Πανεπιστήμιο Κρήτης, Τμήμα Επιστήμης Υπολογιστών
ΗΥ463 - Συστήματα Ανάκτησης Πληροφοριών
2004-2005 Εαρινό Εξάμηνο

**2η Σειρά ασκήσεων**
**(Μοντέλα Ανάκτησης Πληροφοριών και Ευρετήρια)**
Ανάθεση: 16 Μαρτίου
Παράδοση: 29 Μαρτίου

## *Άσκηση 1 (40 βαθμοί) (Διανυσματικό Μοντέλο)*

Θεωρείστε μια συλλογή κειμένων που περιέχει τα ακόλουθα 5 έγγραφα:

Έγγραφο 1: «New Year»
Έγγραφο 2: « New Year New Year »
Έγγραφο 3: «Financial New Times»
Έγγραφο 4: «Financial Year»

1) Δώστε τη διανυσματική παράσταση του κάθε εγγράφου με βάρη TF-IDF (για ευκολία θεωρήστε ότι IDF=N/DF και όχι IDF=log(N/DF)). Θεωρείστε ότι η θέση της κάθε λέξης στα διανύσματα γίνεται κατά αλφαβητική σειρά.

2) Θεωρείστε την επερώτηση q1=«new financial». Υπολογίστε το TF-IDF διάνυσμα αυτής της επερώτησης και δώστε την διάταξη των εγγράφων που θα επιστρέψει ένα σύστημα που βασίζεται στο διανυσματικό μοντέλο.

3) Σχεδιάστε το ανεστραμμένο αρχείο για αυτή τη συλλογή.

## *Λύση*

1. http://www.csd.uoc.gr/~hy463/2006/download/tutorials/tutorial2.pdf

α)

| | **Financial** | **New** | **Times** | **Year** | **MAXk {FREQij}** |
|---|---|---|---|---|---|
| **$D_1$** | 0 | 1 | 0 | 1 | 1 |
| **$D_2$** | 0 | 2 | 0 | 2 | 2 |
| **$D_3$** | 1 | 1 | 1 | 0 | 1 |
| **$D_4$** | 1 | 0 | 0 | 1 | 1 |
| **DF** | 2 | 3 | 1 | 3 | |
| **IDF** | 4/2 | 4/3 | 4/1 | 4/3 | |

| | Financial | New | Times | Year | MAXk {FREQij} |
|---|---|---|---|---|---|
| **$D_1$** | 0 | 1/1*4/3 | 0 | 1/1*4/3 | 1 |
| **$D_2$** | 0 | 2/2*4/3 | 0 | 2/2*4/3 | 2 |
| **$D_3$** | 1/1*4/2 | 1/1*4/3 | 1/1*4/1 | 0 | 1 |
| **$D_4$** | 1/1*4/2 | 0 | 0 | 1/1*4/3 | 1 |
| **DF** | 2 | 3 | 1 | 3 | |
| **IDF** | 4/2=2 | 4/3 | 4/1=4 | 4/3 | |

Οι διανυσματικές αναπαραστάσεις των κειμένων είναι:

$W_1$={0,1.33,0,1.33}, |W1|=5,7689
$W_2$={0,1.33,0,1.33}, |W2|=3,5378
$W_3$={2,1.33,4,0}, |W3|=21,7689
$W_4$={2,0,0,1.33}, |W4|=4

β)

| | Financial | New | Times | Year |
|---|---|---|---|---|
| **$Q_1$= new financial** | 1/1*4/2 | 1/1*4/3 | 0 | 0 |
| **IDF** | 4/2=2 | 4/3 | 4/1=4 | 4/3 |

$Q_1$ = {2,1.33,0,0}, |$Q_1$| = 1.7689
$W_1$*$Q_1$=1.7689
$W_2$*$Q_1$=1.7689
$W_3$*$Q_1$=5.7689
$W_4$*$Q_1$=4

Μέτρο ομοιότητας Συνημίτονου

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

$R(D_1,Q_1)$= 1.7689/(sqrt(5.7689,3.5378))=0.392
$R(D_2,Q_1)$= 1.7689/(sqrt(5.7689,3.5378))=0.392
$R(D_3,Q_1)$= 1.7689/(sqrt(5.7689,21.7689))=0.515
$R(D_4,Q_1)$= 1.7689/(sqrt(5.7689,5.7689))=0.692

Η διάταξη των κειμένων θα είναι $D_4,D_3,\{D_1,D_2\}$. Το αναμενόμενο αποτέλεσμα θα ήταν να έρθει πρώτο στη διάταξη μας το $D_3$ όμως αυτό δεν συνέβη διότι το βάρος του εγγράφου επηρεάστηκε από τον όρο Times ο οποίος εμφανίζεται μονάχα μια φορά στη συλλογή των κειμένων μας.

