+25) ~ 0.69

Cosine Coefficient = (2x1 + 1x2 + 3x1 + 2x5) / (18.31)^1/2 = 0.72

Construction of Similarity (Proximity) Matrices

t1 t2 t3 t4 t5 t6

D =

	d11	1	0	0	1	0
	d21	1	0	1	1	0
0	0	d30	0	1		
0	0	d40	0	1		
0	0	d51	0	1		

S =

1.00	S12	S13	S14	S15	4
X	1.00	S23	S24	S25	3
X	X	1.00	S34	S35	2
X	XX	1.00	S45		1
X	XXX	1.00			

Similarity matrix

Sij = Sji

4 + 3 + 2 + 1 = 10

(N-1) + (N-2) + …… + 2 + 1 = $\sum_{i=1}^{n} i$

n (n-1) / 2 → O(n^2)

Similarity Calculation Methods

1. **Brute Force:** Calculate every similarity value
   m = number of documents

   fori = 1 to m-1
   for j = i + 1 to m
   findSij
   end for
   end for

2. Using the knowledge of term distributions in documents.

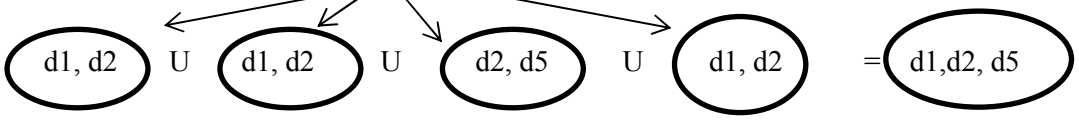
2 documents can have a similarity value > 0 if they have one or more common terms.

t1→ d1, d2
t2→ d1, d2
t3→ d4, d5
t4→ d2, d5
t5→ d1, d2
t6→ d3, d4, d5

Consider d1: contains t1, t2 and t5

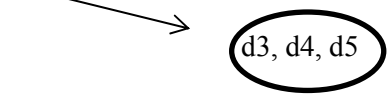
Computed S12

Consider d2: contains t1, t2, t4 and t5



Computed S25

Consider d3: contains t6



Computed S34, S35

Consider d4: S45

Consider d5: anything

Total number of similar items we calculate = 5
There is 50 % saving

The knowledge of terms distribution is good, however we still have some unnecessary computations during term matching.

Consider finding number of common entries between two vectors.

xlng = number of elements in x

X:

5	7	8

Y:

5	7	8

xndx←

yndx←

sum = 0

while (xndx<= xlng and yndx<= ylng)

x(xndx) = y(yndx): sum = sum + 1;

xndx ++;

yndx ++;

x(xndx) < y(yndx): xndx ++

.

.

.

3. Using an inverted index file for terms

t1 →<1,1><2,1>
t2 →<1,1><2,1>
t3 →<4,1><5,1>
t4 →<2,1><5,1>
t5 →<1,1><2,1>
t6 →<3,1><4,1><5,1>

posting list

Document Length Info

3	4	1	2	3
d1	d2	d3	d4	d5

Consider d1 : contains t1, t2 and t5

X	0	0	0	0

← Mail Boxes

due to t1

X	1 ~~0~~	0	0	0

t2

X	2 ~~1~~	0	0	0

t5

X	3 ~~2~~	0	0	0

Final Similarity Value →

X	(2 x 3) / (3+4)	0	0	0

d1 →<t1,1><t2,1><t5,1>
d2 →…….

xd = depth of indexing average number of terms / documents

3 + 4 + 1 + 2 + 3 / 5 = t, where 5 = m

tg = term generality average number of documents / term = t / n, where n = 6

xd .tg + xd . tg + ……. + xd . tg (except last one)

(m-1) . xd .tg = m . xd .tg (since m is a large number)

$$S = \begin{bmatrix} 1.00 & S12 & S13 & S14 & S15 \\ X & 1.00 & S23 & S24 & S25 \\ X & X & 1.00 & S34 & S35 \\ X & XX & 1.00 & S45 & \\ X & XXX & 1.00 & & \end{bmatrix}$$

During the computations go over the posting list from right to left (from higher document to lower document no.)

$$xd \,.tg\,.\,(m-1)/m + xd\,.\,tg\,.\,(m-2)/m + \ldots\ldots + xd\,.tg\,.\,1/m = (xd\,.tg\,/\,m)\,.\,[(m-1)+(m-2)+\ldots..+1]$$

$$= (xd\,.tg\,/\,m)\,.\,(m\,.\,(m-1)\,/\,2)$$

$$\sim m\,.xd\,.tg\,/\,2$$

Performance Evaluation

Effectiveness and Efficiency

Precision time: response time
Recall space

Precision = number of relevant documents / total number of documents retrieved

Recall = number of relevant documents / total number of relevant documents in collection

Cranfield approach to Information Retrieval Experiments

We work in a lab environment.
We have a test collection.
a set of documents
a set of queries
relevant documents for each query

Pooling approach for determining relevant documents for each query

Pooling Approach

Q → IRS1 IRS2 IRS3 …….. IRS10

TREC queries
A few words: ad – hoc
A narrative

F = 2 x Precision x Recall / Precision + Recall → harmonic mean of R & P
This measure can be found in Van Rijsbergen Information Retrieval in the Evaluation part.

