Θα μιλήσουμε για

ΜΟΝΤΕΛΑ ΑΝΑΚΤΗΣΗΣ ΠΛΗΡΟΦΟΡΙΑΣ

Διαφάνειες του καθ. **Γιάννη Τζίτζικα** (Παν. Κρήτης)

http://www.ics.forth.gr/~tzitzik/

Για το πιθανοτικό του καθ. **Απ. Παπαδόπουλου** (Αριστοτέλειο Παν.)

Κεφάλαιο 2 του βιβλίου

1

Διάρθρωση

- **Εισαγωγή στα Μοντέλα Ανάκτησης**
- **Κατηγορίες Μοντέλων**
- **Απόλυτο και Κάλλιστο (ή Βέλτιστο) Ταίριασμα (Exact vs Best Match)**
- **Τα Τρία Κλασσικά Μοντέλα Ανάκτησης**
- **Επεκτάσεις**

Αναπαράσταση Εγγράφων:

Πως βλέπουμε ένα έγγραφο;

- Πως βλέπουμε ένα έγγραφο;
  - Ως έχει (full text);
  - Αγνοώντας λέξεις που δεν φέρουν νόημα (π.χ. τα άρθρα) ;
  - Ως σάκο (bag) όρων ευρετηρίου (bag of index terms), δηλαδή αγνοώντας τη σειρά με την οποία εμφανίζονται οι λέξεις στο κείμενο;
  - Ως σύνολο όρων ευρετηρίου (set of Index terms)
  - Ως δομημένο έγγραφο (π.χ. hypertext, XML)
- Η απάντηση σε αυτό το ερώτημα θα καθορίσει τη μορφή του ευρετηρίου που πρέπει να κατασκευάσουμε.
- Η απάντηση σε αυτό το ερώτημα είναι συνυφασμένη **και** με το μοντέλο ανάκτησης που πρόκειται χρησιμοποιήσουμε.

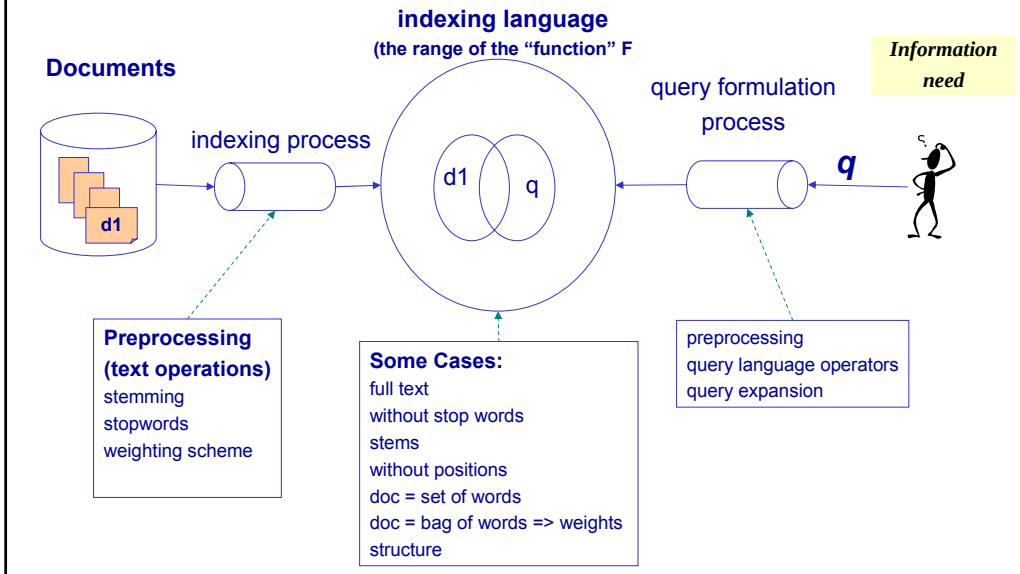
Μοντέλα Ανάκτησης

- Ένα μοντέλο ανάκτησης ορίζει
  - Αναπαράσταση Εγγράφων
  - Αναπαράσταση Επερωτήσεων
  - Καθορίζει και ποσοτικοποιεί την έννοια της συνάφειας
    - ο βαθμός συνάφειας μπορεί να είναι δίτιμος (π.χ. {1,0}), ή συνεχής (π.χ. [0,1])

Έστω **D** η συλλογή εγγράφων και **Q** το σύνολο όλων των πληροφοριακών αναγκών που μπορεί να έχει ένας χρήστης.

Μπορούμε να δούμε ένα ***μοντέλο ανάκτησης πληροφορίας*** ως μια τετράδα [F, D, Q, R] όπου:

- D: λογικές όψεις εγγράφων $D=\{ F(d) \mid \mathbf{d} \in \mathbf{D}\}$
- Q: λογικές όψεις επερωτήσεων $Q=\{ F(q) \mid \mathbf{q} \in \mathbf{Q}\}$
- F: πλαίσιο μοντελοποίησης εγγράφων, επερωτήσεων και των σχέσεων μεταξύ τους
- R: συνάρτηση κατάταξης που αποδίδει μία τιμή σε κάθε ζεύγος $(d,q) \in D \times Q$
  - δίτιμη: $R: D \times Q \rightarrow [True/False]$
  - συνεχής $R: D \times Q \rightarrow [0,1]$



Κατηγορίες Μοντέλων Ανάκτησης

Τι θα δούμε σήμερα:

Λογικό μοντέλο για το κείμενο, την ερώτηση και τη συνάρτηση ομοιότητας μεταξύ τους

- Κλασσικά Μοντέλα
  - Boolean Model
  - Διανυσματικό (Vector Space)
  - Πιθανοκρατικό (Probabilistic)

Λέξεις Κλειδιά (Keywords)

Χρησιμοποιούνται ως αντιπρόσωποι όλου του κειμένου και βοηθούν στη σύντομη περιγραφή του κειμένου (περίληψη).

Απαιτείται προσοχή στην επιλογή τους, έτσι ώστε τα κείμενα να διαχωρίζονται κατάλληλα.

Το πλήθος των όρων είναι συνήθως μεγάλο και προηγείται απαλοιφή τετριμμένων λέξεων (π.χ., άρθρα, σύνδεσμοι κλπ)

Παράδειγμα

Κείμενο 1	Κείμενο 2	Κείμενο 3
... η γεωργική επανάσταση	... η βιομηχανική επανάσταση	... η επανάσταση υψηλής τεχνολογίας

Η επιλογή της λέξης *επανάσταση* σαν λέξη κλειδί για τα τρία κείμενα δημιουργεί πρόβλημα. Γιατί;

Κλασσικά Μοντέλα

Όλες οι λέξεις κλειδιά (αλλιώς όροι -term) δεν έχουν την ίδια βαρύτητα για τις προτιμήσεις των χρηστών. Κάποιες λέξεις μπορεί να είναι σημαντικές ενώ κάποιες άλλες λιγότερο σημαντικές.

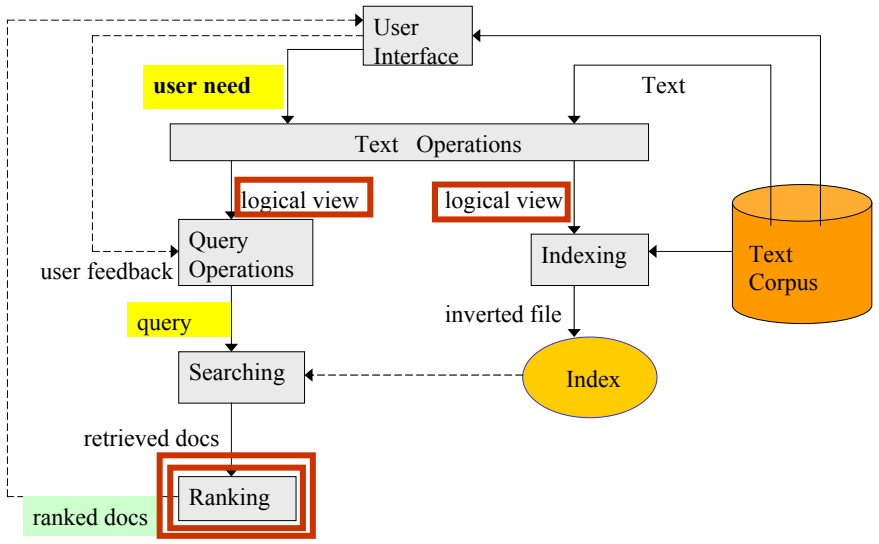
Έστω t_i ένας όρος και d_j ένα έγγραφο. Το βάρος του όρου t_i στο έγγραφο d_j συμβολίζεται ως $w(t_i,d_j) >= 0$ (ή απλούστερα w_{ij}) και δηλώνει το πόσο σημαντικός είναι ο όρος t_i σε σχέση με το έγγραφο d_j.

Έστω m αριθμός των όρων και $T=\{t_1,\ldots,t_m\}$ το σύνολο των μοναδικών όρων. Εάν ο όρος t_i δεν εμφανίζεται στο έγγραφο d_j τότε $w(t_i,d_j)=0$. Διαφορετικά, $w(k_i,d_j) > 0$.

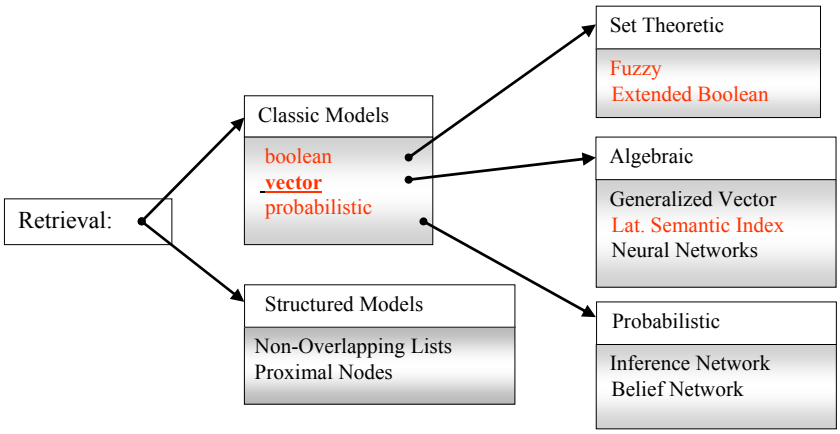
Άρα σε κάθε κείμενο $\mathbf{d}_j$ αντιστοιχεί ένα m-διάστατο διάνυσμα βαρών

$(w_{1,j}, w_{2,j}, \ldots, w_{m,j})$.

Τα τμήματα της αρχιτεκτονικής που εμπλέκονται



Μια Ταξινομία των Μοντέλων Ανάκτησης



Exact vs. Best Match Retrieval Models

- Exact-match (Απόλυτου Ταιριάσματος)
  - μια επερώτηση καθορίζει **αυστηρά (απόλυτα) κριτήρια ανάκτησης**
  - κάθε έγγραφο **είτε ταιριάζει είτε όχι** με μία επερώτηση
  - το αποτέλεσμα είναι ένα **σύνολο** κειμένων
- Best-match (Κάλλιστου Ταιριάσματος)
  - μια επερώτηση **δεν περιγράφει αυστηρά** κριτήρια ανάκτησης
  - **κάθε** έγγραφο ταιριάζει σε μια επερώτηση **σε ένα βαθμό**
  - το αποτέλεσμα είναι μια **διατεταγμένη λίστα** εγγράφων
  - με ένα κατώφλι (στο βαθμό συνάφειας) μπορούμε να ελέγξουμε το μέγεθος της απάντησης
- «Μικτές προσεγγίσεις»
  - συνδυασμός απόλυτου ταιριάσματος με τρόπους διάταξης του συνόλου της απάντησης
    - E.g., best-match query language that incorporates exact-match operators

Βελτίων = συγκριτικός του αγαθός

Information Retrieval Models

Boolean Retrieval Model

Boolean Retrieval Model

- Έγγραφο = σύνολο λέξεων κλειδιών (keywords)
- Επερώτηση = Boolean έκφραση λέξεων κλειδιών (AND,OR, NOT, παρενθέσεις)
  - πχ επερώτησης
    - ((Crete AND Greece) OR (Oia AND Santorini)) AND Hotel AND-NOT Hilton
    - ((Crete & Greece) | (Oia & Santorini)) & Hotel & ! Hilton
- Απάντηση= σύνολο εγγράφων
  - απουσία διάταξης

Παράσταση εγγράφων κατά το Boolean Model

	k_1	k_2	....	k_t
d_1	w_{11}	w_{21}	...	w_{t1}
d_2	w_{12}	w_{22}	...	w_{t2}
:	:	:		:
:	:	:		:
d_n	w_{1n}	w_{2n}	...	w_{tn}

$w_{i,j} \in \{0,1\}$

- K={k_1,…,k_t} : σύνολο όλων των λέξεων ευρετηρίασης
- Κάθε έγγραφο **d_j** παριστάνεται με το διάνυσμα d_j=($w_{1,j}$,…,$w_{t,j}$) όπου:
  - $w_{i,j}$ = **1** αν η λέξη k_i εμφανίζεται στο έγγραφο **d_j** (αλλιώς $w_{i,j}$ =**0**)

Boolean Retrieval Model: Formally

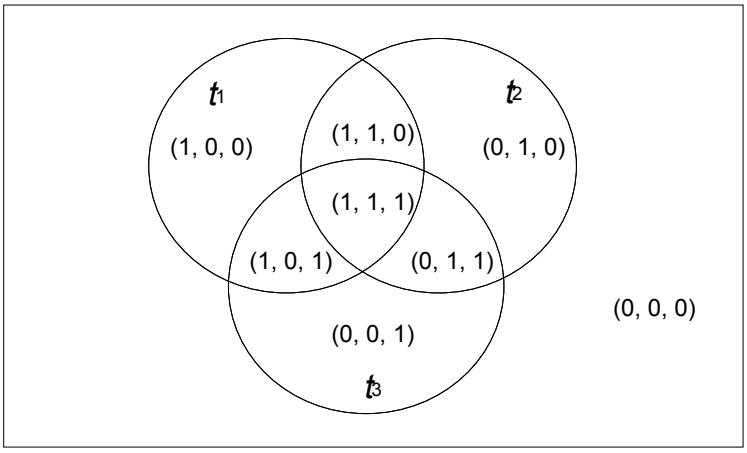
- K={k_1,…,k_t} : σύνολο όλων των λέξεων ευρετηρίασης
- Κάθε έγγραφο **d_j** παριστάνεται με το διάνυσμα d_j=($w_{1,j}$,…,$w_{t,j}$) όπου:
  - $w_{i,j}$ = **1** αν η λέξη k_i εμφανίζεται στο κείμενο **d_j** (αλλιώς $w_{i,j}$ =**0**)
- Μια επερώτηση q είναι μια λογική έκφραση στο K, πχ:

$$q = (k_1 \vee k_2) \wedge k_3$$

Μετατροπή σε DNF

Boolean Μοντέλο

$q = (t_1 \vee t_2) \wedge t_3$ $\quad q = \bigvee (\ldots \wedge \ldots)$



Boolean Μοντέλο

Πίνακας αληθείας του ερωτήματος $(t1 \vee t2) \wedge t3$

t_1	t_2	t_3	διάνυσμα	έκφραση	απάντηση
0	0	0	(0, 0, 0)	$\neg t_1 \wedge \neg t_2 \wedge \neg t_3$	0
0	0	1	(0, 0, 1)	$\neg t_1 \wedge \neg t_2 \wedge t_3$	0
0	1	0	(0, 1, 0)	$\neg t_1 \wedge t_2 \wedge \neg t_3$	0
0	1	1	(0, 1, 1)	$\neg t_1 \wedge t_2 \wedge t_3$	1
1	0	0	(1, 0, 0)	$t_1 \wedge \neg t_2 \wedge \neg t_3$	0
1	0	1	(1, 0, 1)	$t_1 \wedge \neg t_2 \wedge t_3$	1
1	1	0	(1, 1, 0)	$t_1 \wedge t_2 \wedge \neg t_3$	0
1	1	1	(1, 1, 1)	$t_1 \wedge t_2 \wedge t_3$	1

Boolean Retrieval Model: Formally

- K={k_1,…,k_t} : σύνολο όλων των λέξεων ευρετηρίασης
- Κάθε έγγραφο **d_j** παριστάνεται με το διάνυσμα d_j=($w_{1,j}$,…,$w_{t,j}$) όπου:
  - $w_{i,j}$ **= 1** αν η λέξη k_i εμφανίζεται στο κείμενο **d_j** (αλλιώς $w_{i,j}$ =**0**)
- Μια επερώτηση q είναι μια λογική έκφραση στο K, πχ:
  - q = "**k1 and (k2 or not k3))**" δηλαδή q = "**k1 ∧ (k2 ∨ ¬ k3))**"
  - q_{DNF} = "**(k1 ∧ k2 ∧ k3) ∨(k1 ∧ k2 ∧¬ k3) ∨(k1 ∧ ¬ k2 ∧ ¬ k3)**"
  - q_{DNF} = "**(1,1,1) ∨(1,1,0) ∨(1,0,0)**"
- R(d,q)=
  - **True** αν υπάρχει συζευκτική συνιστώσα του q με λέξεις των οποίων τα βάρη είναι τα ίδια με αυτά των αντίστοιχων λέξεων του εγγράφου d
  - **False,** αλλιώς

Boolean Retrieval Model: Ισοδύναμος ορισμός

Αποτίμηση επερωτήσεων (με χρήση λογικής)

- ένα κείμενο d είναι μια **σύζευξη όρων**, όπου **όρος** μια λέξη σε θετική ή αρνητική μορφή (σε θετική αν εμφανίζεται στο κείμενο, αλλιώς σε αρνητική)
- μια επερώτηση q είναι μια οποιαδήποτε λογική έκφραση
- **R(d,q)=True if and only if d |=q**
  - δηλαδή αν κάθε ερμηνεία που αληθεύει το d αληθεύει και το q

Boolean Retrieval Model:
Ένας εναλλακτικός τρόπος ορισμού

Μπορούμε να ορίσουμε ως ερμηνεία μιας λέξης (του K) το σύνολο των εγγράφων που την περιέχουν.
Άρα η ερμηνεία είναι μια συνάρτηση I: $K \rightarrow 2^D$ που ορίζεται ως εξής:
$I(k) = \{ d \mid d$ περιέχει τη λέξη $k\}$
Έστω E το σύνολο των λογικών εκφράσεων με λέξεις από το σύνολο K.

