CS-463 Information Retrieval Systems

Μοντέλα Ανάκτησης (Retrieval Models)

Part A

Yannis Tzitzikas

University of Crete

CS-463,Spring 05

Lecture : 3
Date : 1-3-2005

Διάρθρωση

PART (A)

- *Ανάκτηση και Φιλτράρισμα*
- *Εισαγωγή στα Μοντέλα Άντλησης*
- *Κατηγορίες Μοντέλων*
- *Exact vs Best Match*
- *Τα κλασσικά μοντέλα ανάκτησης*
  - *Το Boolean Μοντέλο*
  - *Στατιστικά Μοντέλα - Βάρυνση Όρων*
  - *Το Διανυσματικό Μοντέλο*
  - *Το Πιθανοκρατικό Μοντέλο*

PART (B):*Εναλλακτικά μοντέλα*

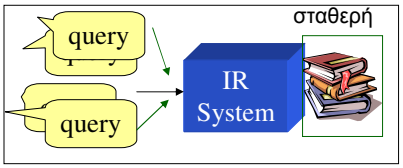
- *(I) Συνολοθεωρητικά μοντέλα*
  - *Fuzzy Retrieval Model*
  - *Extended Boolean Model*
- *(II) Αλγεβρικά Μοντέλα*
  - *Latent Semantic Indexing*
  - *Neural Netwok Model*

PART (C):

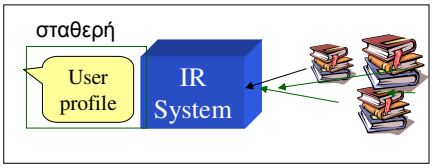
- *(III) Πιθανοκρατικά Μοντέλα*
  - *Inference Network Model*
  - *Belief Network Model*

Ανάκτηση και Φιλτράρισμα

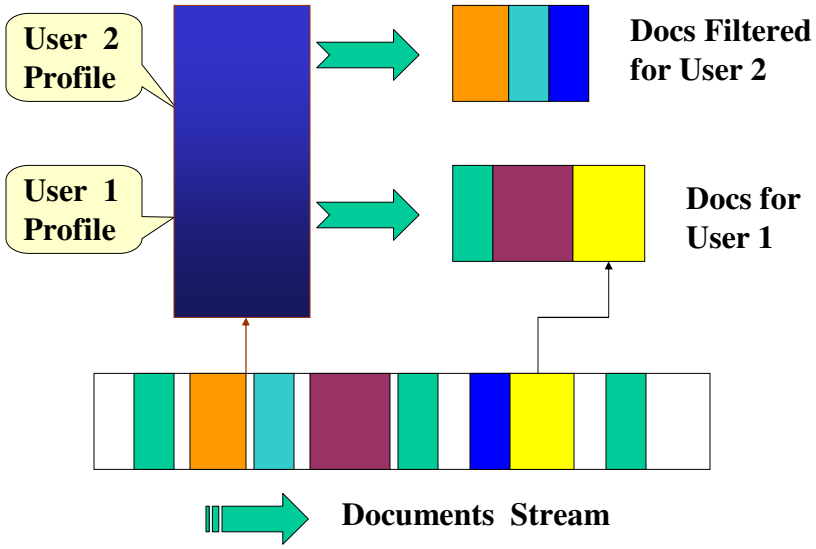
- **Ανάκτηση επί σκοπού (ad hoc retrieval)**:
  - Σταθερή συλλογή εγγράφων, μεταβαλλόμενες επερωτήσεις



- **Φιλτράρισμα(Filtering)**:
  - Σταθερή επερώτηση, **ροή** νέων κειμένων
  - **Προφίλ Χρήστη** = Επερώτηση που εκφράζει πιο μόνιμες προτιμήσεις
  - έμφαση στη δημιουργία/ενημέρωση του προφιλ
  - Routing: Όπως το φιλτράρισμα μόνο που εδώ το σύστημα δίδει διατεταγμένες λίστες



Φιλτράρισμα



Πως βλέπουμε ένα έγγραφο;

- Επιλογές
  - Όροι ευρετηρίασης (Index terms)
  - Πλήρες κείμενο (Full text)
  - Πλήρες κείμενο + Δομή (π.χ. hypertext, XML)