Εναλλακτικά σαν μέτρο ομοιότητας μπορούμε να χρησιμοποιήσουμε το εσωτερικό γινόμενο.

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

γ)

| **Term** | **<Frequency, (Document; Position)** |
|---|---|
| financial | <2 ($D_3$;1), ($D_4$;1)> |
| new | <4 ($D_1$;1), ($D_2$;1), ($D_2$;3), ($D_3$;2)> |
| times | <1 ($D_3$;3)> |
| year | <4 ($D_1$;2), ($D_2$;2), ($D_2$;4), ($D_4$;2)> |

## *Άσκηση 2 (40 βαθμοί) (συνάρτηση διαβάθμισης)*

Θεωρείστε ένα Σύστημα Ανάκτησης Πληροφοριών (ΣΑΠ) από μια μεγάλη συλλογή κειμένων. Θέλουμε να δώσουμε τη δυνατότητα χρήσης του ΣΑΠ μέσω κινητού τηλεφώνου. Για το λόγο αυτό θέλουμε να ορίσουμε μια συνάρτηση διαβάθμισης (ranking function) η οποία να ευνοεί τα μικρά κείμενα, αφενός για να κρατήσουμε σε χαμηλά επίπεδα τον όγκο δεδομένων που θα μεταφέρονται και αφετέρου διότι οι χρήστες κινητών τηλεφώνων προτιμούν τα μικρά κείμενα (ένεκα του μικρού μεγέθους της οθόνης). Θεωρείστε ότι οι επερωτήσεις των χρηστών είναι σάκοι λέξεων (bag of words). Σχεδιάστε μια συνάρτηση διαβάθμισης για το σκοπό αυτό για κάθε μια από τις παρακάτω περιπτώσεις
(α) Το ευρετήριο του ΣΑΠ έχει δυαδικά (0,1) βάρη (όπως για παράδειγμα το ευρετήριο του Boolean μοντέλου)
(β) Το ευρετήριο έχει βάρη TF-IDF.
Τεκμηριώστε τις προτάσεις σας (με αποδείξεις ή παραδείγματα).

## *Λύση*

α) Θέλουμε να τροποποιήσουμε το Boolean μοντέλο έτσι ώστε να ευνοεί τα μικρότερα κείμενα. Η συνάρτηση που θα χρησιμοποιήσουμε είναι η R(d,q) = |d ∩q|/|d| η οποία κανονικοποιεί την συνάρτηση που εκφράζει τη συσχέτιση ενός κειμένου με μια επερώτηση με βάση το μέγεθος του κειμένου.

Γνωρίζουμε ότι R(d,q) = |d ∩q|/|d| = |d ∩q|/(|d ∩q|+|d\q|).

Αν η τομή *d∩q* είναι σταθερή, δηλαδή η έγγραφα έχουν την ίδια συνάφεια, τότε αυτό που έχει μεγαλύτερο μέγεθος θα διαβαθμιστεί πιο χαμηλά, διότι θα μεγαλώσει ο παρανομαστής άρα η συνάρτηση θα επιστρέψει μικρότερη τιμή διαβάθμισης. Στη γενική περίπτωση που έχουμε διαφορετικές συνάφειες τα μικρότερα έγγραφα ευνοούνται έναντι
των μεγαλύτερων.