Μπορούμε να επεκτείνουμε μια ερμηνεία I του K σε μια ερμηνεία J του E ως εξής
$J(t) = I(t)$
$J(e \wedge e') = J(e) \cap J(e')$
$J(e \vee e') = J(e) \cup J(e')$
$J(e \wedge \neg e') = J(e) \setminus J(e')$
Η απάντηση μιας επερώτησης q (κατά το Boolean μοντέλο) είναι η εξής:
$ans(q) = J(q)$

Οι αδυναμίες του Boolean μοντέλου
Η αδυναμία ελέγχου του μεγέθους της απάντησης

- Παράδειγμα:
  - |Answer("Cheap ∧ Tickets ∧ Heraklion") |= 1
  - |Answer("Cheap ∧ Tickets) |= 1000
  - |Answer("Cheap ∧ Heraklion) |= 1000
  - |Answer("Tickets ∧ Heraklion") |= 1000
- Άρα είτε παίρνουμε μια απάντηση με ένα έγγραφο είτε ένα σύνολο 1000 εγγράφων. :(

Οι αδυναμίες του Boolean μοντέλου

- Άκαμπτο: AND σημαίνει όλα, OR σημαίνει οποιοδήποτε
- Δυσκολίες
  - Ο έλεγχος του μεγέθους της απάντησης
    - All matched documents will be returned
  - Ικανοποιητική ακρίβεια (precision) συχνά σημαίνει απαράδεκτη ανάκληση (recall)
  - Η διατύπωση των επερωτήσεων είναι δύσκολη για πολλούς χρήστες
  - Η έκφραση σύνθετων πληροφοριακών αναγκών είναι δύσκολη
  - Δεν μας λέει πώς να διατάξουμε την απάντηση
    - All matched documents logically satisfy the query
  - Τα μοντέλα κατάταξης (ranking models) έχουν αποδειχτεί καλύτερα στην πράξη
  - Η υποστήριξη ανάδρασης συνάφειας δεν είναι εύκολη
    - If a document is identified by the user as relevant or irrelevant, how should the query by modified ?

Τα θετικά του Boolean μοντέλου

- Προβλέψιμο, εύκολα εξηγήσιμο
- Αποτελεσματικό όταν γνωρίζεις ακριβώς τι ψάχνεις και τι περιέχει η συλλογή
- Αποδοτική υλοποίηση

Στατιστικά Μοντέλα

Κοινά χαρακτηριστικά των Στατιστικών Μοντέλων

- Έγγραφο: σάκος (bag) λέξεων
  - Bag = set that allows multiple occurences of the same element
  - So we view a document as an unordered set of words with frequencies
- Επερώτηση: Σύνολο όρων με προαιρετικά βάρη:
  - Weighted query terms: **q=<database 0.5, text 0.8, information 0.2>**
  - Unweighted query terms: **q=<database text information >**
  - No Boolean conditions specified in the query
- Απάντηση: Διατεταγμένο σύνολο συναφών εγγράφων
  - υπολογίζεται βάσει των συχνοτήτων εμφάνισης των λέξεων στα έγγραφα και στις επερωτήσεις

CS463 - Information Retrieval Systems Yannis Tzitzikas, U. of Crete 26

Στατιστικά Μοντέλα: Κρίσιμα Ερωτήματα

- Πώς να καθορίζουμε τη σπουδαιότητα ενός όρου σε ένα έγγραφο και στα πλαίσια ολόκληρης της συλλογής;
- Πώς να καθορίζουμε το βαθμό ομοιότητας μεταξύ ενός εγγράφου και μιας επερώτησης;

Information Retrieval Models

Vector Space Model

(Διανυσματικό Μοντέλο)

(το πιο διαδεδομένο μοντέλο ανάκτησης)

Διανυσματικό Μοντέλο: Εισαγωγή

- K={k_1,…,k_t} : σύνολο όλων των λέξεων ευρετηρίασης
- Κάθε έγγραφο **d_j** παριστάνεται με ένα διάνυσμα d_j = ($w_{1,j}$,…,$w_{t,j}$) όπου $w_{i,j} \in [0,1]$ (πχ $w_{i,j}$=0.3)
- Μια επερώτηση q παριστάνεται με ένα διάνυσμα q = ($w_{1,q}$,…,$w_{t,q}$) όπου πάλι $w_{i,q} \in [0,1]$
- R(d,q) εκφράζει το βαθμό ομοιότητας των διανυσμάτων d και q

Παράσταση εγγράφων στο Διανυσματικό Μοντέλο

	k_1	k_2	....	k_t
d_1	w_{11}	w_{21}	…	w_{t1}
d_2	w_{12}	w_{22}	…	w_{t2}
:	:	:		:
:	:	:		:
d_n	w_{1n}	w_{2n}	…	w_{tn}

$w_{i,j} \in [0,1]$

- K={k_1,…,k_t} : σύνολο όλων των λέξεων ευρετηρίασης
- Κάθε έγγραφο **d_j** παριστάνεται με το διάνυσμα d_j=($w_{1,j}$,…,$w_{t,j}$) όπου:
  - $w_{i,j}$ το βάρος της λέξης k_i για το κείμενο **d_j**

Βάρη Όρων: Συχνότητα όρου (tf)

- Οι πιο συχνοί όροι σε ένα έγγραφο είναι πιο σημαντικοί (υποδηλώνουν το περιεχόμενο του)
  - $freq_{ij}$ = πλήθος εμφανίσεων του όρου i στο έγγραφο j
- Κανονικοποίηση
  - $tf_{ij} = freq_{ij} / max_k \{freq_{kj}\}$
  - *όπου* $max_k \{freq_{kj}\}$ *το μεγαλύτερο πλήθος εμφανίσεων ενός όρου στο έγγραφο j*

Παράδειγμα: Έστω το έγγραφο d_2 ="a a a a b b b c c c c"

$freq_{a2} = 4,$

$tf_{a2} = 4/4=1$

$freq_{b2} = 3,$

$tf_{b2} = 3/4=0.75$

Παράδειγμα

- d_1 = { a a a b c }
- d_2 = { a a a d e }
- d_3 = { a a a f g }

- Το a λαμβάνει το μεγαλύτερο βάρος (άρα το μεγαλύτερο tf) σε κάθε έγγραφο

- Ας σκεφτούμε ολόκληρη τη συλλογή.
- Μας επιτρέπει το a να διακρίνουμε τα κείμενα;
- Αν όχι μήπως δεν θα έπρεπε να λαμβάνει το μεγαλύτερο βάρος (στο διάνυσμα του κάθε εγγράφου);
- Αν η συλλογή είχε μόνο αυτά τα 3 έγγραφα (και ήταν σταθερή) θα μπορούσαμε ακόμα και να ... αγνοήσουμε πλήρως τον όρο a από το ευρετήριο.

Βάρη Όρων: Αντίστροφη Συχνότητα Εγγράφων (Inverse Document Frequency)

- Ιδέα: Όροι που εμφανίζονται σε πολλά διαφορετικά έγγραφα έχουν μικρή διακριτική ικανότητα
- df_i = document frequency of term i
  - πλήθος εγγράφων που περιέχουν τον όρο i
- idf_i = inverse document frequency of term i := $\log_2(N/ df_i)$
  - (N: συνολικό πλήθος εγγράφων)
- **Το *idf* αποτελεί μέτρο της διακριτικής ικανότητας του όρου**
  - ο λογάριθμος ελαφραίνει το βάρος του idf σε σχέση με το tf
- Παράδειγμα:
  - Έστω N=10 και $df_{computer}$ =10, $df_{aristotle}$ =2,
  - Τότε, N/$df_{computer}$ =10/10=1, N/$df_{aristotle}$ =10/2=5
  - Τότε, $idf_{computer}$ =log(1) =0, $idf_{aristotle}$ =log(5)=2.3

TF-IDF Weighting (βάρυνση TF-IDF)

$$\begin{array}{c|cccc} & k_1 & k_2 & \ldots. & k_t \\ d_1 & w_{11} & w_{21} & \ldots & w_{t1} \\ d_2 & w_{12} & w_{22} & \ldots & w_{t2} \\ : & : & : & & : \\ : & : & : & & : \\ d_n & w_{1n} & w_{2n} & \ldots & w_{tn} \end{array}$$

$$w_{ij} = tf_{ij}\, idf_i = tf_{ij} \log_2 (N/ df_i)$$

- Ένας όρος που εμφανίζεται **συχνά** στο έγγραφο, αλλά **σπάνια** στην υπόλοιπη συλλογή, λαμβάνει **υψηλό** βάρος.
- Αν και έχουν προταθεί πολλοί άλλοι τρόποι βάρυνσης, το tf-idf δουλεύει πολύ καλά στην πράξη.

Παράδειγμα υπολογισμού TF-IDF

- Έστω το ακόλουθο έγγραφο:
  - d="A B A B C A"
- Υποθέστε ότι η συλλογή περιέχει 10.000 έγγραφα και οι συχνότητες κειμένου (document frequencies) αυτών των όρων είναι:
  - A(50), B(1300), C(250)

Τότε:

- A: tf=3/3; idf = log(10000/50)= 5.3; tf-idf=5.3
- B: tf=2/3; idf = log(10000/1300)= 2; tf-idf=1.3
- C: tf=1/3; idf = log(10000/250)= 3.7; tf-idf=1.2

Διάνυσμα Επερώτησης

- Τα διανύσματα των επερωτήσεων θεωρούνται ως έγγραφα και επίσης βαρύνονται με tf-idf
  - Μια επερώτηση δεν συγκροτείται πάντα από λίγες λέξεις. Μια επερώτηση μπορεί να είναι μια παράγραφος κειμένου (ή ένα ολόκληρο έγγραφο)
- Εναλλακτικά, ο χρήστης μπορεί να δώσει τα βάρη των όρων της επερώτησης

	k_1	k_2	....	k_t
d_1	w_{11}	w_{21}	...	w_{t1}
d_2	w_{12}	w_{22}	...	w_{t2}
:	:	:		:
:	:	:		:
d_n	w_{1n}	w_{2n}	...	w_{tn}
q	w_{1q}	w_{2q}	...	w_{tq}

$w_{i,j} \in [0,1]$

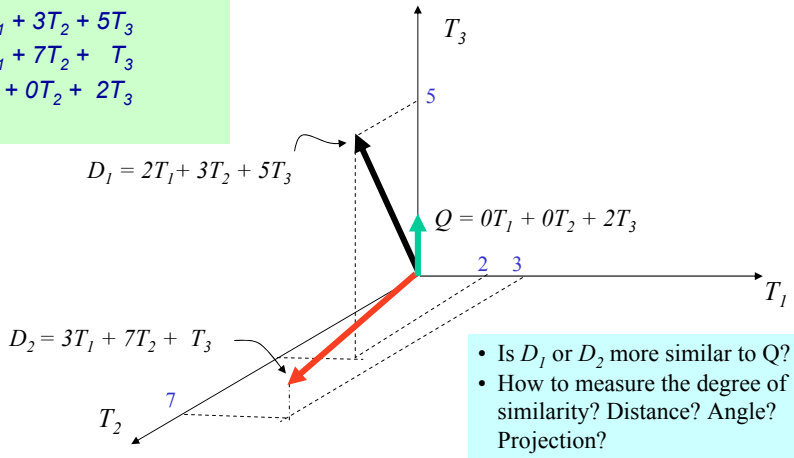
Διανυσματικό Μοντέλο:

- K={k_1,…,k_t} : σύνολο όλων των λέξεων ευρετηρίασης
- Κάθε έγγραφο $\mathbf{d_j}$ παριστάνεται με ένα διάνυσμα d_j=($w_{1,j}$,…,$w_{t,j}$) όπου $w_{i,j}$ = $\mathbf{tf_{ij}}$ $\mathbf{idf_i}$
- Μια επερώτηση q παριστάνεται με ένα διάνυσμα q=($w_{1,q}$,…,$w_{t,q}$) όπου πάλι $w_{i,q}$ = $\mathbf{tf_{iq}}$ $\mathbf{idf_i}$
- R(d,q) = ?

Διανυσματικό Μοντέλο: Μέτρο Ομοιότητας

Έστω οτι το λεξιλόγιο μας αποτελείται από 3 λέξεις T1, T2 και T3

Example:
$D_1 = 2T_1 + 3T_2 + 5T_3$
$D_2 = 3T_1 + 7T_2 + T_3$
$Q = 0T_1 + 0T_2 + 2T_3$



- Is D_1 or D_2 more similar to Q?
- How to measure the degree of similarity? Distance? Angle? Projection?