Μοντέλα Ανάκτησης

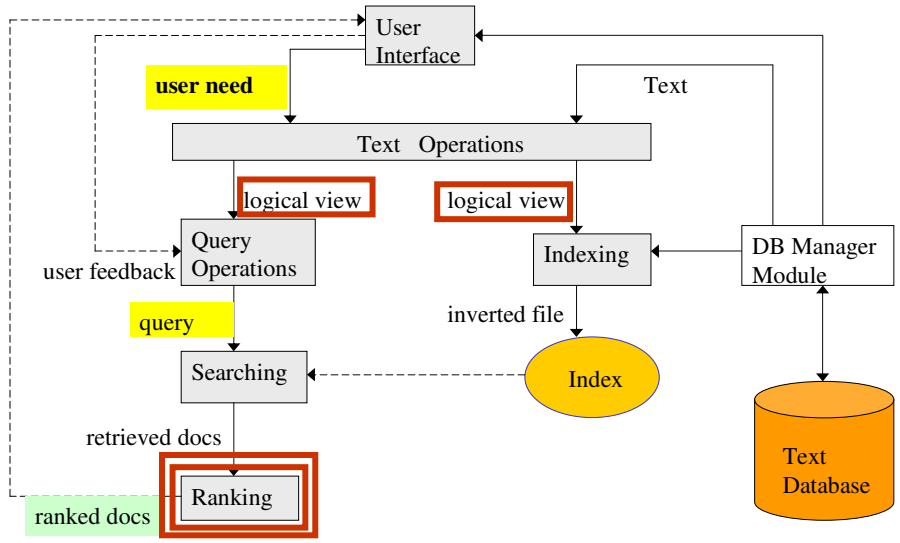
- Ένα μοντέλο ανάκτησης ορίζει
  - Αναπαράσταση Εγγράφων
  - Αναπαράσταση Επερωτήσεων
  - Καθορίζει και ποσοτικοποιεί την έννοια της συνάφειας
    - η βαθμός συνάφειας μπορεί να είναι δίτιμος (πχ. {1,0}), ή συνεχής(πχ [0,1])

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



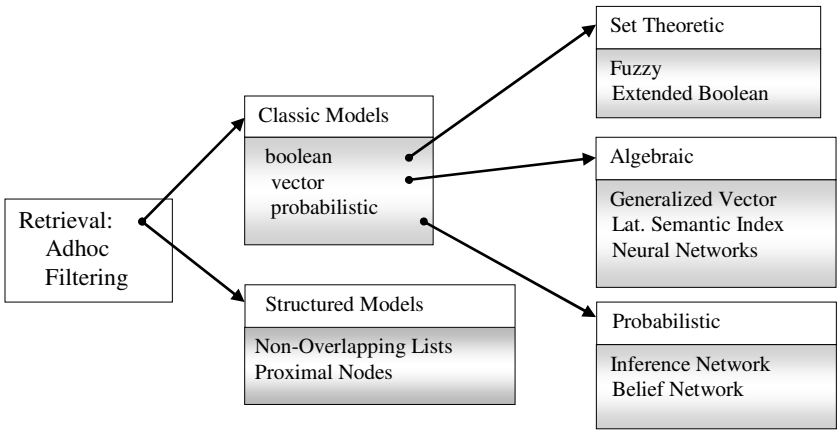
Κατηγορίες Μοντέλων Ανάκτησης (I)

- Κλασσικά Μοντέλα (3)
  - Boolean Model
  - Διανυσματικό (Vector Space)
  - Πιθανοκρατικό (Probabilistic)
- Συνολοθεωρητικά (set theoretic)(2)
  - Εκτεταμένο Boolean (Extended Boolean Model)
  - Fuzzy Model (Ασαφές Μοντέλο)
- Διανυσματικά (στατιστικά/αλγεβρικά) (3)
  - 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) (2)
  - 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>

