HY463 - Συστήματα Ανάκτησης Πληροφοριών
Information Retrieval (IR) Systems

Μοντέλα Ανάκτησης Ι
(Retrieval Models)

Γιάννης Τζίτζικας

Διάλεξη : 3
Ημερομηνία :26&28/-2-2008

Διάρθρωση

- **Εισαγωγή στα Μοντέλα Άντλησης**
- **Κατηγορίες Μοντέλων**
- **Απόλυτο και Κάλλιστο (ή Βέλτιστο) Ταίριασμα (Exact vs Best Match)**
- **Τα κλασσικά μοντέλα ανάκτησης**
- **Το Boolean Μοντέλο**
- **Στατιστικά Μοντέλα - Βάρυνση Όρων**
- **Το Διανυσματικό Μοντέλο**
- **Το Εκτεταμένο Boolean μοντέλο (Extended Boolean Model)**

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.

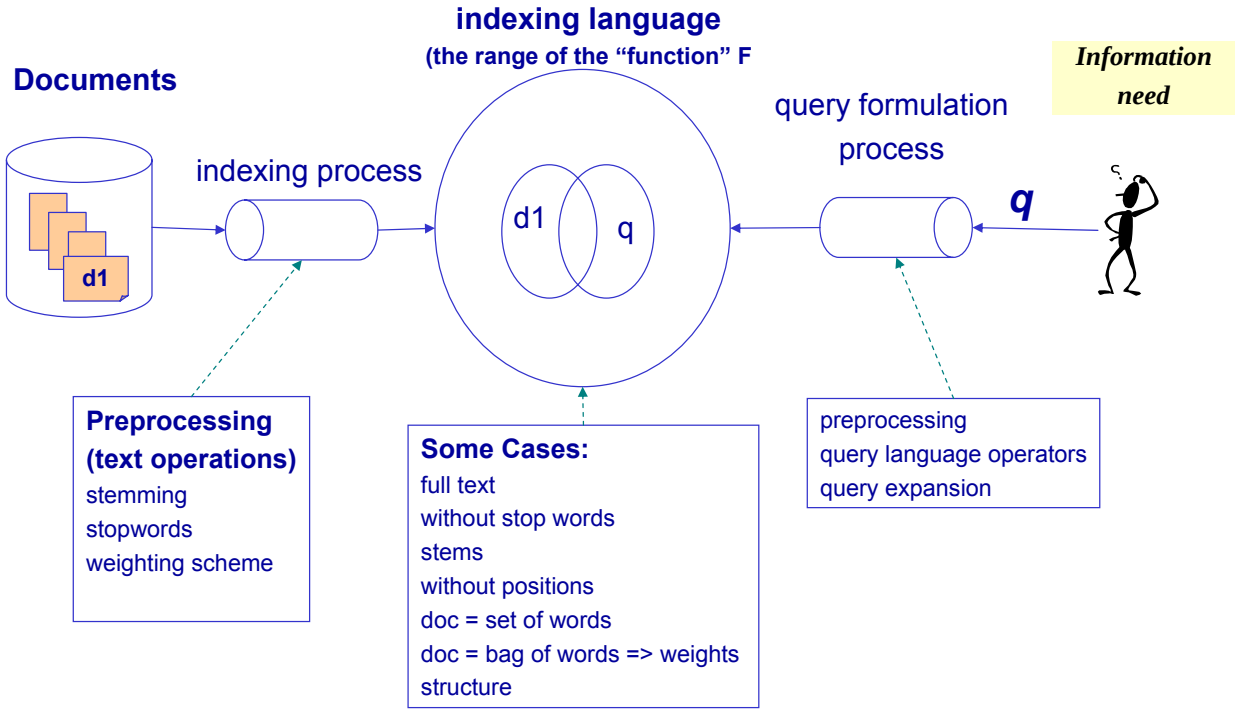
Μοντέλα Ανάκτησης

- 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])