Μέτρο Ομοιότητας: **Εσωτερικό Γινόμενο** (inner product)

- Η ομοιότητα μεταξύ των διανυσμάτων d και q ορίζεται ως το εσωτερικό τους γινόμενο:

$$sim(dj,q) = \vec{dj} \cdot \vec{q} = \sum_{i=1}^{t} w_{ij} \cdot w_{iq}$$

  - όπου w_{ij} το βάρος του όρου i στο έγγραφο j και w_{iq} το βάρος του όρου i στην επερώτηση. Το πλήθος των όρων του λεξιλογίου είναι t
- Για δυαδικά (0/1) διανύσματα το εσωτερικό γινόμενο είναι ο αριθμός των matched query terms in the document (άρα το μέγεθος της τομής)
- Για βεβαρημένα διανύσματα, είναι το άθροισμα των γινομένων των βαρών των matched terms

Παράδειγμα

Binary: retrieval, database, architecture, computer, text, management, information

- d = 1, 1, 1, 0, 1, 1, 0
- q = 1, 0, 1, 0, 0, 1, 1

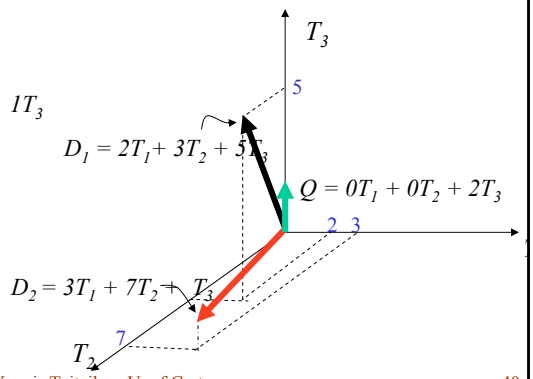
sim(d, q) = 3

Size of vector = size of vocabulary = 7
0 means corresponding term not found in document or query

Weighted:

$D_1 = 2T_1 + 3T_2 + 5T_3$ $D_2 = 3T_1 + 7T_2 + 1T_3$
$Q = 0T_1 + 0T_2 + 2T_3$

$sim(D_1, Q) = 2*0 + 3*0 + 5*2 = 10$
$sim(D_2, Q) = 3*0 + 7*0 + 1*2 = 2$



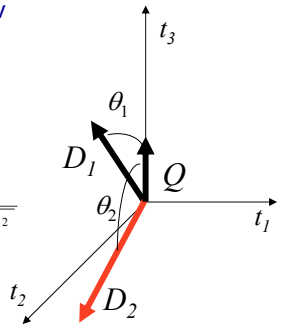
Ιδιότητες του Εσωτερικού Γινομένου

- Το εσωτερικό γινόμενο
  - δεν είναι φραγμένο (unbounded)
  - ευνοεί (μεροληπτεί) μεγάλα έγγραφα με μεγάλο πλήθος διαφορετικών όρων
  - μετρά το πλήθος των όρων που κάνουν match, αλλά αγνοεί αυτούς που δεν κάνουν match

Μέτρο Ομοιότητας Συνημίτονου (Cosine)

- Μετρά το συνημίτονο της γωνίας μεταξύ των 2 διανυσμάτων
- Εσωτερικό γινόμενο κανονικοποιημένο βάσει του μήκους των διανυσμάτων

$$\text{CosSim}(\boldsymbol{d}_j, \boldsymbol{q}) = \frac{\vec{d}_j \cdot \vec{q}}{\left|\vec{d}_j\right| \cdot \left|\vec{q}\right|} = \frac{\sum_{i=1}^{t} (w_{ij} \cdot w_{iq})}{\sqrt{\sum_{i=1}^{t} w_{ij}^2 \cdot \sum_{i=1}^{t} w_{iq}^2}}$$



$D_1 = 2T_1 + 3T_2 + 5T_3$ $\quad \text{CosSim}(D_1, Q) = 10 / \sqrt{(4+9+25)(0+0+4)} = 0.81$
$D_2 = 3T_1 + 7T_2 + 1T_3$ $\quad \text{CosSim}(D_2, Q) = 2 / \sqrt{(9+49+1)(0+0+4)} = 0.13$
$Q = 0T_1 + 0T_2 + 2T_3$

D_1 is 6 times better than D_2 using cosine similarity but only 5 times better using inner product.

Διανυσματικό Μοντέλο: Παρατηρήσεις

- Πλεονεκτήματα
  - Λαμβάνει υπόψη τις **τοπικές** (tf) και **καθολικές** (idf) συχνότητες όρων
  - Παρέχει **μερικό ταίριασμα** (partial matching) και **διατεταγμένα** αποτελέσματα
  - Τείνει να δουλεύει καλά στην πράξη, παρά τις αδυναμίες του
  - Αποδοτική υλοποίηση για μεγάλες συλλογές εγγράφων
- Αδυναμίες
  - Απουσία Σημασιολογίας (π.χ. σημασίας λέξεων)
  - Απουσία Συντακτικής Πληροφορίας (π.χ. δομή φράσης, σειρά λέξεων, εγγύτητα λέξεων)
  - Υπόθεση Ανεξαρτησίας Όρων (π.χ. αγνοεί τα συνώνυμα)
  - Έλλειψη ελέγχου ala Boolean model (π.χ. δεν μπορούμε να απαιτήσουμε την παρουσία ενός όρου στο έγγραφο)
    - Given a two-term query q="A B", may prefer a document containing A frequently but not B, over a document that contains both A and B but both less frequently

Περίληψη του Διανυσματικού Μοντέλου

- $K=\{k_1,\ldots,k_t\}$: σύνολο όλων των λέξεων ευρετηρίασης
- Κάθε έγγραφο $\mathbf{d_j}$ παριστάνεται με το διάνυσμα $d_j=(w_{1,j},\ldots,w_{t,j})$ όπου $w_{ij} = tf_{ij}\, idf_i = tf_{ij} \log_2 (N/ df_i)$
- Μια επερώτηση q παριστάνεται με το διάνυσμα $q=(w_{1,q},\ldots,w_{t,q})$ όπου $w_{iq} = tf_{iq}\, idf_i = tf_{iq} \log_2 (N/ df_i)$
- $R(\mathbf{d_j},\mathbf{q}) = \text{CosSim}(\mathbf{d_j}, \mathbf{q}) = \dfrac{\vec{d}_j \cdot \vec{q}}{|\vec{d}_j| \cdot |\vec{q}|} = \dfrac{\sum_{i=1}^{t} (w_{ij} \cdot w_{iq})}{\sqrt{\sum_{i=1}^{t} w_{ij}^2 \cdot \sum_{i=1}^{t} w_{iq}^2}}$

Υπολογισμός του βαθμού συνάφειας
Απλοϊκή Υλοποίηση

1) Φτιάξε το *tf-idf* διάνυσμα για κάθε έγγραφο $\mathbf{d}_j$ ths συλλογής (έστω V το λεξιλόγιο)

2) Φτιάξε το *tf-idf* διάνυσμα **q** της επερώτησης

3) Για κάθε έγγραφο d_j του D
   Υπολόγισε το σκορ $s_j = \text{cosSim}(d_j, q)$

4) Διέταξε τα έγγραφα σε φθίνουσα σειρά

5) Παρουσίασε τα έγγραφα στο χρήστη

Χρονική πολυπλοκότητα του βήματος (3): O(|V|·|D|)

Πολύ ακριβό αν τα V και D είναι μεγάλα!

|V| = 10,000; |D| = 100,000; |V|·|D| = 1,000,000,000

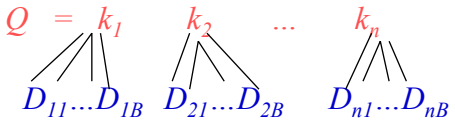
Υπολογισμός του βαθμού συνάφειας
Καλύτερη (γρηγορότερη) Υλοποίηση

- Ένας όρος που δεν εμφανίζεται **και** στην επερώτηση **και** στο έγγραφο δεν επηρεάζει το βαθμό ομοιότητας συνημίτονου
  - Το γινόμενο των βαρών είναι 0 και άρα δεν συνεισφέρει στο εσωτερικό γινόμενο
- Συνήθως η επερώτηση είναι μικρή, άρα το διάνυσμα της είναι εξαιρετικά «αραιό»

- => Μπορούμε να χρησιμοποιήσουμε ένα ευρετήριο ώστε να υπολογίσουμε το βαθμό ομοιότητας μόνο εκείνων των εγγράφων που περιέχουν τουλάχιστον έναν όρο της επερώτησης.

3) Για κάθε έγγραφο d_j του D Υπολόγισε το σκορ $s_j = \text{cosSim}(d_j, q)$	Απλοϊκό
3') Για κάθε έγγραφο d_j **που περιέχει τουλάχιστον έναν όρο του query** Υπολόγισε το σκορ $s_j = \text{cosSim}(d_j, q)$	Καλύτερο

Υπολογισμός του βαθμού συνάφειας
Καλύτερη (γρηγορότερη) Υλοποίηση (II)



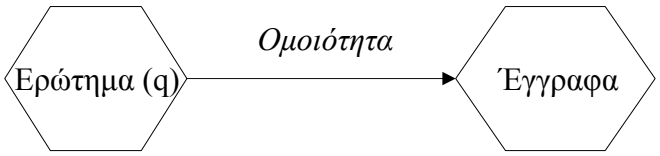
- Ας υποθέσουμε ότι ένας όρος της επερώτησης εμφανίζεται σε *B* έγγραφα
- Τότε η χρονική πολυπλοκότητα είναι O(|*Q*| *B*)

- Το κόστος αυτό είναι συνήθως πολύ μικρότερο του κόστους του απλοϊκού τρόπου (που είχε πολυπλοκότητα O(|V||D|), διότι:
  - |*Q*| << |*V*|, δηλαδή ο αριθμός των λέξεων στην επερώτησης είναι πολύ μικρότερος του συνολικού αριθμού των λέξεων, και
  - *B* << |D|, δηλαδή το πλήθος των εγγράφων που έχουν μια λέξη είναι πολύ μικρότερο του πλήθους των εγγράφων της συλλογής.

Μέθοδοι Υπολογισμού Ομοιότητας (υπενθύμιση)

Περαιτέρω συζήτηση για το διανυσματικό μοντέλο

Μέθοδοι υπολογισμού ομοιότητας: μετρούν το βαθμό ομοιότητας μεταξύ ενός ερωτήματος και των εγγράφων.



Σημειώστε τη διαφορά με τις μεθόδους που υποστηρίζουν μόνο επακριβή αναζήτηση (exact match). Για παράδειγμα, στο Boolean μοντέλο ένα κείμενο χαρακτηρίζεται είτε σχετικό είτε άσχετο ως προς το ερώτημα.

Ομοιότητα Εγγράφων

Πρόβλημα: Πόσο μοιάζουν δύο έγγραφα;

Ιδέα: Όσο περισσότερες κοινές λέξεις έχουν δύο κείμενα, τόσο περισσότερο μοιάζουν. (boolean)

Παράδειγμα:

Έστω τα ακόλουθα έγγραφα. Πόσο μοιάζουν μεταξύ τους;

d_1 *ant ant bee*
d_2 *dog bee dog hog dog ant dog*
d_3 *cat gnu dog eel fox*

Διανυσματικό Μοντέλο: δυαδικά βάρη

Ο χώρος των όρων

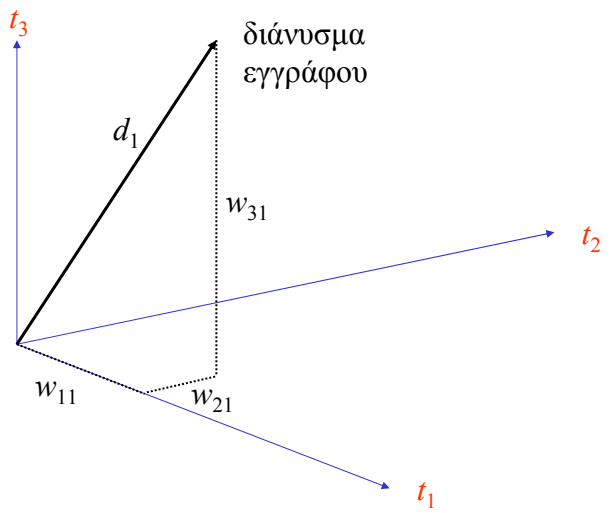
Αποτελείται από *m* διαστάσεις, όπου *m* είναι ο αριθμός των μοναδικών όρων που χρησιμοποιούνται στα έγγραφα.

Διάνυσμα

Το έγγραφο *dj* αναπαρίσταται ως διάνυσμα με συντεταγμένες *wij* (όρος *i*, έγγραφο *j*).

$w_{ij} = 1$ αν ο *i*-οστός όρος εμφανίζεται στο *dj*
$w_{ij} = 0$ διαφορετικά

Διανυσματικό Μοντέλο: δυαδικά βάρη



Διανυσματικό Μοντέλο: δυαδικά βάρη

document	text	terms
d_1	*ant ant bee*	*ant bee*
d_2	*dog bee dog hog dog ant dog*	*ant bee dog hog*
d_3	*cat gnu dog eel fox*	*cat dog eel fox gnu*

	ant	bee	cat	dog	eel	fox	gnu	hog
d_1	1	1						
d_2	1	1		1				1
d_3			1	1	1	1	1	

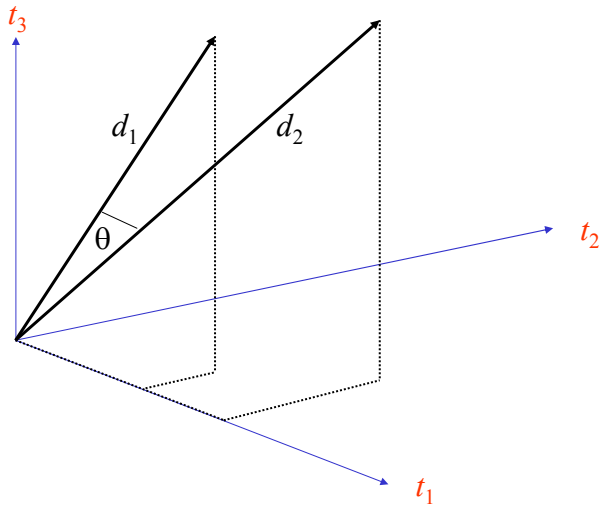
3 διανύσματα
8 διαστάσεις

$w_{ij} = 1$ αν το d_j περιέχει τον i-οστό όρο

Ομοιότητα Εγγράφων

Η ομοιότητα μεταξύ δύο εγγράφων υπολογίζεται με βάση τη γωνία που σχηματίζεται μεταξύ των δύο αντίστοιχων διανυσμάτων.

Πιο συγκεκριμένα, χρησιμοποιείται το συνημίτονο της γωνίας θ.



Μαθηματικές Έννοιες

$\mathbf{x} = (x_1, x_2, x_3, ..., x_n)$ διάνυσμα στο χώρο των n διαστάσεων

Μέτρο του $\mathbf{x}$ δίνεται με βάση το Πυθαγόρειο θεώρημα

$$|\mathbf{x}|^2 = x_1^2 + x_2^2 + x_3^2 + ... + x_n^2$$

Αν $\mathbf{x}_1$ και $\mathbf{x}_2$ είναι διανύσματα:

Εσωτερικό Γινόμενο (dot product) δίνεται από:

$$\mathbf{x}_1.\mathbf{x}_2 = x_{11}x_{21} + x_{12}x_{22} + x_{13}x_{23} + ... + x_{1n}x_{2n}$$

Συνημίτονο γωνίας μεταξύ των διανυσμάτων $\mathbf{x}_1$ and $\mathbf{x}_2$:

$$\cos(\theta) = \frac{\mathbf{x}_1.\mathbf{x}_2}{|\mathbf{x}_1|\,|\mathbf{x}_2|}$$

Παράδειγμα: δυαδικά βάρη

document	text	terms
d_1	*ant ant bee*	*ant bee*
d_2	*dog bee dog hog dog ant dog*	*ant bee dog hog*
d_3	*cat gnu dog eel fox*	*cat dog eel fox gnu*

	ant	bee	cat	dog	eel	fox	gnu	hog	*length*
d_1	1	1							√2
d_2	1	1		1				1	√4
d_3			1	1	1	1	1		√5

Παράδειγμα: δυαδικά βάρη

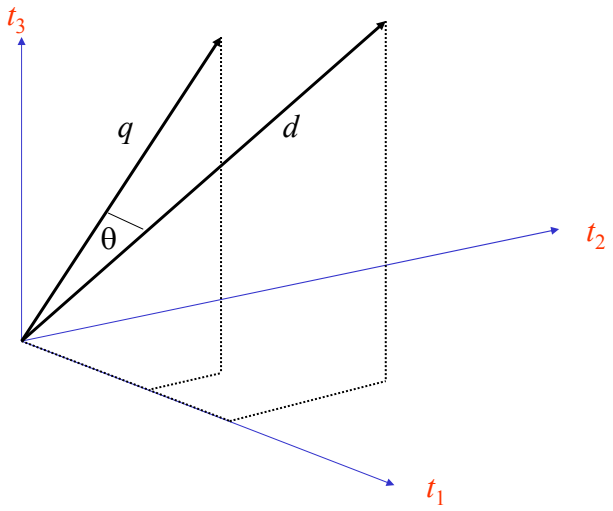
Πίνακας ομοιότητα εγγράφων

	d_1	d_2	d_3
d_1	1	0.71	0
d_2	0.71	1	0.22
d_3	0	0.22	1

Ομοιότητα Ερωτήματος-Εγγράφου

Η ομοιότητα μεταξύ ενός ερωτήματος *q* και ενός εγγράφου *d* προσδιορίζεται πάλι με το συνημίτονο της μεταξύ τους γωνίας.