Μια Ταξινομία των Μοντέλων Ανάκτησης



Exact vs. Best Match Retrieval Models

- Exact-match
  - μια επερώτηση καθορίζει **αυστηρά κριτήρια ανάκτησης**
  - κάθε έγγραφο **είτε ταιριάζει είτε όχι** με μία επερώτηση
  - το αποτέλεσμα είναι ένα **σύνολο** κειμένων
- Best-match
  - 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 μπορούμε να ελέγξουμε το μέγεθος της απάντησης
- «Μικτές προσεγγίσεις»
  - E.g., some type of ranking of result set (best of both worlds)
  - E.g., best-match query language that incorporates exact-match operators

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

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: Formally

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

Παράσταση εγγράφων κατά το Boolean Model

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

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

Ακριβές Ταίριασμα (Exact Match): Συν & Πλην

- Πλεονεκτήματα
  - Προβλέψιμο, εύκολα εξηγήσιμο
  - Αποτελεσματικό όταν γνωρίζεις ακριβώς τι ψάχνεις και τι περιέχει η συλλογή
  - Αποδοτική υλοποίηση
- Αδυναμίες
  - Η διατύπωση των επερωτήσεων είναι δύσκολη για πολλούς χρήστες
  - Ικανοποιητική ακρίβεια (precision) συχνά σημαίνει απαράδεκτη ανάκληση (recall)
  - Τα μοντέλα κατάταξης (ranking models) έχουν αποδειχτεί καλύτερα στην πράξη

Τα προβλήματα του Boolean Model

- Άκαμπτο: AND σημαίνει όλα, OR σημαίνει οποιοδήποτε
- Δυσκολίες
  - η έκφραση σύνθετων πληροφοριακών αναγκών
  - ο έλεγχος του μεγέθους της απάντησης
    - All matched documents will be returned
  - η διάταξη των αποτελεσμάτων
    - All matched documents logically satisfy the query
  - η υποστήριξη ανάδρασης συνάφειας
    - If a document is identified by the user as relevant or irrelevant, how should the query by modified ?

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 στοιχείων. :(

Στατιστικά Μοντέλα

- Έγγραφο: bag or words (unordered words with frequencies)
  - Bag = set that allows multiple occurences of the same element
- 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

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)

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

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*

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)

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

- 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 ένα έγγραφο που περιέχει όρους με τις εξής συχνότητες:
  - A(3), B(2), C(1), πχ. 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

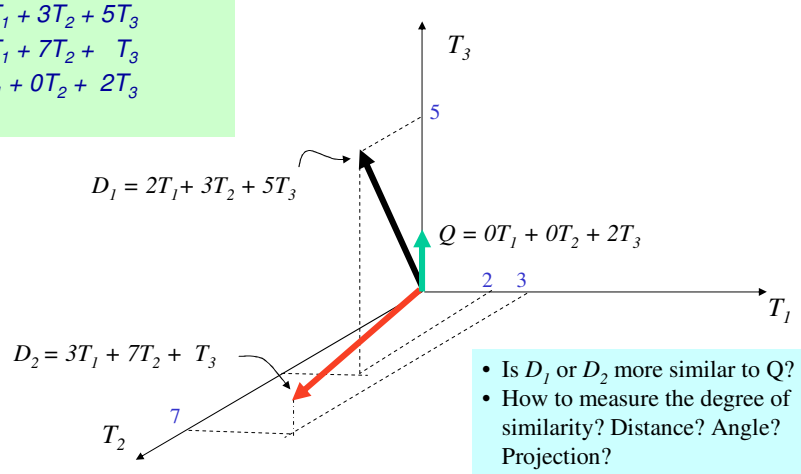
- Τα διανύσματα των επερωτήσεων θεωρούνται ως έγγραφα και επίσης βαρύνονται με tf-idf
- Εναλλακτικά, ο χρήστης μπορεί να δώσει τα βάρη των όρων της επερώτησης

Διανυσματικό Μοντέλο:

- $K=\{k_1,\ldots,k_t\}$: σύνολο όλων των λέξεων ευρετηρίασης
- Κάθε <u>έγγραφο</u> $\mathbf{d_j}$ παριστάνεται με ένα διάνυσμα $d_j=(w_{1,j},\ldots,w_{t,j})$ όπου $w_{i,j} = \mathbf{tf_{ij}\ idf_i}$
- Μια <u>επερώτηση</u> 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