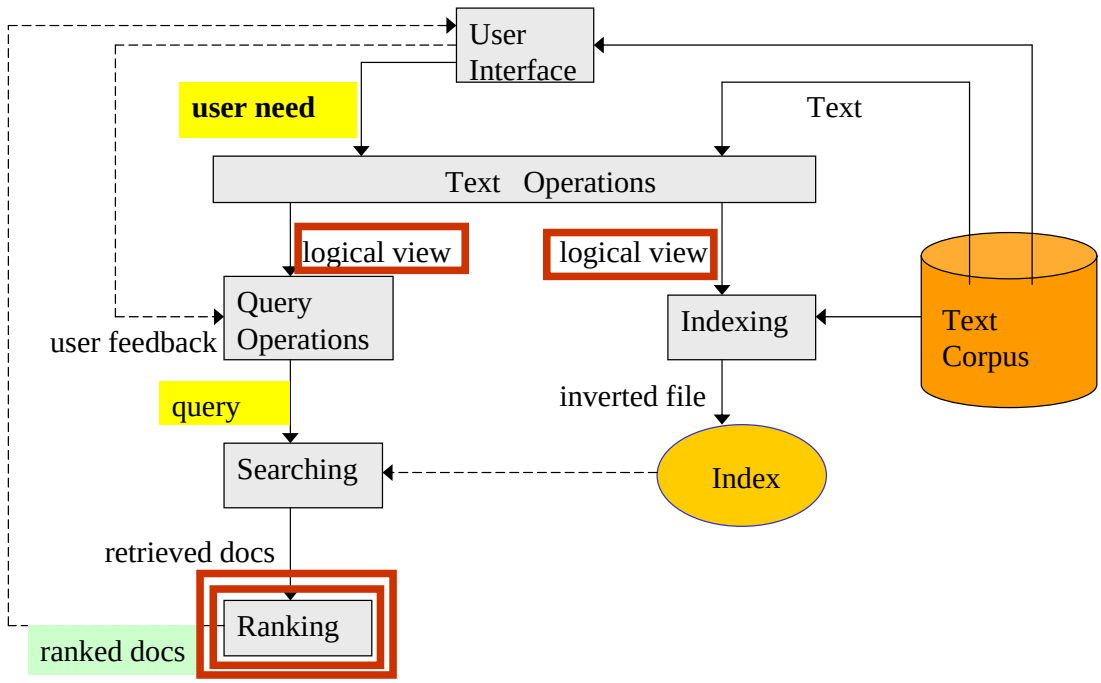
Έστω **D** η συλλογή εγγράφων και **Q** το σύνολο όλων των πληροφοριακών αναγκών που μπορεί να έχει ένας χρήστης.

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

- F: πλαίσιο μοντελοποίησης εγγράφων, επερωτήσεων και των σχέσεων μεταξύ τους
- D: παράσταση εγγράφων $D=\{ F(d) \mid \mathbf{d} \in \mathbf{D}\}$
- Q: παράσταση επερωτήσεων $Q=\{ F(q) \mid \mathbf{q} \in \mathbf{Q}\}$
- 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 $R: D \times Q \rightarrow [0,1]$



Τα τμήματα της αρχιτεκτονικής που εμπλέκονται



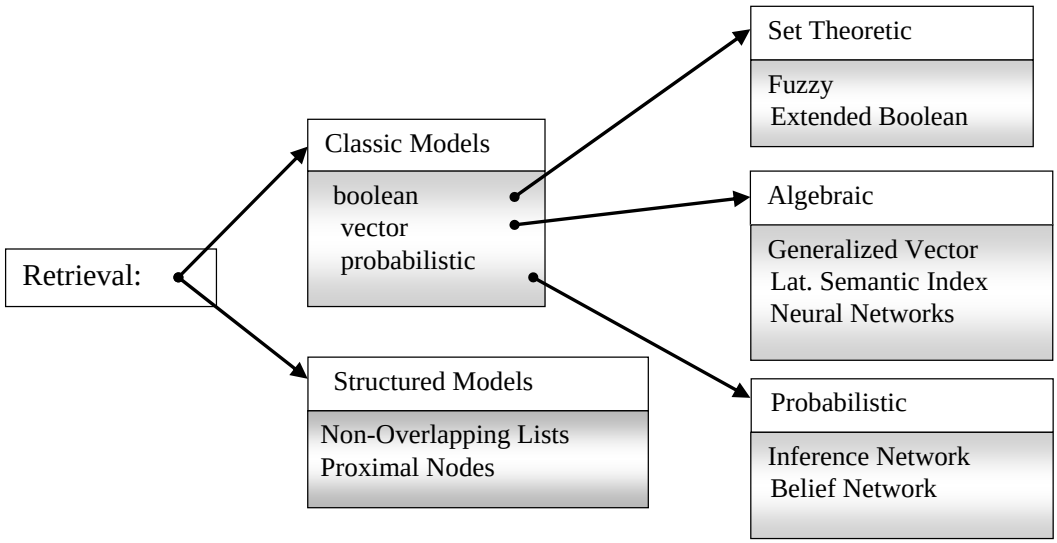
Κατηγορίες Μοντέλων Ανάκτησης (I)