Στην πράξη, ένα ερώτημα έχει πολύ μικρότερο μήκος από ένα έγγραφο



Ομοιότητα Ερωτήματος-Εγγράφου

ερώτημα		
q	*ant dog*	
έγγραφα	περιεχόμενα	διαφορετικοί όροι
d_1	*ant ant bee*	*ant bee*
d_2	*dog bee dog hog dog ant dog*	*ant bee dog hog*
d_3	*cat gnu dog eel fox*	*cat dog eel fox gnu*

	ant	bee	cat	dog	eel	fox	gnu	hog
q	1			1				
d_1	1	1						
d_2	1	1		1				1
d_3			1	1	1	1	1	

Ο πίνακας έχει μηδενικά στις υπόλοιπες θέσεις.

Ομοιότητα Ερωτήματος-Εγγράφου

	d_1	d_2	d_3
q	1/2 0.5	1/√2 0.71	1/√10 0.32

Με βάση το ερώτημα και τα έγγραφα του παραδείγματος το έγγραφο που χαρακτηρίζεται περισσότερο σχετικό ως προς q είναι το d_2, μετά το d_1 και τέλος το d_3.

Χρήση του Διανυσματικού Μοντέλου

Ερώτημα με κατώφλι (περιοχής)

Για το ερώτημα q το σύστημα επιστρέφει όλα τα έγγραφα που έχουν βαθμό ομοιότητας μεγαλύτερο από κάποιο κατώφλι (π.χ., > 0.6).

Ερώτημα top-*k*

Για το ερώτημα q το σύστημα επιστρέφει τα k έγγραφα που έχουν το μεγαλύτερο βαθμό ομοιότητας ως προς το q.

Γενίκευση: μη δυαδικά βάρη

Το Διανυσματικό Μοντέλο βελτιώνεται με την εισαγωγή επιπλέον πληροφορίας για τον προσδιορισμό των βαρών w_{ij}

- Μερικές από τις πληροφορίες αυτές είναι οι εξής:
- Το πλήθος των εγγράφων που περιέχουν τον όρο,
- Πόσες φορές εμφανίζεται ένας όρος σε ένα έγγραφο,
- Το μήκος των εγγράφων.

Διανυσματικό Μοντέλο: μη δυαδικά βάρη

Ο χώρος των όρων

Αποτελείται από *m* διαστάσεις, όπου *m* είναι ο αριθμός των μοναδικών όρων που χρησιμοποιούνται στα έγγραφα.

Διάνυσμα

Το έγγραφο d_j αναπαρίσταται ως διάνυσμα με συντεταγμένες w_{ij} (όρος *i*, έγγραφο *j*).

$w_{ij} > 0$ αν ο *i*-οστός όρος εμφανίζεται στο d_j
$w_{ij} = 0$ διαφορετικά

Η τιμή w_{ij} ορίζεται ως το βάρος του *i*-οστού όρου στο *j*-οστό έγγραφο.

Προσδιορισμός Βαρών

Η **γενική μορφή** προσδιορισμού των βαρών *wij* είναι:

$$w_{ij} = TF_{ij} \text{ x } IDF_i$$

Όπου TF_{ij} είναι ένας παράγοντας που εξαρτάται από τη συχνότητα εμφάνισης του *i*-οστού όρου στο *j*-οστό έγγραφο.

Ο παράγοντας IDF_i εξαρτάται από το πλήθος των εγγράφων που περιέχουν τον όρο t_i.

Προσδιορισμός Βαρών

Εναλλακτικές μορφές του $TF_{t,d}$

περιγραφή	$TF_{t,d}$
δυαδικός σχηματισμός	1 ή 0
συνήθης σχηματισμός	$f_{t,d}$
λογαριθμικός σχηματισμός	$1 + \ln(f_{t,d})$
κανονικοποιημένος σχηματισμός	$\frac{f_{t,d}}{\max_x\{f_{x,d}\}}$
εναλλακτικός κανονικοποιημένος σχηματισμός Το C είναι μία σταθερά η οποία αν λάβει τιμές μεταξύ 0.3 και 0.5 έχει τα καλύτερα αποτελέσματα	$C + (1 - C) \cdot \frac{f_{t,d}}{\max_x\{f_{x,d}\}}$

Προσδιορισμός Βαρών

Εναλλακτικές μορφές του IDF_t

περιγραφή	IDF_t
δυαδικός σχηματισμός	1
1ος λογαριθμικός σχηματισμός	$\ln\left(\frac{N}{n_t}\right)$
2ος λογαριθμικός σχηματισμός	$\ln\left(1+\frac{N}{n_t}\right)$
3ος λογαριθμικός σχηματισμός	$\frac{\ln(N/n_t)}{\ln(N)}$
υπερβολικός σχηματισμός	$\frac{1}{n_t}$
1ος κανονικοποιημένος σχηματισμός	$\ln\left(1+\frac{\max_x\{n_x\}}{n_t}\right)$
2ος κανονικοποιημένος σχηματισμός	$\ln\left(\frac{N-n_t}{n_t}\right)$

Προσδιορισμός Βαρών

Εναλλακτικές μορφές του L_d, L_q Μέγεθος αρχείου, ερώτησης

περιγραφή	L_d
μοναδιαίος σχηματισμός	1
διανυσματικός σχηματισμός	$\sqrt{\sum_{x\in\mathcal{T}_d} w_{x,d}^2}$
1ος προσεγγιστικός σχηματισμός	$\lvert\mathcal{T}_d\rvert$
2ος προσεγγιστικός σχηματισμός	$\sqrt{\lvert\mathcal{T}_d\rvert}$
3ος προσεγγιστικός σχηματισμός	$\log_2(\lvert\mathcal{T}_d\rvert)$
4ος προσεγγιστικός σχηματισμός	f_d
5ος προσεγγιστικός σχηματισμός	$\sqrt{f_d}$

Προσδιορισμός Βαρών

Εναλλακτικές μορφές υπολογισμού ομοιότητας

περιγραφή	$S_{vector}(q,d)$
εσωτερικό γινόμενο	$\sum_{t\in\mathcal{T}_{q,d}}(w_{t,q}\cdot w_{t,d})$
μέθοδος συνημιτόνου	$\frac{1}{L_q\cdot L_d}\cdot\sum_{t\in\mathcal{T}_{q,d}}(w_{t,q}\cdot w_{t,d})$
απλή πιθανοτική μετρική	$\sum_{t\in\mathcal{T}_{q,d}}(C+IDF_t)$
σύνθετη πιθανοτική μετρική	$\sum_{t\in\mathcal{T}_{q,d}}(C+IDF_t)\cdot TF_{t,d}$
εναλλακτικό εσωτερικό γινόμενο	$\sum_{t\in\mathcal{T}_{q,d}}\frac{w_{t,d}}{L_d}$
μέθοδος Dice	$\frac{2}{L_q^2+L_d^2}\cdot\sum_{t\in\mathcal{T}_{q,d}}(w_{t,q}\cdot w_{t,d})$
μέθοδος Jaccard	$\frac{\sum_{t\in\mathcal{T}_{q,d}}(w_{t,q}\cdot w_{t,d})}{L_q^2+L_d^2-\sum_{t\in\mathcal{T}_{q,d}}(w_{t,q}\cdot w_{t,d})}$
μέθοδος επικάλυψης	$\frac{\sum_{t\in\mathcal{T}_{q,d}}(w_{t,q}\cdot w_{t,d})}{\min(L_q^2,L_d^2)}$

Ένα Παράδειγμα Συγκεκριμένου Μοντέλου

περιγραφή	έκφραση
συνάρτηση ομοιότητας	$S_{vector}(q,d)=\frac{1}{L_q\cdot L_d}\cdot\sum_{t\in\mathcal{T}_{q,d}}(w_{t,q}\cdot w_{t,d})$
υπολογισμός IDF_t	$IDF_t=\ln\left(1+\frac{N}{n_t}\right)$
υπολογισμός $w_{t,d}$	$w_{t,d}=TF_{t,d}$
υπολογισμός $TF_{t,d}$	$TF_{t,d}=1+\ln(f_{t,d})$
υπολογισμός L_d	$L_d=\sqrt{\sum_{x\in\mathcal{T}_d}w_{x,d}^2}$
υπολογισμός $w_{t,q}$	$w_{t,q}=TF_{t,q}\cdot IDF_t$
υπολογισμός $TF_{t,q}$	$TF_{t,q}=1+\ln(f_{t,q})$
υπολογισμός L_q	$L_q=1$

Παράδειγμα Υπολογισμού Ομοιότητας

Έστω το ερώτημα *q* = {κομήτης, Χάλεϋ} που αποτελείται από δύο όρους

t_1 = κομήτης και t_2 = Χάλλεϋ

Ενδιαφερόμαστε για το βαθμό ομοιότητας του ερωτήματος *q* με καθένα από τα έγγραφα της συλλογής εγγράφων *D* …

Παράδειγμα Υπολογισμού Ομοιότητας

Συλλογή εγγράφων

d_1 : Ο κομήτης του Χάλλεϋ μας επισκέπτεται περίπου κάθε εβδομήντα έξι χρόνια.
d_2 : Ο κομήτης του Χάλλεϋ πήρε το όνομά του από τον αστρονόμο Έντμοντ Χάλλεϋ.
d_3 : Ένας κομήτης διαγράφει ελλειπτική τροχιά.
d_4 : Ο πλανήτης Άρης έχει δύο φυσικούς δορυφόρους, το Δείμο και το Φόβο.
d_5 : Ο πλανήτης Δίας έχει 63 γνωστούς φυσικούς δορυφόρους.
d_6 : Ένας κομήτης έχει μικρότερη διάμετρο από ότι ένας πλανήτης.
d_7 : Ο Άρης είναι ένας πλανήτης του ηλιακού μας συστήματος.

Information Retrieval Models

Probabilistic Model

Κλασικά Μοντέλα Ανάκτησης

Τρία είναι τα, λεγόμενα, κλασικά μοντέλα ανάκτησης:

Λογικό (Boolean) που βασίζεται στη Θεωρία Συνόλων

Διανυσματικό (Vector) που βασίζεται στη Γραμμική Άλγεβρα

Πιθανοκρατικό (Probabilistic) που βασίζεται στη Θεωρία Πιθανοτήτων

Το Διανυσματικό και το Πιθανοκρατικό έχουν σημαντική επικάλυψη αν και στηρίζονται σε εντελώς διαφορετικές θεωρίες.

Πιθανοκρατικό Μοντέλο

- Στόχος: να ορίσουμε το IR πρόβλημα σε πιθανοτικό πλαίσιο
- Για κάθε ερώτηση (επερώτημα) υπάρχει ένα **ιδανικό σύνολο κειμένων** (R) που το ικανοποιεί.
- Επεξεργαζόμαστε την ερώτηση με βάση *τις ιδιότητες αυτού του συνόλου*.
- Ποιες είναι όμως αυτές οι ιδιότητες;
- Αρχικά γίνεται μία πρόβλεψη και στη συνέχεια η πρόβλεψη βελτιώνεται.

Πιθανοκρατικό Μοντέλο

- Αρχικά επιστρέφεται ένα σύνολο εγγράφων.
- Ο χρήστης εξετάζει τα κείμενα αναζητώντας σχετικά κείμενα.
- Το σύστημα IR χρησιμοποιεί το feedback του χρήστη ώστε να προσδιοριστεί καλύτερα το ιδανικό σύνολο κειμένων.
- Η διαδικασία επαναλαμβάνεται.
- Η περιγραφή του ιδανικού συνόλου κειμένων πραγματοποιείται πιθανοτικά.

Ανεξάρτητες Μεταβλητές και Πιθανότητα υπό Συνθήκη

Έστω *a,* και *b* δύο γεγονότα με πιθανότητες να συμβούν $P(a)$ και $P(b)$ αντίστοιχα.

Ανεξάρτητα Γεγονότα

Τα γεγονότα *a* και *b* είναι ανεξάρτητα αν και μόνο αν:

$$P(a \cap b) = P(b)\,P(a)$$

Υπό Συνθήκη Πιθανότητα

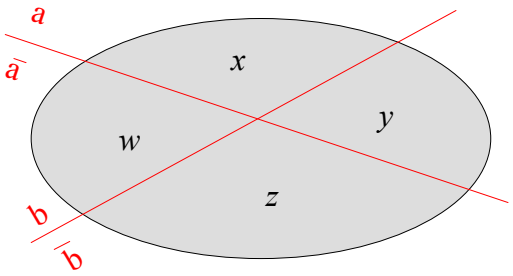
$P(a \mid b)$ είναι η πιθανότητα του *a* δεδομένου του *b*.

Τα γεγονότα $a_1, ..., a_n$ καλούνται υπό συνθήκη ανεξάρτητα αν και μόνο αν:

$$P(a_i \mid a_j) = P(a_i) \quad \text{για όλα τα } i \text{ και } j$$

Παράδειγμα I

$\overline{a}$ είναι η άρνηση του γεγονότος *a*



$$P(a) = x + y$$

$$P(b) = w + x$$

$$P(a \mid b) = x / (w + x)$$

$$\boldsymbol{P(a \mid b)\,P(b) = P(a \cap b) = P(b \mid a)\,P(a)}$$

Παράδειγμα ΙΙ

Ανεξάρτητα γεγονότα

Έστω a και b οι τιμές που φέρνουν δύο ίδια ζάρια. Ισχύει:

$P(a=5 \mid b=3) = P(a=5) = 1/6$

Μη ανεξάρτητα

Έστω a και b οι τιμές που φέρνουν δύο ίδια ζάρια και t το άθροισμά τους. Τότε ισχύει:

$t = a + b$

$P(t=8 \mid a=2) = 1/6$
$P(t=8 \mid a=1) = 0$

Θεώρημα του Bayes

Έστω a και b δύο γεγονότα.

$P(a \mid b)$ είναι η πιθανότητα να συμβεί το γεγονός a δεδομένου ότι έχει συμβεί το γεγονός b.

Θεώρημα Bayes

$$P(a \mid b) = \frac{P(b \mid a)\, P(a)}{P(b)}$$

Ισχύει επίσης ότι:

$P(a \mid b)\, P(b) = P(a \cap b) = P(b \mid a)\, P(a)$

Θεώρημα Bayes: παράδειγμα

Example

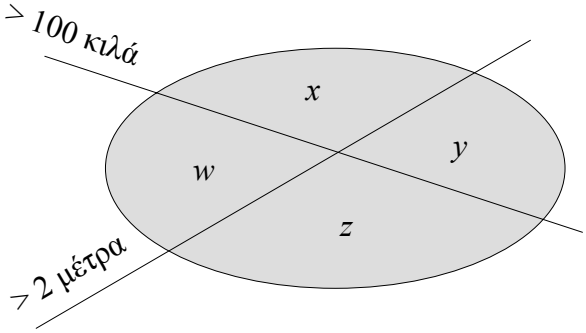
a βάρος πάνω από 100 κιλά

b ύψος πάνω από 2 μέτρα.

$P(a \mid b) = x / (w+x) = x / P(b)$

$P(b \mid a) = x / (x+y) = x / P(a)$

$x = P(a \cap b)$



Αρχή Πιθανοκρατικής Κατάταξης
Probabilistic Ranking Principle (PRP)

"If a reference retrieval system's response to each request is a ranking of the documents in the collections in order of decreasing **probability of usefulness** to the user who submitted the request, where the probabilities are estimated as accurately a possible on the basis of whatever data is made available to the system for this purpose, then the overall effectiveness of the system to its users will be the best that is obtainable on the basis of that data."

Εάν η απάντηση ενός συστήματος ανάκτησης σε κάθε ερώτημα είναι μία λίστα εγγράφων ταξινομημένη με φθίνουσα διάταξη ως προς την **πιθανότητα σχετικότητας** του κάθε εγγράφου ως προς το χρήστη, όπου οι πιθανότητες υπολογίζονται όσο γίνεται ακριβέστερα με βάση τα δεδομένα που είναι διαθέσιμα, η συνολική αποτελεσματικότητα του συστήματος θα είναι η καλύτερη δυνατή.