Πειραματικά αυτό σημαίνει:

| | |
|---|---|
| q = “a b” | R(d,q) = \|d ∩q\|/\|d\| |
| d1 = “a” | 1/1=1 |
| d2 = “a c” | 1/2=0.5 |
| d3 = “a c d” | 1/3=0.33 |
| d4 = “a b c” | 2/3=0.66 |
| d5 = “a b c d a“ | 2/5 = 0.4 |
| Διάταξη εγγράφων | <d1, d4, d2, d5, d3> |

β) Γενικεύουμε την ιδέα του (α) για την περίπτωση που έχουμε μη δυαδικά βάρη. Συγκεκριμένα μπορούμε να ορίσουμε: R(d,q) = d*q / ||d|| (όπου το * είναι το εσωτερικό γινόμενο) όπου το εσωτερικό γινόμενο υπολογίζεται από τον τύπο: $\sum_{i=1}^{t} (w_{ij} \cdot w_{iq})$
και τα $w_{i,j}=tf_{ij}*idf_i$ και $w_{i,q} = tf_{iq}*idf_i$

| | **a** | **b** | **C** | **d** | **MAXk {FREQij}** |
|---|---|---|---|---|---|
| **$D_1$** | 1 | 0 | 0 | 0 | 1 |
| **$D_2$** | 1 | 0 | 1 | 0 | 1 |
| **$D_3$** | 1 | 0 | 1 | 1 | 1 |
| **$D_4$** | 1 | 1 | 1 | 0 | 1 |
| **$D_5$** | 2 | 1 | 1 | 1 | 2 |
| **TF** | 4 | 2 | 4 | 2 | |
| **IDF** | 5/5=1 | 5/2 | 5/4 | 5/2 | |

| | a | b | C | D | MAXk {FREQij} |
|---|---|---|---|---|---|
| **$D_1$** | 1/1*4/4=1 | 0 | 0 | 0 | 1 |
| **$D_2$** | 1/1*4/4=1 | 0 | 1/1*4/4=1 | 0 | 1 |
| **$D_3$** | 1/1*4/4=1 | 0 | 1/1*4/4=1 | 1/1*4/2=2 | 1 |
| **$D_4$** | 1/1*4/4=1 | 1/1*4/2=2 | 1/1*4/4=1 | 0 | 1 |
| **$D_5$** | 2/2*4/4=1 | 1/1*4/2=2 | 1/1*4/4=1 | 1/1*4/2=2 | 1 |
| **Q1** | 1/1*4/4=1 | 1/1*4/2=2 | 0 | 0 | |
| **TF** | 4 | 2 | 4 | 2 | |
| **IDF** | 4/4=1 | 4/2=2 | 4/4=1 | 4/2=2 | |

$W_1*Q_1=(1,0,0,0)*(1,2,0,0)=1$
$W_2*Q_1=(1,0,1,0)*(1,2,0,0)=1$
$W_3*Q_1=(1,0,1,2)*(1,2,0,0)=1$
$W_4*Q_1=(1,2,1,0)*(1,2,0,0)=5$
$W_5*Q_1=(1,2,1,2)*(1,2,0,0)=5$

Εφαρμόζουμε τη συνάρτηση για τον υπολογισμό της συνάφειας:

$R(D_1,Q_1)=1/1=1$
$R(D_2,Q_1)=1/2=0.5$
$R(D_3,Q_1)=1/3=0.33$
$R(D_4,Q_1)=5/3=1.66$
$R(D_5,Q_1)=5/5=1$

Επομένως η διάταξη που επιστρέφει η συνάρτηση μας είναι η D4, {D5,D1}, D2, D3. Παρατηρούμε ότι από τα κείμενα που είναι ποιο σχετικά στο ερώτημα μας (δηλαδή περιέχουν και τους δύο όρους) ευνοήθηκε αυτό με το μικρότερο πλήθος όρων.

## *Άσκηση 3 (20 βαθμοί)*

Έστω ένα ΣΑΠ που βασίζεται στο διανυσματικό μοντέλο, το οποίο υποστηρίζει τον κλασσικό τρόπο αλληλεπίδρασης (ο χρήστης διατυπώνει επερώτηση και το σύστημα επιστρέφει ένα διατεταγμένο σύνολο εγγράφων) συν μια λειτουργία *προσαρμογής ευρετηρίου*. Συγκεκριμένα ο χρήστης μπορεί να αλλάξει τη θέση ενός εγγράφου που εμφανίζεται σε μια απάντηση (π.χ. από τη $2^η$ θέση να το πάει στην $1^η$ ή στην $18^η$). Μετά από μια τέτοια εντολή, το σύστημα πρέπει να «προσαρμοστεί» στην απαίτηση του χρήστη, τροποποιώντας κατάλληλα το διάνυσμα του