- Κλασσικά Μοντέλα
  - Boolean Model
  - Διανυσματικό (Vector Space)
  - Πιθανοκρατικό (Probabilistic)
- Συνολοθεωρητικά (set theoretic)
  - Εκτεταμένο Boolean (Extended Boolean Model)
  - Fuzzy Model (Ασαφές Μοντέλο)
- Διανυσματικά (στατιστικά/αλγεβρικά)
  - Γενικευμένο Διανυσματικό (Generalized Vector Space Model)
  - Latent Semantic Indexing (Λανθάνων/Άδηλος/Υποβόσκων σημασιολογικός ευρετηριασμός)
  - Μοντέλο Νευρωνικού Δικτύου (Neural Network Model)

Κατηγορίες Μοντέλων Ανάκτησης (II)

- Πιθανοκρατικά (Probabilistic)
  - Inference Network Model (Μοντέλο Δικτύου Επαγωγών)
  - Belief Network Model (Μοντέλο Δικτύου Πεποιθήσεων)
- Μοντέλα Βασισμένα στη Λογική
- Μοντέλα Δομημένου Κειμένου (Structured Text Retrieval Models)
  - Non-Overlapping Lists
  - Proximal Nodes
  - Μοντέλα Ανάκτησης XML Εγγράφων

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

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

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

- 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 κείμενο **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 = "**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 κ$\}$

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:

$ans(q) = J(q)$

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

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

- $K=\{k_1,\ldots,k_t\}$: σύνολο όλων των λέξεων ευρετηρίασης
- Κάθε έγγραφο $\mathbf{d_j}$ παριστάνεται με ένα διάνυσμα $d_j=(w_{1,j},\ldots,w_{t,j})$ όπου $w_{i,j} \in [0,1]$ (πχ $w_{i,j}=0.3$)
- Μια επερώτηση q παριστάνεται με ένα διάνυσμα $q=(w_{1,q},\ldots,w_{t,q})$ 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

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

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

- 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

- d1 = { a a a b c }
- d2 = { a a a d e }
- d3 = { 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}

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

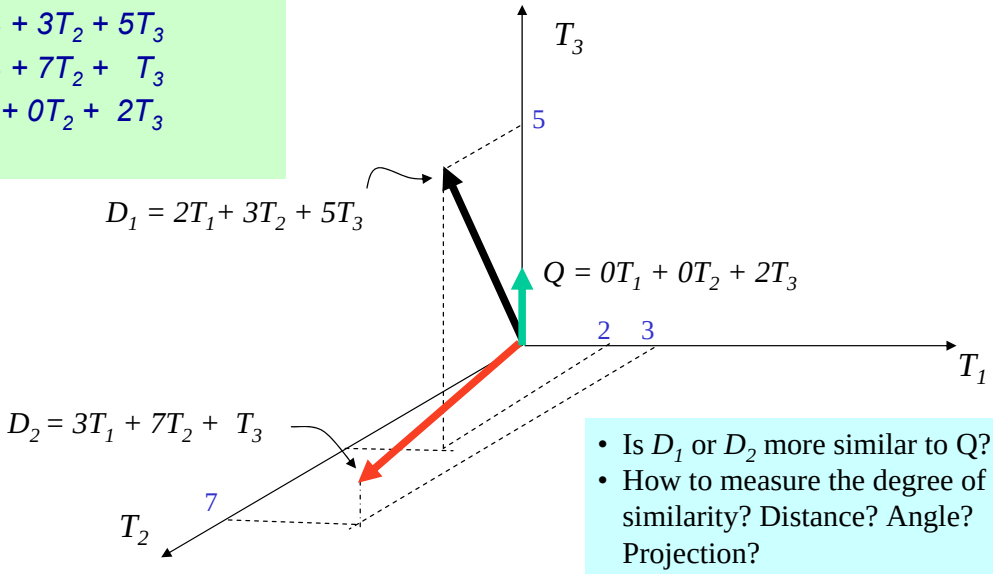
Διανυσματικό Μοντέλο:

- $K=\{k_1,\ldots,k_t\}$: σύνολο όλων των λέξεων ευρετηρίασης
- Κάθε έγγραφο $\mathbf{d_j}$ παριστάνεται με ένα διάνυσμα $d_j=(w_{1,j},\ldots,w_{t,j})$ όπου $w_{i,j} = \mathbf{tf_{ij}\ idf_i}$
- Μια επερώτηση q παριστάνεται με ένα διάνυσμα $q=(w_{1,q},\ldots,w_{t,q})$ όπου πάλι $w_{i,q} = \mathbf{tf_{iq}\ idf_i}$
- 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 wij το βάρος του όρου i στο έγγραφο j και wiq 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

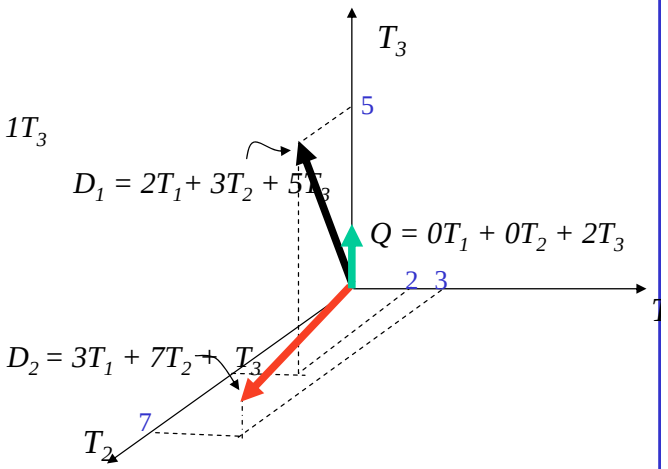
sim(d, q) = 3

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

Weighted:

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

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



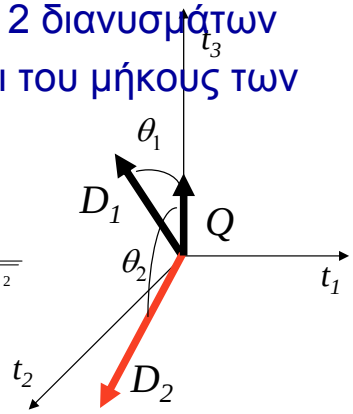
Ιδιότητες του Εσωτερικού Γινομένου

- 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

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



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

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 $w_{iq} = tf_{iq}\, idf_i = tf_{iq} \log_2 (N/ df_i)$
- $R(\boldsymbol{d_j},\mathbf{q}) = \text{CosSim}(\boldsymbol{d_j}, \boldsymbol{q}) = \dfrac{\vec{d}_j \cdot \vec{q}}{|\vec{d}_j| \cdot |\vec{q}|} = \dfrac{\sum_{i=1}^{t} (w_{ij} \cdot w_{iq})}{\sqrt{\sum_{i=1}^{t} w_{ij}^2 \cdot \sum_{i=1}^{t} w_{iq}^2}}$