W.S. Cooper

Πιθανοκρατική Βαθμολόγηση

"Για ένα δεδομένο ερώτημα, εάν γνωρίζουμε κάποια από τα σχετικά έγγραφα, οι **όροι** που εμφανίζονται σε αυτά θα πρέπει να έχουν μεγαλύτερη **βαρύτητα** κατά την αναζήτηση άλλων σχετικών εγγράφων.

Κάνοντας διάφορες παραδοχές σχετικά με την κατανομή των όρων και χρησιμοποιώντας το θεώρημα του Bayes είναι δυνατόν να υπολογίσουμε τα βάρη αυτά."

Van Rijsbergen

Βασικές Έννοιες

- Η πιθανότητα ένα έγγραφο να είναι σχετικό ως προς το ερώτημα θεωρείται ότι εξαρτάται μόνο από τους όρους που περιέχονται στο έγγραφο και από τους όρους που περιέχονται στο ερώτημα.

- Η σχετικότητα ενός εγγράφου *d* ως προς το ερώτημα *q* δεν εξαρτάται από τη σχετικότητα άλλων εγγράφων της συλλογής.

- Για κάποιο ερώτημα *q* το σύνολο των σχετικών εγγράφων ***R*** είναι το **ιδανικό σύνολο** που μπορούμε να έχουμε ως απάντηση.

Βασικές Έννοιες

Για ένα ερώτημα q και ένα έγγραφο d το πιθανοκρατικό μοντέλο χρειάζεται μία εκτίμηση για την πιθανότητα $P(R \mid d)$ που δηλώνει την πιθανότητα το έγγραφο d να είναι σχετικό ως προς το ερώτημα.

P(R|d) πιθανότητα το έγγραφο να είναι σχετικό με το ερώτημα

$P(\overline{R}|d)$ πιθανότητα το έγγραφο να μην είναι σχετικό με το ερώτημα

Μέτρο Ομοιότητας (odds of being relevant to q):

$S(q, d)$, ομοιότητα του εγγράφου d ως προς το ερώτημα q:

$$\frac{\text{πιθανότητα } d \text{ σχετικό}}{\text{πιθανότητα } d \text{ μη σχετικό}} = \frac{P(R \mid d)}{P(\overline{R} \mid d)}$$

Οι τιμές της $S(\ )$ μπορεί να είναι από πολύ μικρές έως πολύ μεγάλες και για αυτό χρησιμοποιείται συνήθως ο λογάριθμος για την άμβλυνση των διαφορών.

Βασικές Έννοιες

$$S(q, d) = \frac{P(R \mid d)}{P(\overline{R} \mid d)}$$

$$= \frac{P(d \mid R)\, P(R)}{P(d \mid \overline{R})\, P(\overline{R})} \qquad \text{θεώρημα Bayes}$$

$P(d \mid R)$ είναι η πιθανότητα να διαλέξουμε τυχαία το έγγραφο d από τη συλλογή των σχετικών με την ερώτηση εγγράφων R.

$$\frac{P(d \mid R)\, \cancel{P(R)}}{P(d \mid \overline{R})\, \cancel{P(\overline{R})}}$$

Ίδια για όλα τα έγγραφα της συλλογής (έστω μια σταθερά k)

Άρα πρέπει να εκτιμήσουμε/υπολογίσουμε αυτές τις πιθανότητες

Πως; Κοιτάμε τους όρους (terms) που εμφανίζονται στο d

Βασικές Έννοιες

$$\frac{P(d \mid R)\ \cancel{P(R)}}{P(d \mid \bar{R})\ \cancel{P(\bar{R})}}$$

P(d | R): Πιθανότητα να επιλέξουμε το έγγραφο d από τα σχετικά με την ερώτηση

Θα χρησιμοποιήσουμε τους όρους k_i που έχει το έγγραφο d για να την υπολογίσουμε

Βασικές Έννοιες

Ανάκτηση Δυαδικής Ανεξαρτησίας

Binary Independence Retrieval (BIR)

Τα βάρη των όρων είναι δυαδικά και οι όροι είναι ανεξάρτητοι μεταξύ τους (η παρουσία ή μη κάποιου όρου δεν επηρεάζει τους υπόλοιπους).

Το βάρος ενός όρου σε ένα έγγραφο είναι είτε 1 (αν ο όρος περιέχεται στο έγγραφο) είτε 0 (σε διαφορετική περίπτωση).

Όπως και στο Λογικό αλλά και στο Διανυσματικό μοντέλο, η σχετικότητα ενός εγγράφου καθορίζεται από τους όρους που περιέχονται σε αυτό.

Naïve Bayes

Έστω **x** = (x_1, x_2, ... x_n) το διάνυσμα του εγγράφου *d* όπου
x_i = 1 αν ο i-οστός όρος περιέχεται στο έγγραφο, x_i = 0 διαφορετικά.

Η εκτίμηση της πιθανότητας $P(d \mid R)$ γίνεται χρησιμοποιώντας την πιθανότητα $P(\mathbf{x} \mid R)$

Εάν οι όροι είναι ανεξάρτητοι τότε:

$\boldsymbol{P(x \mid R)} = P(x_1 \cap R)\, P(x_2 \cap R) \ldots P(x_n \cap R)$

$= P(x_1 \mid R)\, P(x_2 \mid R) \ldots P(x_n \mid R)$

$= \prod \boldsymbol{P(x_i \mid R)}$

$\boldsymbol{P(x_i \mid R)}$ είναι η πιθανότητα ο όρος x_i να βρίσκεται σε ένα έγγραφο που επιλέγεται τυχαία από το ιδανικό σύνολο *R*.

Αντίστοιχα $P(\bar{x}_i \mid R)$

Το μοντέλο αυτό είναι γνωστό και ως **Naive Bayes**

Συνάρτηση Ομοιότητας

$$S(q, d) = k \frac{\prod P(x_i \mid R)}{\prod P(x_i \mid \bar{R})}$$

Αφού το κάθε x_i είναι 0 ή 1 έχουμε:

$$S = k \prod_{x_i = 1} \frac{P(x_{i=1} \mid R)}{P(x_{i=1} \mid \bar{R})} \prod_{x_i = 0} \frac{P(x_{i=0} \mid R)}{P(x_{i=0} \mid \bar{R})}$$

Το σπάμε: όροι που το x_i είναι 1 και όροι που το x_i είναι 0

Συνάρτηση Ομοιότητας

Για τους όρους που εμφανίζονται στο ερώτημα θέτουμε:

$p_i = P(x_i = 1 \mid R)$ p_i πιθανότητα ότι ένα έγγραφο που επιλέγεται από το ιδανικό σύνολο έχει τον όρο x_i

$r_i = P(x_i = 1 \mid \overline{R})$ r_i το ίδιο για το μη ιδανικό

Για τους όρους που δεν εμφανίζονται στο ερώτημα έστω:

$p_i = r_i$

όροι με $q_i = 0$ είναι ίσοι με $p_i/r_i = 1$

$$S = k \prod_{x_i = q_i = 1} \frac{p_i}{r_i} \prod_{x_i = 0,\, q_i = 1} \frac{1 - p_i}{1 - r_i}$$

Πολλαπλασιάζουμε το δεξί γινόμενο με τους όρους που υπάρχουν στο έγγραφο και διαιρούμε το αριστερό γινόμενο με τον ίδιο όρο

$$= k \prod_{x_i = q_i = 1} \frac{p_i(1 - r_i)}{r_i(1 - p_i)} \prod_{q_i = 1} \frac{1 - p_i}{1 - r_i}$$

σταθερή ποσότητα για δεδομένο ερώτημα (ανεξάρτητη του εγγράφου)

Συνάρτηση Ομοιότητας

Με λογαρίθμηση της σχέσης και αγνοώντας σταθερούς παράγοντες η συνάρτηση ομοιότητας $S_{prob}(q,d)$ παίρνει τη μορφή:

$$S_{prob}(q,d) = \log\left(S(q,d)\right)$$

$$\boxed{S_{prob}(q,d) = \sum_i \log \frac{p_i \cdot (1 - r_i)}{r_i \cdot (1 - p_i)}}$$

Όπου η άθροιση αφορά στους όρους που βρίσκονται και στο ερώτημα και στο έγγραφο.

Σχέση με το Διανυσματικό Μοντέλο

Στο Διανυσματικό μοντέλο ανάκτησης θεωρήστε ότι η *i*-οστή συνιστώσα του διανύσματος ενός εγγράφου (βάρος) ισούται με την ποσότητα

$$\log \frac{p_i \cdot (1 - r_i)}{r_i \cdot (1 - p_i)}$$

ενώ το διάνυσμα του ερωτήματος *q* ισούται με άσσους για τους όρους που ανήκουν στο ερώτημα και μηδενικά διαφορετικά.

Τότε, η συνάρτηση ομοιότητας $S_{prob}(q,d)$ ισούται με το εσωτερικό γινόμενο των δύο διανυσμάτων.

Αρχική Εκτίμηση των $P(x_i \mid R)$

Αρχικά θέτουμε τιμές στις πιθανότητες :

$$p_i = P(x_i \mid R) = c$$

p_i πιθανότητα ότι ένα έγγραφο που επιλέγεται από το ιδανικό σύνολο έχει τον όρο x_i

$$r_i = P(x_i \mid \overline{R}) = n_i / N$$

r_i το ίδιο για το μη ιδανικό

όπου:

c είναι μία τυχαία σταθερά (π.χ., 0.5) ίδια για όλους τους όρους

η κατανομή των όρων ανάμεσα στα μη σχετικά ακολουθεί την κατανομή που ακολουθεί σε όλη τη συλλογή – δεν επηρεάζει την επιλογή

n_i είναι το πλήθος των εγγράφων που περιέχουν τον i-οστό όρο

N πλήθος εγγράφων συλλογής

Προσαρμογή Τιμών των $P(x_i \mid R)$

Είναι προφανές ότι η αυθαίρετη ανάθεση τιμών δεν μπορεί να οδηγεί πάντα σε ικανοποιητικά αποτελέσματα. Για τη βελτίωση της ποιότητας των αποτελεσμάτων οι πρώτες εφαρμογές του Πιθανοκρατικού μοντέλου χρειάζονταν την παρέμβαση του χρήστη για την αναπροσαρμογή των τιμών.

Εναλλακτικά μπορεί να χρησιμοποιηθεί και αυτοματοποιημένος τρόπος. Αρχικά εκτελείται το ερώτημα με τις αρχικές εκτιμήσεις. Επιλέγονται τα k καλύτερα έγγραφα. Έστω k_i ο αριθμός των εγγράφων που περιέχουν τον i-οστό όρο. Θέτουμε:

$$p_i = P(x_i \mid R) = k_i / k$$

$$r_i = P(x_i \mid \bar{R}) = (n_i - k_i) / (N - k)$$

Πλεονεκτήματα-Μειονεκτήματα

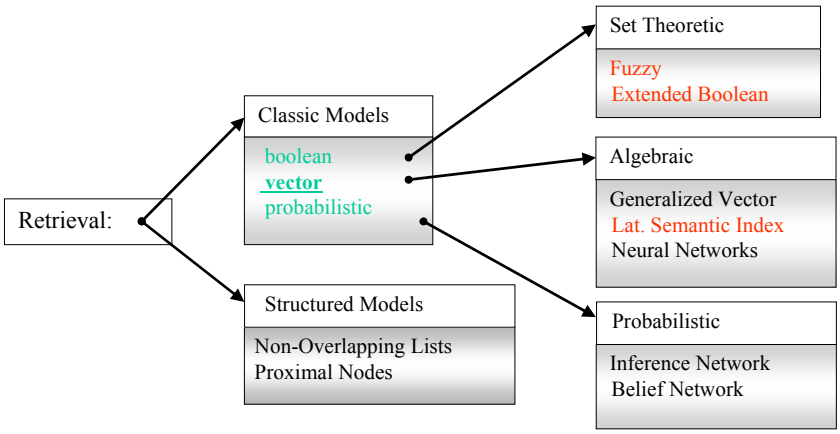
Πλεονεκτήματα:

1. Απλό μοντέλο
2. Τα κείμενα ταξινομούνται σε φθίνουσα διάταξη ως προς την πιθανότητα να είναι σχετικά

Μειονεκτήματα:

1. Χρειάζεται να μαντέψουμε
2. Δε λαμβάνεται υπ' όψιν η συχνότητα εμφάνισης
3. Θεωρεί ότι οι όροι είναι ανεξάρτητοι

Μια Ταξινομία των Μοντέλων Ανάκτησης



Information Retrieval Models

Extended Boolean Model

Extended Boolean Model

- Κίνητρο
  - Το Boolean model είναι απλό και κομψό αλλά δεν παρέχει κατάταξη (διαβάθμιση των συναφών εγγράφων)
- Προσέγγιση
  - Επέκταση του Boolean model με **βάρυνση όρων** και **μερικό ταίριασμα**
  - Συνδυασμός χαρακτηριστικών του Vector model και ιδιοτήτων της Boolean algebra

[Salton, Fox, and Wu, 1983]

Σκεπτικό / Κίνητρο

Έστω $q = k_x \wedge ky$.

Σύμφωνα με το Boolean model ένα έγγραφο που περιέχει **μόνο ένα** από τα k_x, k_y είναι **μη-συναφές**, και μάλιστα τόσο μη-συναφές, όσο ένα έγγραφο που δεν περιέχει **κανένα** από τους 2 όρους.

Έστω ότι έχουμε μόνο δύο όρους k_x , k_y

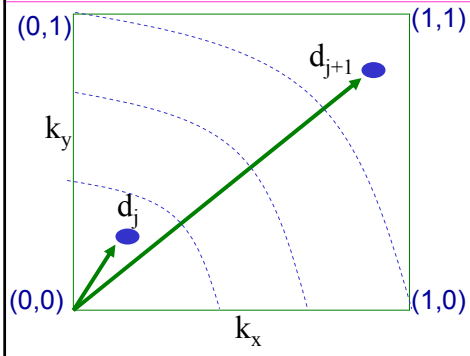
Μπορούμε να θεωρήσουμε κάθε όρο ως μια διάσταση
Άρα έγγραφα και επερωτήσεις απεικονίζονται στο 2D χώρο.