εγγράφου που μετακινήθηκε. Η τροποποίηση πρέπει να είναι τέτοια ώστε αν ο χρήστης επανυποβάλει την επερώτηση, τότε το εν λόγω έγγραφο να τοποθετηθεί στη θέση που όρισε ο χρήστης.
Περιγράψτε με ποιο τρόπο θα τροποποιούσατε το διάνυσμα του εγγράφου προκειμένου να επιτύχετε την παραπάνω λειτουργικότητα.
Λάβετε υπόψη ότι αν υπάρχουν πολλοί τρόποι υλοποίησης μιας λειτουργίας προσαρμογής, τότε ως κριτήριο για την επιλογή του καταλληλότερου τρόπου συχνά θεωρείται η αρχή της ελάχιστης αλλαγής.
Υπόδειξη: Θεωρείστε αρχικά ότι το πλήθος των όρων είναι 1, κατόπιν 2 και εν συνεχεία γενικεύστε.

**Λύση**

Θα αντιμετωπίσουμε το πρόβλημα πάνω σε 2 άξονες. Αρχικά θα το μοντελοποιήσουμε με μαθηματικά και έπειτα θα δώσουμε έναν αλγόριθμο που θα προσπαθεί να το επιλύσει. Σημειώστε ότι το πρώτο δεν οδηγεί απευθείας (σίγουρα) στο δεύτερο. Θα αρχίσουμε όμως προσεγγίζοντας το πρόβλημα διαισθητικά.

Όταν ο χρήστης επανακατατάσει ένα έγγραφο, ουσιαστικά θέλει να αλλάξει το ranking του, για αυτήν την συγκεκριμένη επερώτηση. Έτσι εμείς θέλουμε να τροποποιήσουμε το διάνυσμα του εγγράφου έτσι ώστε στην επόμενη ίδια επερώτηση το έγγραφο να παρουσιαστεί καταταγμένο στην προτιμητέα θέση του χρήστη. Θέλουμε όμως ταυτόχρονα *το διάνυσμα να μην αλλάξει πολύ, ώστε σε διαφορετικές επερωτήσεις το διάνυσμα να παραμένει να εμφανίζεται σχεδόν στις ίδιες θέσεις που εμφανιζόταν έως τώρα*. Άρα ψάχνουμε ένα καινούριο διάνυσμα d' που να διαφέρει όσο γίνεται λιγότερο από το παλιό.

Για να καταλάβουμε την θέση που θέλει ο χρήστης να τοποθετήσει το document, μπορούμε απλά να δούμε τα rankings των εγγράφων πάνω και κάτω από την καινούρια θέση που τοποθετήσαμε το document και να θεωρήσουμε ότι το διάνυσμα του εγγράφου μας πρέπει να τροποποιηθεί έτσι ώστε η συνάρτηση συνάφειας να μας δίνει τιμή ανάμεσα στις τιμές του αμέσως επόμενου και αμέσως προηγούμενου εγγράφου. Για παράδειγμα ας θεωρήσουμε το διάνυσμα του εγγράφου, που θέλει ο χρήστης να επανακατατάξει, με d. Το rankning του εγγράφου με διάνυσμα d (πριν πάρουμε feedback από τον χρήστη), στο vector space model, θα δίνεται από το $\cos(d,q) = (d*q)/|d|*|q| = r$.

Όταν ο χρήστης κάνει την ερώτηση q και επανακατατάξει το έγγραφο με διάνυσμα d θα το τοποθετήσει ανάμεσα στα έγγραφα με διανύσματα a και b. Τα έγγραφα αυτά θα έχουν ranking $\cos(a,q)$ και $\cos(b,q)$ αντίστοιχα. Άρα αυτό που θέλουμε είναι να τροποποιήσουμε το d σε d', ώστε $\cos(a,q) \geq \cos(d',q) \geq \cos(b,q)$. Ας θεωρήσουμε r' το καινούριο ranking, δηλαδή $\cos(d',q) = r'$. Παρόλο που περιορίσαμε το διάστημα στο οποίο κινείται το r', εντούτοις δεν το καθορίσαμε ακόμα. Σημειώστε πως η επιλογή μας αυτή, παρόλο που δεν θα επηρεάσει τη μαθηματική διατύπωση του προβλήματος, εντούτοις θα επηρεάσει την δυνατότητα εύρεσης αλγόριθμου που να το επιλύει (έστω προσεγγιστικά). Για να φανεί αυτό θα κάνουμε μια επιλογή για το r' και θα σχολιάσουμε παρακάτω τις συνέπειες της επιλογής μας.