Υπολογισμός του βαθμού συνάφειας
Απλοϊκή Υλοποίηση

1) Φτιάξε το *tf-idf* διάνυσμα για κάθε έγγραφο **dj** 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): $O(|V|\cdot|D|)$
Πολύ ακριβό αν τα V και D είναι μεγάλα!
$|V| = 10{,}000$; $|D| = 100{,}000$; $|V|\cdot|D| = 1{,}000{,}000{,}000$

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 $s_j = \text{cosSim}(d_j, q)$

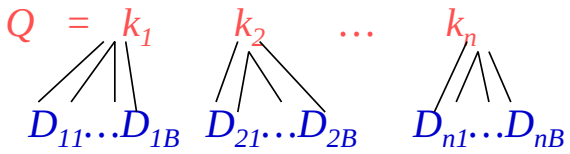
Απλοϊκό

↓

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 έγγράφων που έχουν μια λέξη είναι πολύ μικρότερο του πλήθους των εγγράφων της συλλογής.

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**
  - Συνδιασμός 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

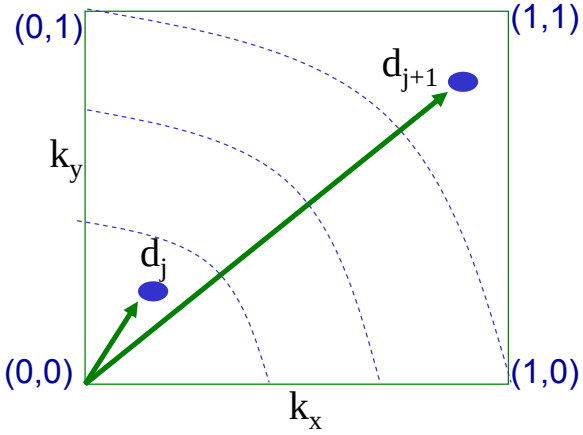
Μπορούμε να θεωρήσουμε κάθε όρο ως μια διάσταση
Άρα έγγραφα και επερωτήσεις απεικονίζονται στο 2D χώρο.

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 dj είναι οι (x,y)

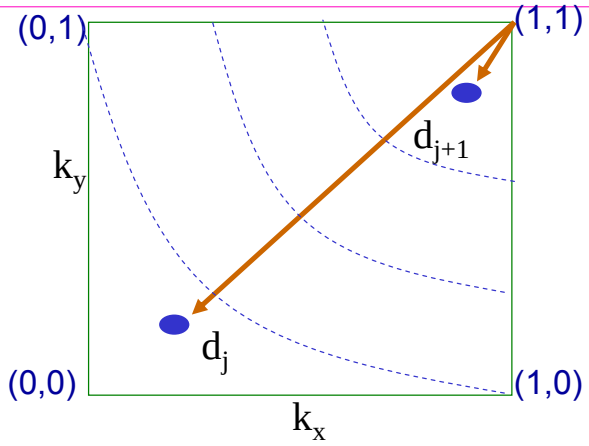
Η γενική ιδέα



Έστω $q_{OR}=k_x \vee k_y$

Το σημείο (0,0) είναι η θέση προς αποφυγή.

Άρα μπορούμε να θεωρήσουμε την *απόσταση* του dj από αυτό το σημείο ως το **βαθμό ομοιότητας**

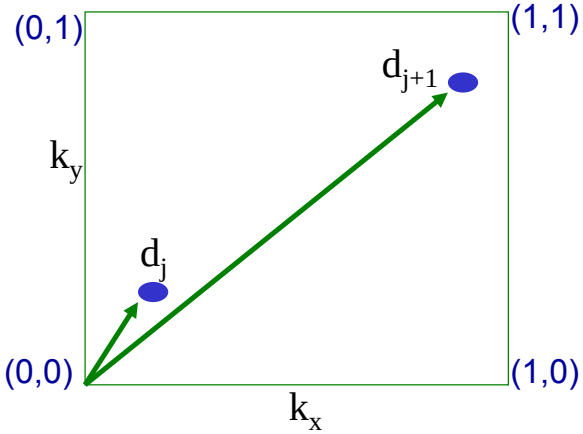


Έστω $q_{AND}=k_x \wedge k_y$

Το σημείο (1,1) είναι η πιο επιθυμητή θέση.