Ένα έγγραφο d_j τοποθετείται βάσει των, βαρών $w_{x,j}$ και $w_{y,j.}$
Έστω ότι τα βάρη αυτά είναι κανονικοποιημένα στο [0,1],
π.χ. :

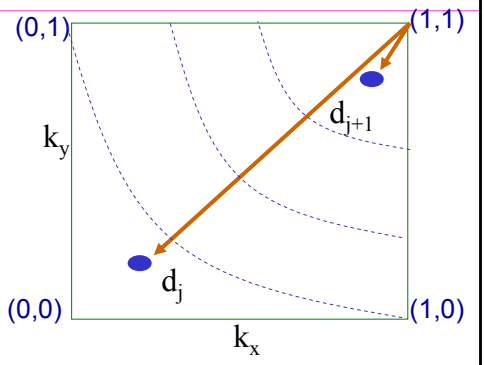
$$w_{x,j} = tf_{x,j}\, idf_x$$
$$w_{y,j} = tf_{y,j}\, idf_y$$

Για συντομία έστω $x = w_{x,j}$ και $y = w_{y,j}$
Άρα οι συντεταγμένες του d_j είναι οι (x,y)

Η γενική ιδέα

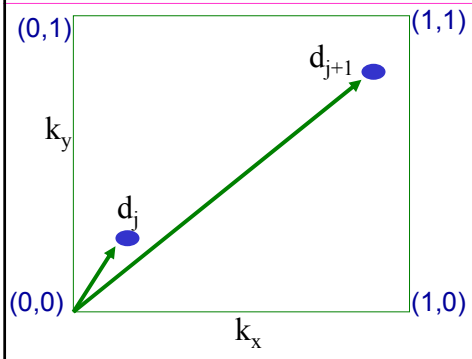


Έστω $q_{OR}=k_x \vee k_y$
Το σημείο (0,0) είναι η θέση προς αποφυγή.
Άρα μπορούμε να θεωρήσουμε την *απόσταση* του d_j από αυτό το σημείο
ως το **βαθμό ομοιότητας** *(όσο πιο μακριά, τόσο πιο όμοιο)*



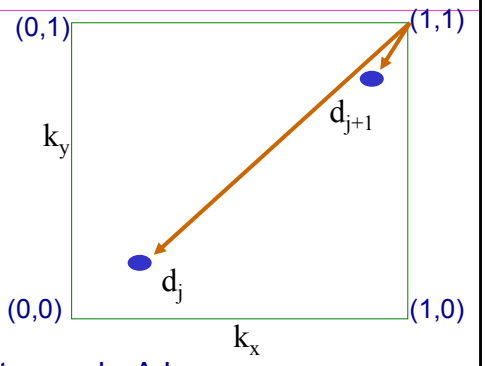
Έστω $q_{AND}=k_x \wedge k_y$
Το σημείο (1,1) είναι η πιο επιθυμητή θέση.
Άρα μπορούμε να θεωρήσουμε το *συμπλήρωμα της απόστασης* του d_j από αυτό το σημείο
ως το **βαθμό ομοιότητας** *(όσο πιο κοντά, τόσο πιο όμοιο)*

Η γενική ιδεά (II)



Let $q_{OR}=k_x \vee k_y$

$$sim(q_{OR}, d) = \sqrt{\frac{x^2 + y^2}{2}}$$



Let $q_{AND}=k_x \wedge k_y$

$$sim(q_{AND}, d) = 1 - \sqrt{\frac{(1-x)^2 + (1-y)^2}{2}}$$

Όταν δυαδικό;

("2" for normalisation to [0,1])

Γενικεύοντας την ιδέα (για >2 όρους)

- Μπορούμε να γενικεύσουμε το προηγούμενο μοντέλο χρησιμοποιώντας την Ευκλείδεια απόσταση στον **t-διάστατο χώρο**
- Αυτό μπορεί να γίνει χρησιμοποιώντας ***p-norms*** που γενικεύουν την έννοια της απόστασης, όπου $1 \leq p \leq \infty$.

- *Διαζευκτικές* επερωτήσεις
  - q_{OR} = k1 V k2 V . . V km

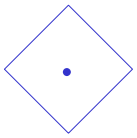
$$sim(q_{OR}, d) = \left(\frac{x_1^p + x_2^p + ... + x_m^p}{m} \right)^{\frac{1}{p}}$$

- *Συζευκτικές* επερωτήσεις
  - q_{AND} = k1 Λ k2 Λ . . . Λ km

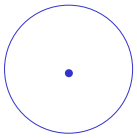
$$sim(q_{AND}, d) = 1 - \left(\frac{(1-x_1)^p + ... + (1-x_m)^p}{m} \right)^{\frac{1}{p}}$$

Ισομετρικές καμπύλες $\sqrt[p]{(x^p + y^p)}$

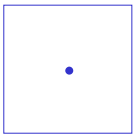
L_1 L_2 L_∞



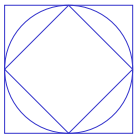
$x + y = 1$



$\sqrt[2]{(x^2 + y^2)} = 1$



$\max(x, y) = 1$



Μερικές ενδιαφέρουσες ιδιότητες

- Μεταβάλλοντας το *p*, μπορούμε να κάνουμε το μοντέλο να συμπεριφέρεται όπως το Vector, το Fuzzy, ή ενδιάμεσα σε αυτά τα δυο.
- Αν p = 1 τότε (Vector like)
  - $sim(q_{OR},d_j) = sim(q_{AND},d_j) = \frac{x_1 + \ldots + x_m}{m}$
- Αν p = ∞ τότε (Fuzzy like)
  - $sim(q_{OR},d_j) = \max(x_i)$
  - $sim(q_{AND},d_j) = \min(x_i)$

Ερώτηση: Που πήγαν οι όροι της επερώτησης;

Σύνθετες επερωτήσεις

- Έστω q = $(k_1 \Lambda k_2) V k_3$
- Εφαρμόζουμε τους ορισμούς σεβόμενοι τη σειρά, εδώ:

$$sim(q\ ,d) = \left(\frac{(1-(\frac{(1-x_1)^p + (1-x_2)^p}{2})^{1/p})^p + x_3{}^p}{2} \right)^{\frac{1}{p}}$$

- Έστω q = $(k_1 V^2 k_2) \Lambda^{\infty} k_3$
  - K_1 and k_2 should be used as in a vector system but the presence of k_3 is required

Μερικές Παρατηρήσεις

- Είναι αρκετά ισχυρό μοντέλο με ενδιαφέρουσες ιδιότητες
- Η επιμεριστική ιδιότητα δεν ισχύει:
  - $q_1 = (k_1 \vee k_2) \wedge k_3$
  - $q_2 = (k_1 \wedge k_3) \vee (k_2 \wedge k_3)$
  - $sim(q_1,d_j) \neq sim(q_2,d_j)$

Information Retrieval Models

Fuzzy Set-based Retrieval Model

Μοντέλα Βασισμένα στη Θεωρία Ασαφών Συνόλων

(Fuzzy Set-based Retrieval Models)

Κίνητρο

– Επέκταση του Boolean model με **μερικό** ταίριασμα (και άρα με δυνατότητες διαβάθμισης των στοιχείων των απαντήσεων)

Τι είναι ένα ασαφές σύνολο;

«Κλασσικά» σύνολα (crispy or Boolean sets): ένα στοιχείο ανήκει ή δεν ανήκει

Ασαφή σύνολα: ένα στοιχείο του συνόλου ανήκει με ένα βαθμό συμμετοχής (<=1)

Ιδέα:

Κάθε όρος της ερώτησης ένα ασαφές σύνολο

Ένα έγγραφο ανήκει σε αυτό το ασαφές σύνολο του όρου με ένα βαθμό

Μοντέλα Βασισμένα στη Θεωρία Ασαφών Συνόλων
(Fuzzy Set-based Retrieval Models)

Έχουν προταθεί αρκετά μοντέλα που βασίζονται σε fuzzy sets. Εδώ θα δούμε δύο:

- Ένα απλό μοντέλο που βασίζεται σε TF-IDF και fuzzy theory
- Το μοντέλο που προτάθηκε στο [Ogawa, Morita, and Kobayashi, 1991]

Background: Fuzzy Set Theory [Zadeh 1965]

- Framework for representing classes whose boundaries are not well defined
- Key idea is to introduce the notion of a ***degree of membership*** (βαθμός συμμετοχής) associated with the elements of a set
- This degree of membership varies from 0 to 1(τιμές στο διάστημα [0, 1]) and allows modeling the notion of *marginal* membership
- Thus, membership is now a *gradual* notion, contrary to the crispy notion enforced by classic Boolean logic

Background: Fuzzy Set Theory [Zadeh 1965]

- U: universe of discourse
- A fuzzy subset A of U is characterized by a membership function

$$\mu_A(u) : U \rightarrow [0,1]$$

which associates with each element *u* of U a number $\mu_A(u)$ in [0,1]

Βασικές πράξεις σε ασαφή σύνολα (συμπλήρωμα, τομή και ένωση)

- Let A and B be two fuzzy subsets of U, and ¬A be the complement of A. Then,
  - $\mu_{\neg A}(u) = 1 - \mu_A(u)$
  - $\mu_{A \cup B}(u) = \max(\mu_A(u), \mu_B(u))$
  - $\mu_{A \cap B}(u) = \min(\mu_A(u), \mu_B(u))$

A Simple Retrieval Model based on Fuzzy Theory
Παράσταση εγγράφων

$$\begin{array}{c|cccc} & k_1 & k_2 & \ldots & k_t \\ d_1 & w_{11} & w_{21} & \ldots & w_{t1} \\ d_2 & w_{12} & w_{22} & \ldots & w_{t2} \\ \vdots & \vdots & \vdots & & \vdots \\ \vdots & \vdots & \vdots & & \vdots \\ d_n & w_{1n} & w_{2n} & \ldots & w_{tn} \end{array}$$

$w_{i,j} \in [0,1]$

- K={k_1,…,k_t} : σύνολο όλων των λέξεων ευρετηρίασης
- Κάθε έγγραφο $\mathbf{d_j}$ παριστάνεται με το διάνυσμα $d_j=(w_{1,j},\ldots,w_{t,j})$ όπου
  - $w_{i,j}$ το βάρος της λέξης k_i για το κείμενο $\mathbf{d_j}$
  - για παράδειγμα $w_{i,j}$ = $\mathbf{tf_{ij}\ idf_i}$

A Simple Retrieval Model based on Fuzzy Theory

Boolean Queries and Ranking Function

- Μια επερώτηση q είναι μια λογική έκφραση στο K, πχ:
  - q = "**k1 and (k2 or not k3))**" δηλαδή q = "**k1** $\wedge$ **(k2** $\vee \neg$ **k3))**"
- $R(d_j,q) = \mu_q(d_j)$, άρα είναι ο βαθμός συμμετοχής του d_j στο σύνολο που προσδιορίζεται από τη λογική έκφραση q.
- Μπορούμε να υπολογίσουμε το $R(d_j,q)$ βάσει των κανόνων της θεωρίας των Fuzzy sets, θεωρώντας ότι $R(d_j,t_i) = \mu_{ti}(d_j) = w_{i,j}$
- Για παράδειγμα
  - $R(d_j,\ t1 \vee t2) = \max (R(d_j, t1), R(d_j, t2)) = \max (w_{1j}, w_{2j})$.
  - $R(d_j,\ t1 \wedge t2) = \min (R(d_j, t1), R(d_j, t2)) = \min (w_{1j}, w_{2j})$.

A Simple Retrieval Model based on Fuzzy Theory

Παρατηρήσεις

- Έστω $q = k_x \wedge k_y$. Σύμφωνα με το Boolean model ένα έγγραφο που περιέχει **μόνο έναν** από τους όρους k_x, k_y είναι **μη-συναφές**, και μάλιστα τόσο μη-συναφές, όσο ένα έγγραφο που δεν περιέχει **κανένα** από τους 2 όρους.
  - Ερώτηση: Τι συμβαίνει εδώ;
  - Απάντηση: Το ίδιο
- Έστω $q = k_x \vee k_y$. Σύμφωνα με το Boolean model ένα έγγραφο που περιέχει **και τους δύο όρους** (k_x, k_y) είναι **το ίδιο συναφές**, με ένα έγγραφο που περιέχει **έναν** από τους 2 όρους.
  - Ερώτηση: Τι συμβαίνει εδώ;
  - Απάντηση: ...
  - Άρα το παρόν μοντέλο διαβαθμίζει τα στοιχεία της απάντησης του $q = k_x \vee ky$ (κάτι που δεν είναι δυνατό με το Boolean Μοντέλο).
- Το παρόν είναι μια ειδική περίπτωση του Extended Boolean Model (συγκεκριμένα αντιστοιχεί στην περίπτωση που $p = \infty$).

A Simple Retrieval Model based on Fuzzy Theory

Παρατηρήσεις

Πως θα υπολογίζουμε τη συνάρτηση συμμετοχής

[Ogawa, Morita, and Kobayashi,1991]

Fuzzy Set Retrieval Model [Ogawa, Morita, and Kobayashi,1991]

Εδώ θα δούμε το μοντέλο που προτάθηκε στο [Ogawa,Morita, Kobayashi,1991)

- Βασική Ιδέα:
  - Έγγραφα και επερωτήσεις παριστάνονται με **σύνολα** όρων ευρετηρίου (εδώ δεν έχουμε βάρη στο [0,1])
  - Κάθε **όρος** συσχετίζεται με ένα **fuzzy set**
  - Κάθε έγγραφο έχει ένα degree of membership σε αυτό το fuzzy set
- Παράδειγμα:
  - Έστω επερώτηση ***q* = *αυτοκίνητο***
  - Έστω έγγραφο d1 που *δεν περιέχει* τη λέξη αυτοκίνητο αλλά περιέχει τη λέξη «όχημα».
  - Αν υπάρχουν **πολλά** έγγραφα που περιέχουν και τις δυο λέξεις, τότε, υπάρχει ισχυρή συσχέτιση των δυο αυτών λέξεων, και
  - => άρα το d1 μπορεί να θεωρηθεί **συναφές** με την επερώτηση q.

Fuzzy Set Retrieval Model
Πίνακας Συσχέτισης (correlation matrix) και εγγύτητα όρων

Πίνακα συσχέτισης μεταξύ των όρων

term-term correlation matrix ή keyword connection matrix

	k_1	k_2	....	k_t
k_1	c_{11}	c_{21}	...	c_{t1}
k_2	c_{12}	c_{22}	...	c_{t2}
:	:	:		:
:	:	:		:
k_t	c_{1n}	c_{2n}	...	c_{tn}

Ορίζουμε ποσοτικά την εγγύτητα (proximity) μεταξύ δυο όρων k_i και k_l->
ως την συν-εμφάνισή τους στα έγγραφα της συλλογής

$$c_{i,l} = \frac{n_{i,l}}{n_i + n_l - n_{i,l}}$$

where:
$n_{i,l}$: number of docs which contain both k_i and k_l
n_i: number of docs which contain k_i
n_l: number of docs which contain k_l

Fuzzy Set Retrieval Model
Πίνακας Συσχέτισης (correlation matrix) και εγγύτητα όρων

$$\begin{array}{c|cccc} & k_1 & k_2 & \dots & k_t \\ k_1 & c_{11} & c_{21} & \dots & c_{t1} \\ k_2 & c_{12} & c_{22} & \dots & c_{t2} \\ \vdots & \vdots & \vdots & & \vdots \\ \vdots & \vdots & \vdots & & \vdots \\ k_t & c_{1n} & c_{2n} & \dots & c_{tn} \end{array}$$

$$\mathbf{c(i,l) = c_{i,l} = \frac{n_{i,l}}{n_i + n_l - n(i,l)}}$$

where:
$n_{i,l}$: number of docs which contain both k_i and k_l
n_i: number of docs which contain k_i
n_l: number of docs which contain k_l

Τέτοιες πίνακες είναι αρκετά συνηθισμένοι (θα τους ξαναδούμε σε αλγόριθμους clustering)

Πχ
$n_{il}=0$ => $c_{il}=0$
$n_{il}=3, n_i=3, n_l=9$ => $c_{il}=0.3$
$n_{il}=3, n_i=3, n_l=30$ => $c_{il}=0.1$
$n_{il}=3, n_i=3, n_l=3$ => $c_{il}=1$

Fuzzy Set Retrieval Model
Μορφή Ευρετηρίου: όπως και στο Boolean model.

$$\begin{array}{c|cccc} & k_1 & k_2 & \dots & k_t \\ d_1 & w_{11} & w_{21} & \dots & w_{t1} \\ d_2 & w_{12} & w_{22} & \dots & w_{t2} \\ \vdots & \vdots & \vdots & & \vdots \\ \vdots & \vdots & \vdots & & \vdots \\ d_n & w_{1n} & w_{2n} & \dots & w_{tn} \end{array}$$

$w_{i,j} \in \{0,1\}$

- K={k_1,…,k_t}: σύνολο όλων των λέξεων ευρετηρίασης
- Κάθε έγγραφο **d_j** παριστάνεται με το διάνυσμα $d_j=(w_{1,j},\dots,w_{t,j})$ όπου:
  - $w_{i,j}$ = **1** αν η λέξη k_i εμφανίζεται στο κείμενο **d_j** (αλλιώς $w_{i,j}$ =**0**)

Βάσει αυτού του πίνακα θα δημιουργήσουμε έναν πίνακα συσχέτισης όρων
(για να καταχωρήσουμε σχέσεις όπως «αυτοκίνητο» ≈ «όχημα»)

Fuzzy Set Retrieval Model [Ogawa, Morita, and Kobayashi,1991]

Έστω όρος k_i και έγγραφο d_j θέλουμε το βαθμό συμμετοχής του εγγράφου στο ασαφές σύνολο που ορίζει το k_i (συνάρτηση συμμετοχής μ_i)

$$\mu_i(j) = \sum_{k_w \in dj} c_{i,w}$$

Άθροισμα του βαθμού συσχέτισης του k_i με τους όρους που εμφανίζονται στο d_j (θεωρούμε άθροισμα αντί για max, πιο ήπια διαβάθμιση)

$$= 1 - \prod_{k_w \in dj} (1 - c_{i,w})$$

Βασίζεται στο:

$$(\cup A_i)^c = \cap A_i^c$$

$$\cup A_i = \Omega - (\cup A_i)^c = \Omega - \cap A_i^c$$

Για παράδειγμα έστω ότι το έγγραφο d_j δεν περιέχει τον όρο k_i

- Αν το έγγραφο d_j περιέχει έναν όρο k_w που σχετίζεται ισχυρά με τον k_i τότε
  - θα έχουμε $c_{iw} \sim 1$
  - και άρα θα μπορούσαμε να θεωρήσουμε ότι $\mu_i(j) \sim 1$. Με άλλα λόγια, αν και ο όρος k_i δεν εμφανίζεται στο d_j, εντούτοις περιγράφει το περιεχόμενο του d_j

Fuzzy Set Retrieval Model
Fuzzy Information Retrieval

Έστω q σε DNF $q = cc1 \vee \ldots \vee cck$, όπου cci είναι μια συζευκτική συνιστώσα

Σύμφωνα με τη fuzzy set theory:

$$\mu_q(j) = \max(\mu_{cc1}(j), \ldots, \mu_{cck}(j))$$

Παρά ταύτα, εδώ προτείνεται η χρήση αθροίσματος αντί του μεγίστου.