User's Perspective in terms of Evaluation

- Database / collection coverage
- Live links (alive pages)
- Response time
- More recent ones are provided at the beginning (for a football match showing recent results are more important)
- Duplicates are eliminated (near duplicates)
- Personalized results
- Query suggestion / correction (query completion)
- Query diversification
- Context based results (according to region)

	Relevant	**Not Relevant**
Retrieved	**a (TP)**	**b (FP)**
Not retrieved	**c (FN)**	**d (TN)**

(P) Precision = a / a + b

(R) Recall = a / a + c

TP = True Positive

TN = True Negative

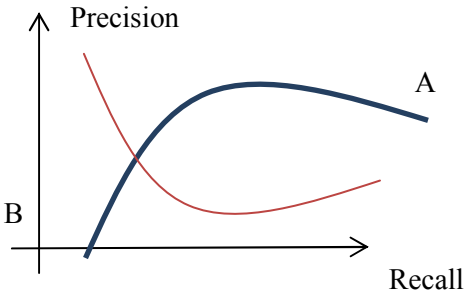
FP = False Positive

FN = False Negative

P = TP / TP + FP

R = TP / TP + FN

True means correct decision about its relevance.



Which is more logical? B

Inverted files for text search engines, J. Zobel, A. Moffat, which is on the course web site, is a good source for this topic.

Questions

1. Search engines have some similiarities with database systems. However, in some points they are different from each other. What are these main differences? Write 3 of them.

2. To stem and casefold is removing variant ending from words and converting to lowercase. Also, stopping is the process of removing the stop words such as and, in, the, had, etc. According to the below example, apply the casefolding, stemming and stopping methods respectively.

```
1    The old night keeper keeps the keep in the town
2    In the big old house in the big old gown.
3    The house in the town had the big old keep
4    Where the old night keeper never did sleep.
5    The night keeper keeps the keep in the night
6    And keeps in the dark and sleeps in the light.
```

Fig. 1. The Keeper database. It consists of six one-line documents.

3. All current search engines use ranking to identify potential answers. In a ranked query, a statistical similarity heuristic or similarity measure is used to assess the closeness of each document to the textual query. A typical older formulation that is effective in practice calculates the cosine of the angle in n-dimensional space between a query vector $<wq,t>$ and a document vector $<wd,t>$. **[1]** Write down these cosine formulas.

4. Suppose that we have an information retrieval system that includes 5 relevant documents, and 10 nonrelevant documents. There are 25 relevant documents in the collection. What is the precision and recall of the system?

Answers

1.

Database Systems	Search Engines
To contend with arbitrarily complex queries.	Most of the queries are lists of terms and phrases.
A match is a record that meets a specified logical condition.	A match is a document that is appropriate to the query according to statistical heuristics and may not even contain all of the query terms.
They return all matching records.	They return a fixed number of matches which are ranked by their statistical similarity.

2.

With casefolding method:

```
and big dark did gown had house in keep keeper keeps light
never night old sleep sleeps the town where
```

[2]

After casefolding, with stemming method:

```
and big dark did gown had house in keep light never night old
sleep the town where
```

[2]

After stemming, with stopping method:

```
big dark gown house keep light night old sleep town
```

[2]

3.

$$w_{q,t} = \ln\left(1 + \frac{N}{f_t}\right) \qquad w_{d,t} = 1 + \ln f_{d,t}$$

$$W_d = \sqrt{\sum_t w_{d,t}^2} \qquad W_q = \sqrt{\sum_t w_{q,t}^2}$$

$$S_{q,d} = \frac{\sum_t w_{d,t} \cdot w_{q,t}}{W_d \cdot W_q}.$$

[2]

—*fd,t:* the frequency of term *t* in document *d*
—*f* : the number of documents containing one or more occurrences of term *t*
—*N*: the number of documents in the collection

4. Precision = # relevant items retrieved / # retrieved items = 5/15
Recall = # relevant items retrieved / # relevant items = 5/25

[1], [2], [3]: Inverted files for text search engines, J. Zobel, A. Moffat.