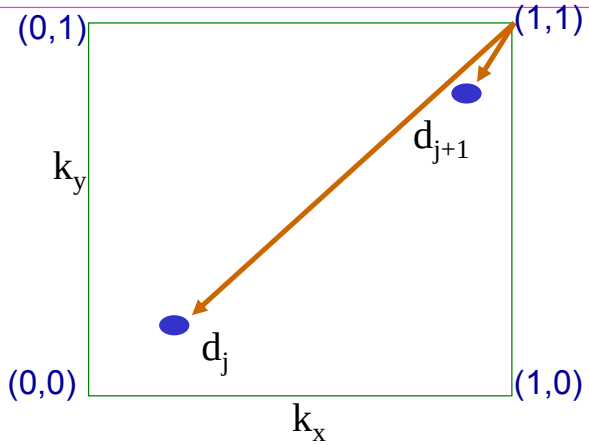
Άρα μπορούμε να θεωρήσουμε το *συμπλήρωμα της απόστασης* του dj από αυτό το σημείο ως **βαθμό ομοιότητας**

Η γενική ιδέα (II)



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

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



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

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

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

Γενικεύοντας την ιδέα (για >2 όρους)

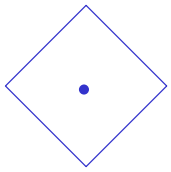
- Μπορούμε να γενικεύσουμε το προηγούμενο μοντέλο χρησιμοποιώντας την Ευκλείδεια απόσταση στον **t-διάστατο χώρο**
- Αυτό μπορεί να γίνει χρησιμοποιώντας ***p-norms*** που γενικεύουν την έννοια της απόστασης, όπου $1 \le p \le \infty$.
  - *Διαζευκτικές* επερωτήσεις
    - q_{OR} = k1 V k2 V . . V km
  - *Συζευκτικές* επερωτήσεις
    - q_{AND} = k1 Λ k2 Λ . . . Λ km

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

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

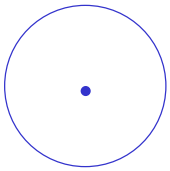
Ισομετρικές καμπύλες $\sqrt[p]{(x^p + y^p)}$

L_1



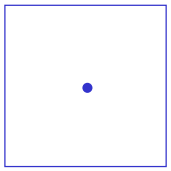
$x + y = 1$

L_2

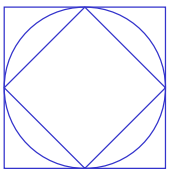


$\sqrt[2]{(x^2 + y^2)} = 1$

L_∞



$\max(x, y) = 1$



Μερικές ενδιαφέρουσες ιδιότητες

- Μεταβάλλοντας το *p*, μπορούμε να κάνουμε το μοντέλο να συμπεριφέρεται όπως το Vector, το Fuzzy (που θα δούμε στο επόμενο μάθημα), ή ενδιάμεσα σε αυτά τα δυο.

- Αν p = 1 τότε (Vector like)
  - $sim(q_{OR},dj) = sim(q_{AND},dj) = \frac{x1 + \ldots + xm}{m}$
- Αν p = ∞ τότε (Fuzzy like)
  - $sim(q_{OR},dj) = \max(x_i)$
  - $sim(q_{AND},dj) = \min(x_i)$

Ερώτηση: Που πήγαν οι όροι της επερώτησης;

Σύνθετες επερωτήσεις

- Έστω q = (k1 Λ k2) V k3
- 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 = (k1 V^2 k2) Λ^∞ k3
  - K1 and k2 should be used as in a vector system but the presence of k3 is required

Μερικές Παρατηρήσεις

- Είναι αρκετά ισχυρό μοντέλο με ενδιαφέρουσες ιδιότητες
- Η επιμεριστική ιδιότητα δεν ισχύει:
  - $q1 = (k1 \vee k2) \wedge k3$
  - $q2 = (k1 \wedge k3) \vee (k2 \wedge k3)$
  - $sim(q1,dj) \neq sim(q2,dj)$