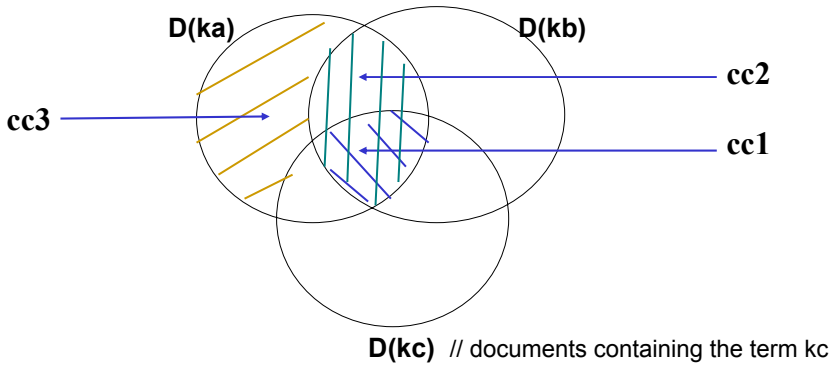
$R(d_j,q) = \mu_q(d_j) = \Sigma\, \mu_{cc}(d_j)$ για κάθε συζευκτική συνιστώσα cc του q_{DNF}

Fuzzy Set Retrieval Model
Παράδειγμα

$q = ka \wedge (kb \vee \neg kc)$

$vec(q_{dnf}) = (1,1,1) + (1,1,0) + (1,0,0)$

$= vec(cc1) + vec(cc2) + vec(cc3)$



Fuzzy Set Retrieval Model
Παράδειγμα (II)

$q = ka \wedge (kb \vee \neg kc)$

$vec(q_{dnf}) = (1,1,1) + (1,1,0) + (1,0,0)$

$= vec(cc1) + vec(cc2) + vec(cc3)$

$$\mu_q(d_j) = \mu_{cc1+cc2+cc3}(d_j) = 1 - \prod_{i=1..3} (1 - \mu_{cci}(d_j))$$

$= 1 - (1- [1,1,1]) * (1 - [1,1,0]) * (1 - [1,0,0])$

$\mu_a(d_j)\ \mu_b(d_j)\ \mu_c(d_j))$ $\mu_a(d_j)\ \mu_b(d_j)\ (1-\mu_c(d_j)))$ $\mu_a(d_j)\ (1-\mu_b(d_j))\ (1-\mu_c(d_j)))$

Fuzzy Set Retrieval Model
Σύνοψη

- K={k_1,…,k_t} : σύνολο όλων των λέξεων ευρετηρίασης
- Κάθε έγγραφο **d_j** παριστάνεται με το διάνυσμα d_j=($w_{1,j}$,…,$w_{t,j}$) όπου:
  - $w_{i,j}$ **= 1** αν η λέξη k_i εμφανίζεται στο κείμενο **d_j** (αλλιώς $w_{i,j}$ =**0**)
- Μια επερώτηση q είναι μια λογική έκφραση στο K, πχ:
  - q = "**k1 and (k2 or not k3))**" δηλαδή q = "**k1 ∧ (k2 ∨ ¬ k3))**"
  - q_{DNF} = "**(k1 ∧ k2 ∧ k3) ∨(k1 ∧ k2 ∧¬ k3) ∨(k1 ∧ ¬ k2 ∧ ¬ k3)**"
  - q_{DNF} = "**(1,1,1) ∨(1,1,0) ∨(1,0,0)**"
- R(dj,q) = $\mu_q(dj) = \Sigma\ \mu_{cc}(dj)$ για κάθε συζευκτική συνιστώσα cc του q_{DNF}
  - $\mu_{ki}(dj) = 1 - \prod_{k_w \in dj} (1 - c(k_i,k_w))$
  - $c(k_i,k_j)$ καθορίζεται από την συνεμφάνιση των όρων k_i και k_j στη συλλογή

Fuzzy Set Retrieval Model
Γενικά σχόλια

- Έχουν συζητηθεί κυρίως στο χώρο της fuzzy theory
- Δεν έχουμε επαρκή αποτελέσματα πειραματικής αξιολόγησης για να τα αντιπαραβάλλουμε με τα προηγούμενα μοντέλα

Information Retrieval Models

Latent Semantic Indexing (LSI)

Λανθάνουσα Σημασιολογική Ευρετηρίαση

Σκεπτικό / Κίνητρο

- Classic IR might lead to poor retrieval due to:
  - relevant documents that do not contain at least one index term are not retrieved
  - A document that shares concepts with another document known to be relevant might be of interest
- The user information need is more related to **concepts** and **ideas** than to index terms
- We want to capture the concepts instead of the words.
- Concepts are reflected in the words. However:
  - One term may have ***multiple*** meanings (**polysemy**)
  - *Different* terms may have the *same* meaning (**synonymy**)

LSI: The approach

- LSI approach tries to overcome the deficiencies of term-matching retrieval by treating the unreliability of observed term-document association data as a **statistical problem**.
- The goal is to find effective models to represent the relationship between terms and documents.
- Hence a set of terms, which is by itself incomplete and unreliable, will be replaced by some set of entities which are more reliable indicants.

Γιατί λέγεται “Latent …”

- Διότι γίνεται η υπόθεση ότι υπάρχει μια «λανθάνουσα» δομή στον τρόπο χρήσης των λέξεων στα έγγραφα
- Το LSI αξιοποιεί στατιστικές τεχνικές για την εκτίμησή της

LSI: The idea

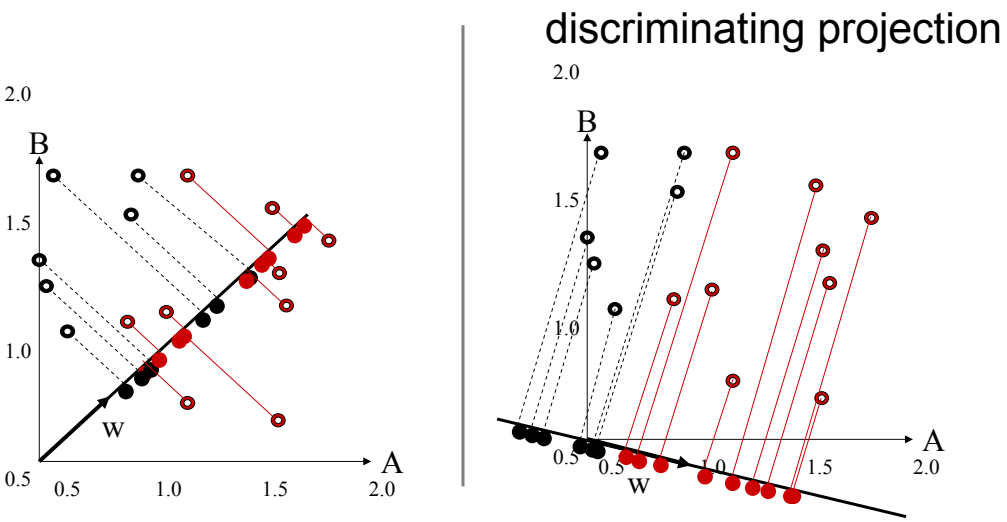
- The key idea is to map documents and queries into a **lower dimensional space**
  - (i.e., composed of higher level concepts which are fewer in number than the index terms)

- Retrieval in the reduced concept space might be superior to retrieval in the space of index terms

- But how to learn the concepts from data?

Μείωση Διαστάσεων και Διακριτική Ικανότητα
(μπορεί να έχουμε μείωση της διακριτικής ικανότητας, μπορεί όμως και όχι)

Παράδειγμα προβολής 2 διαστάσεων σε μία



SVD (Singular Value Decomposition)

- LSI is based on SVD (Singular Value Decomposition)
- So SVD is applied to derive the latent semantic structure model.

- What is SVD?
  - A dimensionality reduction technique

  - For more about matrices and SVD see:
  - The Matrix Cookbook
    http://www.imm.dtu.dk/pubdb/views/edoc_download.php/3274/pdf/imm3274.pdf
  - http://kwon3d.com/theory/jkinem/svd.html
  - http://mathworld.wolfram.com/SingularValueDecomposition.html
  - http://www.miislita.com/information-retrieval-tutorial/svd-lsi-tutorial-1-understanding.html

SVD (HIDE)

- SVD: Διάσπαση σε ιδιάζουσες τιμές
- Ένας μεγάλος πίνακας όρων-εγγράφων αναλύεται σε ένα σύνολο από κ (100..200) ορθοκανονικούς παράγοντες από τους οποίους ο αρχικός πίνακας μπορεί να προσεγγιστεί με γραμμικό συνδυασμό.
- Πλέον έγγραφα και επερωτήσεις παριστάνονται βάσει αυτών των κ διαστάσεων

- Αφού οι διαστάσεις μειώθηκαν, οι λέξεις δεν μπορεί πλέον να είναι ανεξάρτητες

Definitions

- t: total number of index terms
- d: total number of documents

Αρχικός Πίνακας (t x d)

x

$$\begin{array}{c|cccc} & d_1 & d_2 & \ldots & d_d \\ k_1 & w_{11} & w_{21} & \ldots & w_{d1} \\ k_2 & w_{12} & w_{22} & \ldots & w_{d2} \\ \vdots & \vdots & \vdots & & \vdots \\ \vdots & \vdots & \vdots & & \vdots \\ k_t & w_{1t} & w_{2t} & \ldots & w_{dt} \end{array}$$

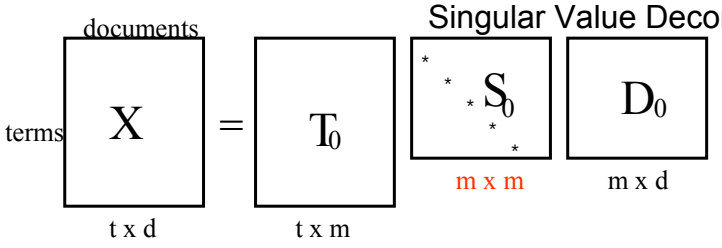
- (X_{ij}): be a term-document matrix with t rows and d columns
  - To each element of this matrix a weight $\mathbf{w_{ij}}$ associated is assigned with the pair $\mathbf{[k_i,d_j]}$
  - The weight w_{ij} can be $freq_{ij}$
    - (or based on a **tf-idf** weighting scheme)

$w_{i,j} \in [\mathbf{0},\mathbf{1}]$

Latent Semantic Indexing: Ο τρόπος

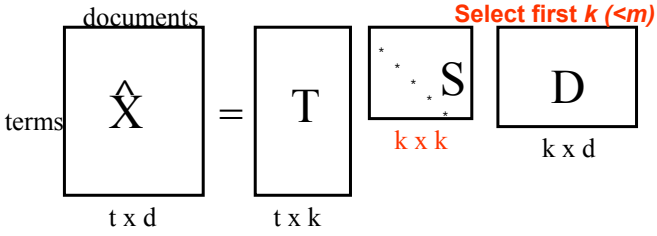
t: total number of index terms
d: total number of documents

Singular Value Decomposition

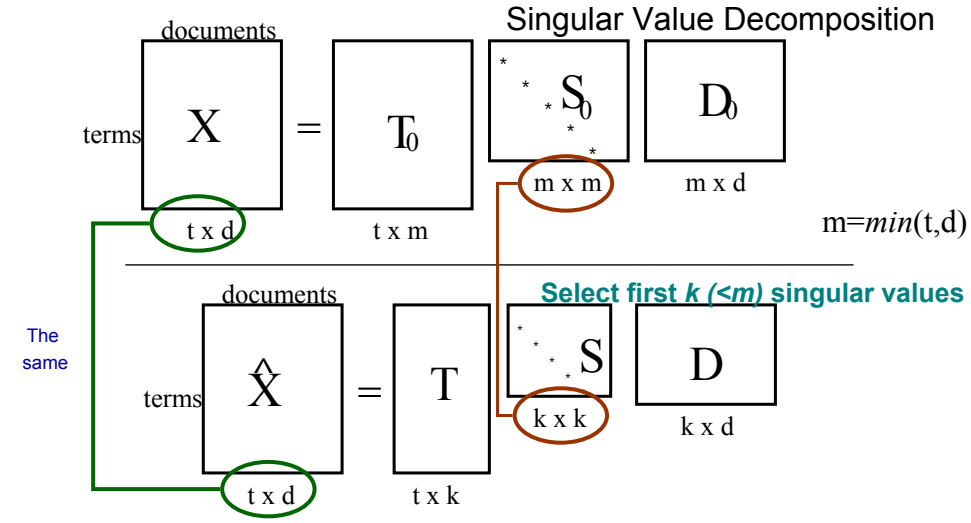


m=*min*(t,d)

Select first *k (<m)* singular values



t: total number of index terms
d: total number of documents



SVD

- SVD of the term-by-document matrix X:

$$X = T_0 S_0 D_0'$$

- If the singular values of S_0 are ordered by size, we only keep the first k largest values and get a reduced model:

$$\hat{X} = TSD'$$

  - $\hat{X}$ doesn't exactly match X and it gets closer as more and more singular values are kept
  - This is what we want. We don't want perfect fit since we think some of 0's in X should be 1 and vice versa.
  - It reflects the major associative patterns in the data, and ignores the smaller, less important influence and noise.