Θα μπορούσαμε να επιλέξουμε $r' = (\cos(a,q) + \cos(b,q))/2$ για να πάρουμε την ενδιάμεση τιμή ανάμεσα στα $\cos(a,q)$ και $\cos(b,q)$. Μια άλλη προσέγγιση θα ήταν η εξής: αν το έγγραφο

μετακινήθηκε προς τα πάνω, δηλαδή ήταν πιο κοντά στο έγγραφο με διάνυσμα b τότε διαλέγουμε ένα r' που θα είναι πιο κοντά στο b π.χ. η τιμή του b συν το 1/10 της απόστασης από το a έως το b, δηλαδή r '= cos(b,q) + (cos(a,q) - cos(b,q))/10 . Αντί για εν δέκατο μπορούμε να πάρουμε οτιδήποτε θεωρήσουμε καταλληλότερο (ένα εικοστό, ή εκατοστό), εξαρτάται από την πολιτική που θα ακολουθήσουμε. Παρόμοια αν το έγγραφο ήταν πάνω από το έγγραφο με διάνυσμα a και μεταφέρθηκε ανάμεσα σε a και b μπορούμε να πάρουμε r '= cos(a,q) - (cos(a,q) - cos(b,q))/10.

Υπάρχουν και οι τετριμμένες περιπτώσεις όταν ένα έγγραφο δεν επανακατατάσεται μεταξύ δυο
άλλων, αλλά το βάζουμε στην αρχή ή το τέλος της κατάταξης δηλαδή πάνω από το έγγραφο με το καλύτερο ranking ή τελευταίο, κάτω από το έγγραφο με το χειρότερο ranking. Στην περίπτωση όπου κατατάσσουμε το document πρώτο (πάνω από το έγγραφο με διάνυσμα a) μπορούμε να επιλέξουμε r '= cos(a,q) +x με $0 \leq x \leq 1 - \cos(a,q)$. Όταν x=0 δίνουμε το ranking του a στο d', ενώ όταν x = 1- cosSim(a,q) , τότε r' =1 (κάνουμε το έγγραφο d' 100% συναφές για αυτήν την επερώτηση). Αντίστοιχο r' (0<=r'<=cosSim(b,q)) μπορούμε να δώσουμε και όταν το έγγραφο τοποθετείται τελευταίο, κάτω από ένα έγγραφο με διάνυσμα b. Εν πάσει περιπτώσει, επιλέγουμε ένα r' ανάλογα με την πολιτική μας και στην περαιτέρω ανάλυση, θεωρούμε το r' γνωστό.

Άρα έχουμε να λύσουμε την εξίσωση cos(d',q) = (d*q)/(|d|*|q|) = r' (1) και να βρούμε ένα διάνυσμα d' που την ικανοποιεί. Η εξίσωση αυτή μπορεί να έχει πολλές λύσεις, πώς όμως θα διαλέξουμε το κατάλληλο διάνυσμα; Πρέπει να πάρουμε το διάνυσμα εκείνο που έχει τις λιγότερες αλλαγές. Έχουμε διάφορες πιθανές συναρτήσεις που μπορούν να μοντελοποιήσουν αυτήν την τροποποίηση του d σε d'. Μια υποψήφια είναι το μέτρο της διαφοράς των διανυσμάτων d-d'. Αυτό που ζητάμε συνεπώς, είναι να ελαχιστοποιήσουμε την f(d') = |d-d'| (2) υπό την συνθήκη (1). Μια άλλη προσέγγιση θα πρότεινε να μεγιστοποιήσουμε την cos(d,d') (3), αφού το ότι θέλουμε τα διανύσματα να διαφέρουν λίγο, σημαίνει ότι θέλουμε να μοιάζουν πολύ.