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:

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

- όπου wij το βάρος του όρου i στο έγγραφο j και wiq το βάρος του όρου i στην επερώτηση
- 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

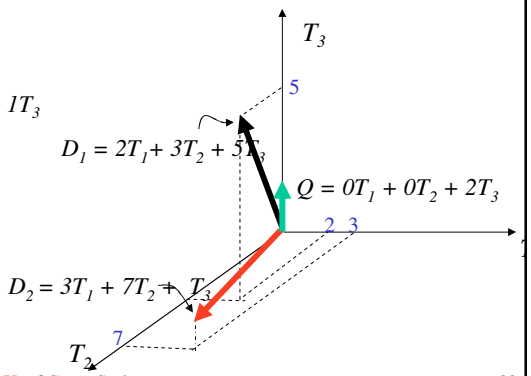
sim(d, q) = 3

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

Weighted:

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

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



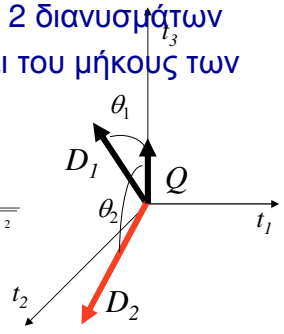
Ιδιότητες του Εσωτερικού Γινομένου

- 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

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



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

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

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: O(|V|·|D|)
Κακό για μεγάλο V & D !
|V| = 10,000; |D| = 100,000; |V|·|D| = 1,000,000,000

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

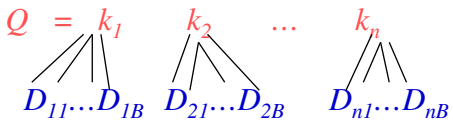
- => Μπορούμε να χρησιμοποιήσουμε ένα ευρετήριο ώστε να υπολογίσουμε το βαθμό ομοιότητας μόνο εκείνων των εγγράφων που περιέχουν τουλάχιστον έναν όρο της επερώτησης.

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

↓

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

Γρηγορότερη Υλοποίηση (II)



- Ας υποθέσουμε ότι ένας όρος της επερώτησης εμφανίζεται σε *B* έγγραφα
- Τότε η χρονική πολυπλοκότητα είναι O(|*Q*| *B*)
- Συνήθως είναι πολύ μικρότερη του απλοϊκού τρόπου (που είχε πολυπλοκότητα O(|V||D|), διότι:
  - |*Q*| << |*V*| και
  - *B* << |D|.

Διάρθρωση Διάλεξης

PART (A)
- *Ανάκτηση και Φιλτράρισμα*
- *Εισαγωγή στα Μοντέλα Άντλησης*
- *Κατηγορίες Μοντέλων*
- *Exact vs Best Match*
- *Τα κλασσικά μοντέλα ανάκτησης*
  - *Το Boolean Μοντέλο*
  - *Στατιστικά Μοντέλα - Βάρυνση Όρων*
  - *Το Διανυσματικό Μοντέλο*
  - **Το Πιθανοκρατικό Μοντέλο**

PART (B):***Εναλλακτικά μοντέλα***
- *(I) Συνολοθεωρητικά μοντέλα*
  - *Fuzzy Retrieval Model*
  - *Extended Boolean Model*
- *(II) Αλγεβρικά Μοντέλα*
  - *Latent Semantic Indexing*
  - *Neural Netwok Model*

PART (C):
- *(III) Πιθανοκρατικά Μοντέλα*
  - *Inference Network Model*
  - *Belief Network Model*

Information Retrieval Models

Probabilistic Model

The Probabilistic Model
(πιθανοκρατικό μοντέλο)

- Στόχος:
  - σύλληψη του προβλήματος της ΑΠ με χρήση πιθανοτήτων
- Προσέγγιση
  - Υπόθεση: Για κάθε επερώτηση υπάρχει μια **ιδανική** απάντηση
  - Υπολογισμός της απάντησης βάσει των *ιδιοτήτων* (όρων) της ιδανικής απάντησης
  - Κρίσιμο ερώτημα: Ποιες είναι αυτές οι ιδιότητες;
  - Προσέγγιση: Μάντεψε αρχικά και κατόπιν βελτίωσε με επαναλήψεις