LSI Paper example

Index terms in italics

Titles:

c1: *Human* machine *interface* for Lab ABC *computer* applications
c2: A *survey* of *user* opinion of *computer system response time*
c3: The *EPS user interface* management *system*
c4: *System* and *human system* engineering testing of *EPS*
c5: Relation of *user*-perceived *response time* to error measurement

m1: The generation of random, binary, unordered *trees*
m2: The intersection *graph* of paths in *trees*
m3: *Graph minors* IV: Widths of *trees* and well-quasi-ordering
m4: *Graph minors*: A *survey*

term-document Matrix

Terms	Documents								
	c1	c2	c3	c4	c5	m1	m2	m3	m4
human	1	0	0	1	0	0	0	0	0
interface	1	0	1	0	0	0	0	0	0
computer	1	1	0	0	0	0	0	0	0
user	0	1	1	0	1	0	0	0	0
system	0	1	1	2	0	0	0	0	0
response	0	1	0	0	1	0	0	0	0
time	0	1	0	0	1	0	0	0	0
EPS	0	0	1	1	0	0	0	0	0
survey	0	1	0	0	0	0	0	0	1
trees	0	0	0	0	0	1	1	1	0
graph	0	0	0	0	0	0	1	1	1
minors	0	0	0	0	0	0	0	1	1

Weight = number of occurrences

T_0

$$\begin{bmatrix}
0.22 & -0.11 & 0.29 & -0.41 & -0.11 & -0.34 & 0.52 & -0.06 & -0.41 \\
0.20 & -0.07 & 0.14 & -0.55 & 0.28 & 0.50 & -0.07 & -0.01 & -0.11 \\
0.24 & 0.04 & -0.16 & -0.59 & -0.11 & -0.25 & -0.30 & 0.06 & 0.49 \\
0.40 & 0.06 & -0.34 & 0.10 & 0.33 & 0.38 & 0.00 & 0.00 & 0.01 \\
0.64 & -0.17 & 0.36 & 0.33 & -0.16 & -0.21 & -0.17 & 0.03 & 0.27 \\
0.27 & 0.11 & -0.43 & 0.07 & 0.08 & -0.17 & 0.28 & -0.02 & -0.05 \\
0.27 & 0.11 & -0.43 & 0.07 & 0.08 & -0.17 & 0.28 & -0.02 & -0.05 \\
0.30 & -0.14 & 0.33 & 0.19 & 0.11 & 0.27 & 0.03 & -0.02 & -0.17 \\
0.21 & 0.27 & -0.18 & -0.03 & -0.54 & 0.08 & -0.47 & -0.04 & -0.58 \\
0.01 & 0.49 & 0.23 & 0.03 & 0.59 & -0.39 & -0.29 & 0.25 & -0.23 \\
0.04 & 0.62 & 0.22 & 0.00 & -0.07 & 0.11 & 0.16 & -0.68 & 0.23 \\
0.03 & 0.45 & 0.14 & -0.01 & -0.30 & 0.28 & 0.34 & 0.68 & 0.18
\end{bmatrix}$$

S_0

$$\begin{bmatrix}
3.34 & & & & & & & & \\
& 2.54 & & & & & & & \\
& & 2.35 & & & & & & \\
& & & 1.64 & & & & & \\
& & & & 1.50 & & & & \\
& & & & & 1.31 & & & \\
& & & & & & 0.85 & & \\
& & & & & & & 0.56 & \\
& & & & & & & & 0.36
\end{bmatrix}$$

D_0

$$\begin{bmatrix}
0.20 & -0.06 & 0.11 & -0.95 & 0.05 & -0.08 & 0.18 & -0.01 & -0.06 \\
0.61 & 0.17 & -0.50 & -0.03 & -0.21 & -0.26 & -0.43 & 0.05 & 0.24 \\
0.46 & -0.13 & 0.21 & 0.04 & 0.38 & 0.72 & -0.24 & 0.01 & 0.02 \\
0.54 & -0.23 & 0.57 & 0.27 & -0.21 & -0.37 & 0.26 & -0.02 & -0.08 \\
0.28 & 0.11 & -0.51 & 0.15 & 0.33 & 0.03 & 0.67 & -0.06 & -0.26 \\
0.00 & 0.19 & 0.10 & 0.02 & 0.39 & -0.30 & -0.34 & 0.45 & -0.62 \\
0.01 & 0.44 & 0.19 & 0.02 & 0.35 & -0.21 & -0.15 & -0.76 & 0.02 \\
0.02 & 0.62 & 0.25 & 0.01 & 0.15 & 0.00 & 0.25 & 0.45 & 0.52 \\
0.08 & 0.53 & 0.08 & -0.03 & -0.60 & 0.36 & 0.04 & -0.07 & -0.45
\end{bmatrix}$$

SVD with minor terms dropped

T S D'

$$\begin{bmatrix}
0.22 & -0.11 \\
0.20 & -0.07 \\
0.24 & 0.04 \\
0.40 & 0.06 \\
0.64 & -0.17 \\
0.27 & 0.11 \\
0.27 & 0.11 \\
0.30 & -0.14 \\
0.21 & 0.27 \\
0.01 & 0.49 \\
0.04 & 0.62 \\
0.03 & 0.45
\end{bmatrix}
\begin{bmatrix}
3.34 & \\
 & 2.54
\end{bmatrix}
\begin{bmatrix}
0.20 & 0.61 & 0.46 & 0.54 & 0.28 & 0.00 & 0.02 & 0.02 & 0.08 \\
-0.06 & 0.17 & -0.13 & -0.23 & 0.11 & 0.19 & 0.44 & 0.62 & 0.53
\end{bmatrix}$$

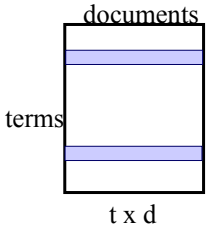
TS define coordinates for documents in latent space

Παρατηρήσεις

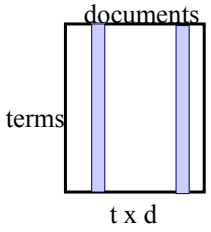
- Η παράμετρος k (<m) πρέπει να είναι:
  - large enough to allow fitting the characteristics of the data
  - small enough to filter out the non-relevant representational details

Τρόπος Σύγκρισης Όρων και Εγγράφων

- Τρόπος σύγκρισης 2 όρων:
  - the **dot product** (or cosine) between two **row vectors** reflects the extent to which two terms have a similar pattern of occurrence across the set of document.

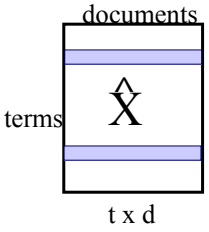


- Τρόπος σύγκρισης δύο εγγράφων:
  - **dot product** (or cosine) between two **column vectors**

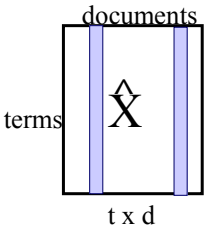


Τρόπος Σύγκρισης Όρων και Εγγράφων $\hat{X}$

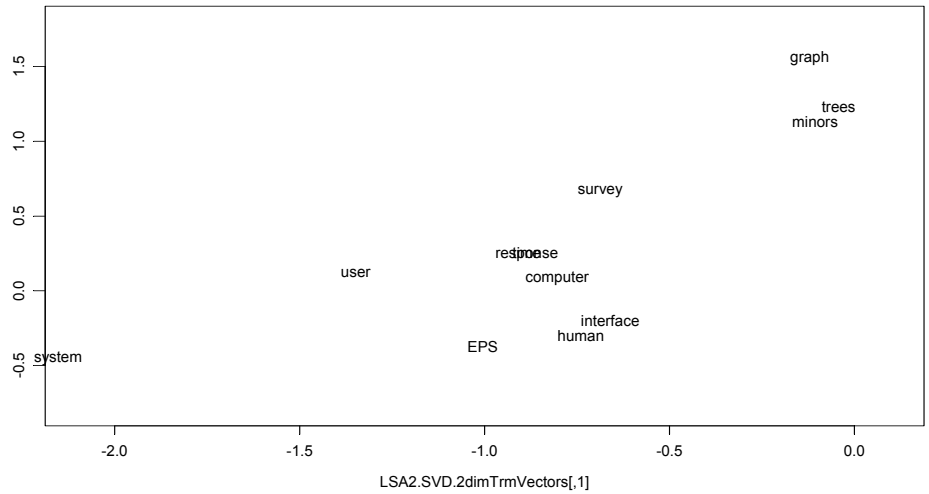
- Τρόπος σύγκρισης 2 όρων:
  - the **dot product** (or cosine) between two **row vectors** reflects the extent to which two terms have a similar pattern of occurrence across the set of document.



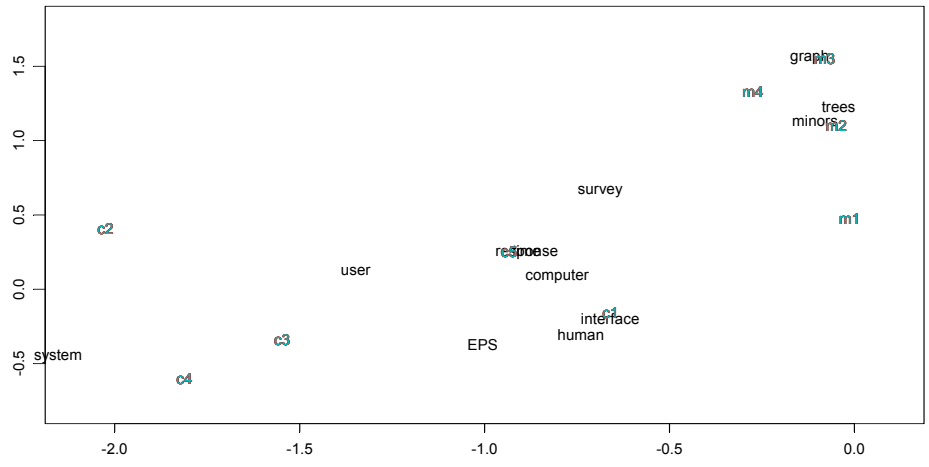
- Τρόπος σύγκρισης δύο εγγράφων:
  - **dot product** (or cosine) between two **column vectors**



Terms Graphed in Two Dimensions



Documents and Terms



Change in Text Correlation

Correlations between text in raw data

	c1	c2	c2	c4	c5	m1	m2	m3	m4
c1	1.000								
c2	-0.192	1.000							
c3	0.000	0.000	1.000						
c4	0.000	0.000	0.472	1.000					
c5	-0.333	0.577	0.000	-0.309	1.000				
m1	-0.174	-0.302	-0.213	-0.161	-0.174	1.000			
m2	-0.258	-0.447	-0.316	-0.239	-0.258	0.674	1.000		
m3	-0.333	-0.577	-0.408	-0.309	-0.333	0.522	0.775	1.000	
m4	-0.333	-0.192	-0.408	-0.309	-0.333	-0.174	0.258	0.556	1.000

Correlations in two-dimensional space

	c1	c2	c2	c4	c5	m1	m2	m3	m4
c1	1.000								
c2	0.910	1.000							
c3	1.000	0.912	1.000						
c4	0.998	0.884	0.998	1.000					
c5	0.842	0.990	0.844	0.809	1.000				
m1	-0.858	-0.568	-0.856	-0.887	-0.445	1.000			
m2	-0.853	-0.562	-0.851	-0.883	-0.438	1.000	1.000		
m3	-0.852	-0.559	-0.850	-0.881	-0.435	1.000	1.000	1.000	
m4	-0.811	-0.497	-0.809	-0.845	-0.368	0.996	0.997	0.997	1.000

Latent Semantic Indexing: Ranking

- Η επερώτηση ***q*** του χρήστη μοντελοποιείται ως ένα **ψευδο-έγγραφο** στον αρχικό πίνακα X

$$\mathbf{X} = \begin{pmatrix} & d_1 & d_2 & .... & d_d & \mathbf{q} \\ k_1 & w_{11} & w_{21} & \ldots & w_{d1} & w_{q1} \\ k_2 & w_{12} & w_{22} & \ldots & w_{d2} & w_{q2} \\ \vdots & \vdots & \vdots & & \vdots & \\ \vdots & \vdots & \vdots & & \vdots & \\ k_t & w_{1t} & w_{2t} & \ldots & w_{dt} & w_{qt} \end{pmatrix}$$

LSI: Συμπεράσματα

- Latent semantic indexing provides an interesting conceptualization of the IR problem
- It allows reducing the complexity of the underline representational framework which might be explored, for instance, with the purpose of interfacing with the user
- Problems
  - If new documents are added then we have to recompute X^

Το υπολογιστικό κόστος για το SVD πολύ μεγάλο

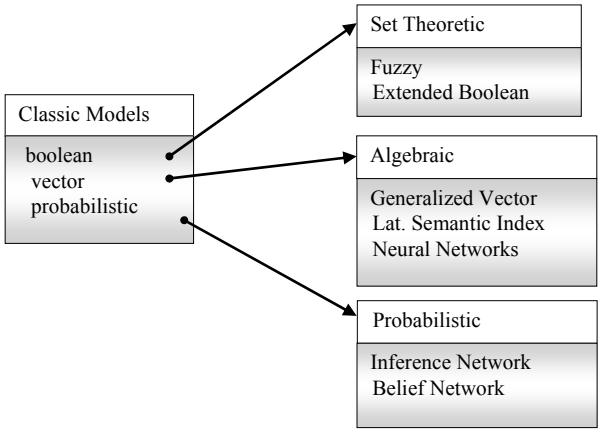
Δουλεύει καλύτερα σε εφαρμογές που υπάρχει μικρή επικάλυψη μεταξύ των ερωτημάτων και των εγγράφων

Μικρές τιμές του k (εκατοντάδες)

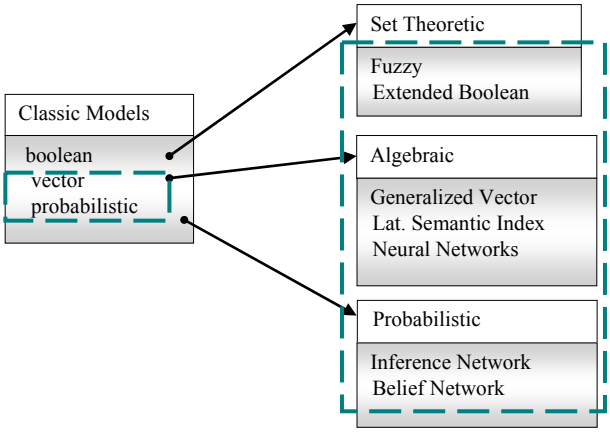
Δεν υπάρχει τρόπος να εκφραστεί απουσία όρου και exact match

Επισκόπηση των Μοντέλων Ανάκτησης
που έχουμε εξετάσει μέχρι τώρα

Ταξινομία Μοντέλων που εξετάσαμε

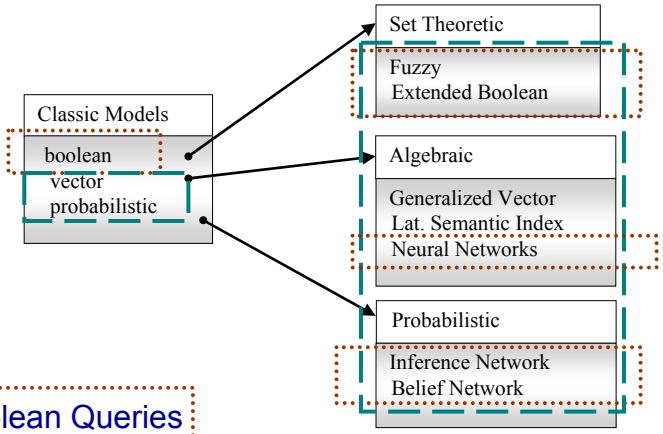


Ταξινομία Μοντέλων που εξετάσαμε



Partial Matching

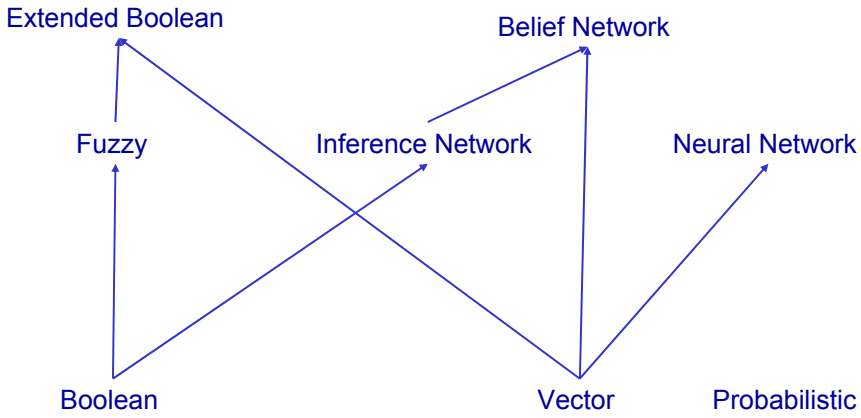
Ταξινομία Μοντέλων που εξετάσαμε



Boolean Queries

Partial Matching

Βάσει της εκφραστικής τους ικανότητας (incomplete)



Άλλοι τύποι Μοντέλων Ανάκτησης που ενδεχομένως να προλάβουμε να δούμε αργότερα

- Μοντέλα Ανάκτησης Πληροφοριών από **Ιστοσελίδες**
  - Έμφαση στους συνδέσμους
- Μοντέλα Ανάκτησης **Πολυμέσων**
- Μοντέλα Ανάκτησης **Δομημένων** Εγγράφων (π.χ. XML)
- Μοντέλα Βασισμένα στη **Λογική**

Θα δούμε τα «κόκκινα» αργότερα στο μάθημα