Συνεπώς το πρόβλημα ανάγεται στην διαδικασία εύρεσης ακρότατων μιας συνάρτησης (της (2) ή της (3)) υπό συνθήκη (την σχέση (1)). Η διαδικασία αυτή είναι γνωστή από τον διανυσματικό λογισμό και περιγράφεται συνοπτικά παρακάτω.

Όταν θέλουμε να βρούμε τα ακρότατα μιας συνάρτησης υπό συνθήκη μια πολύ συνηθισμένη μέθοδος είναι οι πολλαπλασιαστές Lagrange.

**Θεώρημα Lagrange:**

Έστω $f: U \subset R^n \Rightarrow R$ και $g: U \subset R^n \Rightarrow R$ δοσμένες λείες συναρτήσεις. Έστω διάνυσμα $x_0 \in U$ και $g(x_0) = c$ , και S το σύνολο στάθμης της g με τιμή c (υπενθυμίζουμε ότι αυτό είναι το σύνολο των σημείων $x \in R^n$ για τα οποία $g(x) = c$ ). Υποθέτουμε ότι $\nabla g(x_0) \neq 0$ .

Αν ο περιορισμός της f στο S, έχει μέγιστο ή ελάχιστο στο S στο σημείο $x_0$, τότε υπάρχει πραγματικός αριθμός λ τέτοιος ώστε:

$$\nabla f(x_0) = \lambda \nabla g(x_0)$$

Το θεώρημα Lagrange αυτό που μας λέει πρακτικά είναι ότι οι μερικές παράγωγοι της f είναι ανάλογες αυτών της g. Για να βρούμε τα σημεία που συμβαίνει κάτι τέτοιο πρέπει να λύσουμε το σύστημα των εξισώσεων:

$$\frac{\partial f}{\partial x_1}(x_{1,\dots},x_n)=\lambda\frac{\partial g}{\partial x_1}(x_{1,\dots},x_n)$$
$$\frac{\partial f}{\partial x_2}(x_{1,\dots},x_n)=\lambda\frac{\partial g}{\partial x_2}(x_{1,\dots},x_n)$$
$$\vdots$$
$$\frac{\partial f}{\partial x_n}(x_{1,\dots},x_n)=\lambda\frac{\partial g}{\partial x_n}(x_{1,\dots},x_n)$$
$$g(x_{1,\dots},x_n)=c$$

Επιστρέφοντας στο πρόβλημα μας η f για μας είναι η (2) ή η (3) και αντίστοιχα κρατάμε τα ελάχιστα σημεία που θα βρούμε ή τα μέγιστα υπό την g που για μας είναι η (1). Δεδομένου ότι η μέθοδος αυτή μας δίνει τοπικά ελάχιστα (μέγιστα) θα πρέπει να συγκρίνουμε τα αποτελέσματά του για να βρούμε το ολικό ελάχιστο αν χρησιμοποιούμε την (2) ή το ολικό μέγιστο αν χρησιμοποιούμε την (3).

Σ' αυτό το σημείο έχουμε τελειώσει με τη μοντελοποίηση του προβλήματος, *παρόλα αυτά δεν έχουμε βρει ακόμα ένα αλγόριθμο που να μας το λύνει*. Για να γίνει αυτό σαφές, σημειώστε ότι δε σχολιάσαμε καθόλου τι σημαίνει «λύνουμε το σύστημα των εξισώσεων» του θεωρήματος του Lagrange. Ας διατυπώσουμε το πρόβλημα λίγο διαφορετικά (έχοντας πάντα σαν αφετηρία το παραπάνω θεώρημα).