Τα βήματα

1/ Ένα αρχικό σύνολο εγγράφων ανακτάται με κάποιο τρόπο

2/ Ο χρήστης τα παρατηρεί (πχ τα πρώτα 10-20) και μαρκάρει τα συναφή

3/ Το ΣΑΠ αξιοποιεί αυτά τα έγγραφα για να εκλεπτύνει την περιγραφή της ιδανικής απάντησης

4/ Επαναλαμβάνοντας αυτή τη διαδικασία αναμένουμε ότι η περιγραφή της ιδανικής απάντησης όλο και βελτιώνεται

Παρατηρήσεις

- Πάντα πρέπει να μαντέψουμε αρχικά την περιγραφή της ιδανικής απάντησης
- Η περιγραφή της ιδανικής απάντησης γίνεται πιθανοκρατικά

Πιθανοκρατική Διάταξη

Έστω επερώτηση q και έγγραφο dj

Το πιθανοκρατικό μοντέλο προσπαθεί να εκτιμήσει την πιθανότητα να βρει ο χρήστης το έγγραφο dj συναφές με την επερώτηση q.

Το μοντέλο κάνει την υπόθεση ότι αυτή η πιθανότητα εξαρτάται **μόνο** από το dj και q (και όχι από τα υπόλοιπα έγγραφα)

Η ιδανική απάντηση (σύνολο συναφών εγγράφων) συμβολίζεται με R.

Ερωτήματα:

πώς να υπολογίσουμε τις πιθανότητες;

ποιος ο δειγματικός χώρος;

Πιθανοκρατική Διάταξη (II)

- Η πιθανοκρατική διάταξη ορίζεται ως εξής:
  - **sim(dj,q) = P(dj relevant-to q) / P(dj non-relevant-to q)**
  - This is the odds of the document dj being relevant
  - Taking the odds minimize the probability of an erroneous judgement
- Ορισμοί:
  - *wij* $\in$ *{0,1}*
  - Μία επερώτηση είναι ένα **σύνολο** όρων.
  - P(R | vec(dj)) : πιθανότητα το doc να είναι συναφές (να ανήκει στην ιδανική απάντηση R)
  - P(¬R | vec(dj)) : πιθανότητα το doc να **μην** είναι συναφές

Πιθανοκρατική Διάταξη (III)

sim(dj,q) = P(R | vec(dj)) / P(¬R | vec(dj))

Bayes Rule:
$P(H|e) = \frac{P(e|H)\,P(H)}{P(e)}$

$= \frac{[P(vec(dj) \mid R) * P(R)]}{[P(vec(dj) \mid \neg R) * P(\neg R)]}$ (Bayes' rule)

$\sim \frac{P(vec(dj) \mid R)}{P(vec(dj) \mid \neg R)}$ (υποθέτοντας P(R)=P(¬ R))

όπου:

P(R) : πιθανότητα ένα τυχαίο επιλεγμένο έγγραφο να είναι συναφές

P(¬ R) : πιθανότητα ένα τυχαίο επιλεγμένο έγγραφο να μην είναι συναφές

P(vec(dj) | R) : πιθανότητα επιλογής του dj από το σύνολο R των συναφών

Πιθανοκρατική Διάταξη (IV)

•Ερώτημα:

- Πως μπορούμε να υπολογίσουμε το **P(vec(dj) | R)** ;
- Έστω vec(dj)= <1,0,1,0>, ποια η τιμή του P(vec(dj)|R);

•Τρόπος:

- Έστω **P(ki|R)** : η πιθανότητα εμφάνισης του όρου κi σε ένα τυχαία επιλεγμένο έγγραφο από το σύνολο των συναφών εγγράφων R
- •Με υπόθεση ανεξαρτησίας όρων παίρνουμε:

$$P(vec(dj) \mid R) =$$
$$P(\langle 1,0,1,0\rangle \mid R) =$$
$$P(k1|R) * P(\neg k2|R) * P(k3|R) * P(\neg k4|R)$$

Πιθανοκρατική Διάταξη (V)