Βρες το x το οποίο:

ελαχιστοποιεί την $f(x)$

υπό τη συνθήκες: $g_i(x) \geq 0 \quad i = 1, \dots, m$
$h_j(x) = 0 \quad j = 1, \dots, p$

Η παραπάνω διατύπωση αποτελεί τη γενική μορφή του Προβλήματος του Μη-Γραμμικού-Προγραμματισμού (ΠΜΓΠ). Το πρόβλημα αυτό είναι επιλύσημο (υπάρχει αλγόριθμος επίλυσής του) μόνο αν η f είναι *κυρτή* (*convex*), η $g_i$ είναι *κοίλη* (*concave*) για κάθε i και η $h_j$ είναι *γραμμική* (*linear*) για κάθε j. Στην περίπτωση αυτή το πρόβλημα ονομάζεται Πρόβλημα Κυρτού Προγραμματισμού (ΠΚΠ) και υπάρχουν ικανές (sufficient) συνθήκες για τη βελτιστοποίηση, οι λεγόμενες συνθήκες Kuhn-Tucker. Οι συνθήκες αυτές απορρέουν από το θεώρημα του Lagrange συν του γεγονότος ότι το ΠΚΠ έχει τη χρήσιμη ιδιότητα ότι *η τοπική*

*βελτιστοποίηση συνεπάγεται ολική βελτιστοποίηση* (η μέθοδος του Lagrange μας δίνει τοπικά ακρότατα).

Στην περίπτωση μας δεν έχουμε καμία συνθήκη τύπου g. Έχουμε μία συνθήκη τύπου h (που απορρέει από την (1)) την $\cos(d', q) - r' = 0$. Είναι προφανές ότι η συνάρτηση αυτή δεν είναι γραμμική. Σημειώστε ότι η επιλογή του r' ήταν αυθαίρετη. Θα μπορούσαμε να αντικαταστήσουμε τη συνθήκη (1) με την $\cos(\alpha, q) \geq \cos(d', q) \geq \cos(\beta, q)$, η οποία αναλύεται σε δύο ανισότητες:

$g_1(d') = -\cos(d', q) + \cos(\alpha, q) \geq 0$
$g_2(d') = \cos(d', q) - \cos(\beta, q) \geq 0$

Και με αυτόν τον τρόπο αποτυγχάνουμε να ανάγουμε το πρόβλημα σε ένα ΠΚΠ γιατί όπως εύκολα διαπιστώνουμε οι $g_1$, $g_2$ δεν είναι κοίλες συναρτήσεις. Πληροφοριακά σημειώνουμε πως είτε επιλέξουμε σαν f την $f_1(d') = |d - d'|$ (2) είτε την $f_2(d') = \cos(d, d')$ (3), δεν καλύπτουμε ούτε τη κυρτότητα της f. Σε κάθε περίπτωση δηλαδή αποτυγχάνουμε (οικτρά) να ανάγουμε το πρόβλημα σε ένα ΠΚΠ.

Δεδομένης της παραπάνω αποτυχίας στρέφουμε το ενδιαφέρον μας στη διατύπωση και χρήση ενός heuristic, το οποίο θα οδηγεί σε εύρεση ενός αλγόριθμου που θα λύνει το πρόβλημα μόνο προσεγγιστικά, δηλαδή δε θα μας εγγυάται την ελάχιστη δυνατή αλλαγή του διανύσματος. Παρακάτω λοιπόν εγκαταλείπουμε τις προσεγγίσεις που βασίζονται στους πολλαπλασιαστές του Lagrange.

**<u>Ορολογία</u>**

Θα συμβολίζουμε με

- S την πηγή των δεδομένων, στην περίπτωσή μας το σύνολο των διανυσμάτων των εγγράφων της συλλογής μας
- q την επερώτηση
- S(q) η διατεταγμένη λίστα των απαντήσεων του ΣΑΠ στην επερώτηση q
- B δοσμένη διατεταγμένη λίστα εγγράφων του ΣΑΠ

Ας υποθέσουμε ότι αρχικά το ΣΑΠ επέστρεψε τη λίστα S(q) = B και ότι ο χρήστης δημιούργησε με την αλλαγή των θέσεων των εγγράφων τη λίστα B'. Προφανώς {B} ={B'}, δηλαδή το σύνολο (αγνοούμε τη διάταξη) των εγγράφων του B και του B' είναι ίδια. Ο στόχος μας είναι να αλλάξουμε τα διανύσματα των εγγράφων, δηλαδή να βρούμε ένα S', τέτοιο ώστε S'(q) = B'. Θέλουμε επίσης η αλλαγή του S' από το S να είναι όσο το δυνατόν μικρότερη. Το heuristic που θα χρησιμοποιήσουμε είναι ότι ζητάμε οι βαθμοί ομοιότητας των εγγράφων του B' με το q να είναι οι ίδιοι με τους αντίστοιχους του B (με τη διαφορά ότι κάποιοι από αυτούς θα αντιστοιχούν σε διαφορετικά έγγραφα). Με αυτό τον τρόπο κρατάμε το άθροισμα των βαθμών ομοιότητας των εγγράφων σταθερό. Το heuristic αυτό λύνει το πρόβλημά μας με σχετικά μικρή (αλλά χωρίς να εγγυάται την ελάχιστη δυνατή) αλλαγή στο S.

Το πρόβλημά μας μετά την παραπάνω υπόθεση ανάγεται στο γνωστό "String-to-string correction" πρόβλημα. Συνοπτικά ο αλγόριθμος αυτός μετατρέπει ένα δοσμένο String B σε ένα άλλο επίσης δοσμένο B' με το μικρότερο κόστος. Επιτρέπονται 3 λειτουργίες τροποποίησης:

- άλλαξε έναν χαρακτήρα σε έναν άλλο (α → β)
- πρόσθεσε ένα χαρακτήρα (κενό → α)
- σβήσε ένα χαρακτήρα (α → κενό)

Σε κάθε λειτουργία τροποποίησης δίνεται μια συνάρτηση κόστους γ(x→y) η οποία συσχετίζει ένα μη-αρνητικό πραγματικό αριθμό (κόστος) σε κάθε μετασχηματισμό x→y.

Στη δική μας περίπτωση τα ids των εγγράφων παίζουν το ρόλο των χαρακτήρων. Επίσης, αφού {B} ={B'}, δε χρειαζόμαστε τις λειτουργίες 'πρόσθεσε' και 'σβήσε' και επομένως ορίζουμε γ(x→y) = +∞ (ιδεατά) αν το x ή το y είναι το κενό. Για κάθε άλλο ζεύγος x, y ορίζουμε γ(x→y) = 1 – cos (x, y), δηλαδή την απόστασή των διανυσμάτων τους σύμφωνα με το μέτρο του συνημητόνου. Ο αλγόριθμος μας επιστρέφει την ακολουθία λειτουργία τροποποίησης που οδηγούν από το B στο B' με το μικρότερο συνολικό (αθροιστικά) κόστος σε χρόνο O(|B|*|B'|) = O(|B|^2).

Τώρα για κάθε μετασχηματισμό α → β χρειαζόμαστε μια μέθοδο modifyVec(β, q, c) που να μετατρέπει το διάνυσμα του β στο m(β) έτσι ώστε cos(m(β), q) = cos (a,q) = c.

Για να πετύχουμε αυτή την τροποποίηση με την ελάχιστη δυνατή αλλαγή, θα προσθέσουμε στο διάνυσμα β, το διάνυσμα κατεύθυνσης q-β και ελάχιστου δυνατού μέτρου, ώστε να πετύχουμε cos(m(β), q) = c. Σημειώστε ότι αυτή η προσέγγιση επιτρέπει τον εμπλουτισμό του διανύσματος του β με όρους που δεν περιέχονται στο β αλλά περιέχονται στο q. Π.χ. αν c=1 η μέθοδος επιστρέφει το q.
Συνοψίζοντας:
m(β) = β + (1/x)*(q-a)

Για να βρούμε την τιμή του x λύνουμε την:
cos(m(β), q) = c

η οποία μετά από πράξεις καταλήγει να είναι δευτεροβάθμια εξίσωση ως προς x. Θυμίζουμε ότι υπάρχει αλγόριθμος επίλυσης $2^o$ βάθμιων εξισώσεων (η μέθοδος της διακρίνουσας). Από αυτή την εξίσωση θα προκύψουν 2 λύσεις, οι οποίες γεωμετρικά αντιστοιχούν σε διάνυσμα γωνίας θ και –θ αντίστοιχα από το διάνυσμα του q. Λαμβάνοντας υπ' όψιν την αρχή της ελάχιστης αλλαγής θεωρούμε ως λύση εκείνη με τη μικρότερη απόλυτη τιμή, δεδομένου ότι το x εμφανίζεται στον παρονομαστή και άρα το m(β) – β ελαχιστοποιείται.