Αρα:

$$sim(dj,q) \sim \frac{P(vec(dj) \mid R)}{P(vec(dj) \mid \neg R)}$$

(assuming term independence)

$$\sim \log \frac{[\prod_{wij=1} P(ki \mid R)] * [\prod_{wij=0} P(\neg ki \mid R)]}{[\prod_{wij=1} P(ki \mid \neg R)] * [\prod_{wij=0} P(\neg ki \mid \neg R)]}$$

Πιθανοκρατική Διάταξη (VI)

- $sim(dj,q) \sim \log \frac{[\Pi P(ki \mid R)] * [\Pi P(\neg kj \mid R)]}{[\Pi P(ki \mid \neg R)] * [\Pi P(\neg kj \mid \neg R)]}$

$$\sim \sum wiq * wij * \left(\log \frac{P(ki \mid R)}{P(\neg ki \mid R)} + \log \frac{P(ki \mid \neg R)}{P(\neg ki \mid \neg R)}\right)$$

όπου $P(\neg ki \mid R) = 1 - P(ki \mid R)$
$P(\neg ki \mid \neg R) = 1 - P(ki \mid \neg R)$

•***Ερώτημα: από πού ήρθε το wiq ?***

Η Αρχική Διάταξη

- Ποιές είναι οι τιμές των P(ki | R) και P(ki | ¬R) ?
- Έστω
  - **N** το πλήθος των εγγράφων, και
  - **ni** το πλήθος των εγγράφων που περιέχουν τον όρο ki
- Κάνουμε μια εκτίμηση βασισμένη στις υποθέσεις:
  - **P(ki | R) = 0.5**
  - **$P(ki \mid \neg R) = \frac{ni}{N}$** *(το ποσοστό των εγγράφων του έχουν τον όρο ki)*
  - Χρησιμοποιούμε αυτό το αρχικό μάντεμα για να ανακτήσουμε μια αρχική διάταξη (**εγγράφων που περιέχουν τους όρους της επερώτησης**)
  - Κατόπιν βελτιώνουμε τη διάταξη

Βελτίωση της Αρχικής Διάταξης (χωρίς επέμβαση του χρήστη)

- Έστω
  - V : το σύνολο των εγγράφων που ανακτήθηκαν αρχικά
  - Vi : το υποσύνολο των εγγράφων του V που περιέχουν τον όρο ki
- Επαναποτίμηση:
  - $P(ki \mid R) = \frac{Vi}{V}$ (*το ποσοστό των ανακτημένων που περιέχουν το ki)*
  - $P(ki \mid \neg R) = \frac{ni - Vi}{N - V}$ (*μη ανακτημένα έγγραφα που περιέχουν τον ki) / (μη ανακτημένα έγγραφα)*
- Επανέλαβε αναδρομικά

Βελτίωση της Αρχικής Διάταξης

- Για να μην έχουμε πρόβλημα στην περίπτωση που V=1 και Vi=0, ορίζουμε:
  - $P(ki \mid R) = \frac{Vi + 0.5}{V + 1}$
  - $P(ki \mid \neg R) = \frac{ni - Vi + 0.5}{N - V + 1}$
- Εναλλακτικά:
  - $P(ki \mid R) = \frac{Vi + ni/N}{V + 1}$
  - $P(ki \mid \neg R) = \frac{ni - Vi + ni/N}{N - V + 1}$

Συν και Πλην του Πιθανοκρατικού Μοντέλου

- Συν:
  - Διάταξη εγγράφων ως προς την **πιθανότητα** συνάφειας τους
- Πλην
  - ανάγκη μαντέματος των P(ki | R) για κάθε ki
  - δεν λαμβάνονται υπόψη τα tf και idf

Σύντομη Σύγκριση των Κλασσικών Μοντέλων

- Το Boolean model δεν υποστηρίζει μερικό ταίριασμα και είναι το πιο αδύνατο από τα κλασσικά μοντέλα
- Μια σειρά πειραμάτων απέδειξε ότι γενικά το **διανυσματικό μοντέλο είναι αποτελεσματικότερο** του πιθανοκρατικού [Salton & Buckley]
- Η ερευνητική κοινότητα φαίνεται να συμμερίζεται αυτήν την άποψη