Πανεπιστήμιο Ιωαννίνων
Σχολή Θετικών Επιστημών
Τμήμα Πληροφορικής

ΜΑΘΗΜΑ
Ανάκτηση Πληροφορίας

ΘΕΜΑ
Μία αξιωματική προσέγγιση για τη διαφοροποίηση των αποτελεσμάτων
(Sreenivas Gollapudi and Aneesh Sharma, *WWW 2009***)**

Παππάς Χρήστος ΑΜ 204

Ιωάννινα, Ιανουάριος 2010

ΠΕΡΙΕΧΟΜΕΝΑ

Κεφάλαιο 1: Εισαγωγή

1.1 Πρόβλημα

Ένα σύστημα αναζήτησης εγγράφων, βάσει επερωτήσεων, είναι αποτελεσματικό όταν ικανοποιεί τις ανάγκες του χρήστη. Όταν δεν υπάρχει σαφής καθορισμός των επιδιώξεων του χρήστη, τότε το σύστημα θα πρέπει να διαφοροποιήσει τα αποτελέσματα. Αυτό σημαίνει ότι τα αποτελέσματα θέλουμε να αντιστοιχούν ιδανικά σε κάθε πιθανό επιδιωκόμενο σκοπό του χρήστη. Επομένως, το σύστημα δεν αρκεί να επιστρέφει τα πιο «σχετικά» έγγραφα, αλλά να συνυπολογίζει και την ποικιλομορφία τους.

1.2 Σημαντικότητα προβλήματος

Η σημαντικότητα της διαφοροποίησης έγκειται στο ότι ο χρήστης, πλέον, είναι πολύ πιο πιθανό να ικανοποιηθεί. Αυτό συμβαίνει καθώς, όταν ρωτά κάτι ασαφές, του επιστρέφονται αποτελέσματα που καλύπτουν διαφορετικές πιθανές προθέσεις του. Δηλαδή, το σύστημα δεν κάνει μία «τυχαία» πρόβλεψη(που ίσως καλύπτει μόνο μία πρόθεση, βάσει του πιθανοτικού μοντέλου σχετικότητας), αλλά προσπαθεί να διαφοροποιήσει τα έγγραφα, ώστε να μπορεί να ικανοποιηθεί μία οποιαδήποτε πιθανή πρόθεση του χρήστη.

1.3 Ενδιαφέροντα θέματα προβλήματος

Τα συστήματα διαφοροποίησης αποτελεσμάτων πρέπει να αποφασίσουν πώς να ισοσταθμίσουν τις έννοιες της «σχετικότητας» και της «ποικιλομορφίας». Αυτό χαρακτηρίζεται συχνά ως πρόβλημα βελτιστοποίησης δύο-κριτηρίων, καθώς θέλουμε «σχετικά» αλλά, ταυτόχρονα, «πρωτότυπα» αποτελέσματα(συνδυασμός ranking-clustering). Η ανάγκη αυτή οδήγησε στην ανάπτυξη πολλών και ενδιαφερόντων συναρτήσεων διαφοροποίησης καθώς και αντίστοιχων αλγορίθμων βελτιστοποίησής τους.

Κεφάλαιο 2: Τεχνικό περιεχόμενο

2.1 Ορισμοί εννοιών

Έστω σύνολο εγγράφων $U = \{u_1, u_2, \ldots, u_n\}$ n>=2 και σύνολο ερωτήσεων Q. Δοθέντος ερώτησης $q \in Q$ και ακεραίου k, θέλουμε ένα υποσύνολο $S_k \subseteq U$ που να είναι και «σχετικό» και «πρωτότυπο». Η σχετικότητα εγγράφου ορίζεται από την $w : U \times Q \to \mathbf{R}^+$ και η διαφοροποίηση έμμεσα με την απόσταση εγγράφων $d : U \times U \to \mathbf{R}^+$ (καλώς διαφοροποιημένα αποτελέσματα σημαίνει μεγάλη απόσταση μεταξύ τους).

Σκοπός είναι να βρούμε το καλύτερο υποσύνολο μεγέθους k(έπειτα, το ταξινομούμε χρησιμοποιώντας τη σχετικότητα). Ορίζουμε τη συνάρτηση επιλογής υποσυνόλου $f : 2^U \times Q \times w \times d \to \mathbf{R}$. Ο στόχος είναι, δοθέντος q, $w(\cdot)$, $d(\cdot,\cdot)$ και ακεραίου k>=2, να βρούμε το υποσύνολο $S_k \subseteq U$ που μεγιστοποιείται την f. Δηλαδή, το $S_k^* = \underset{S_k \subseteq U}{\operatorname{argmax}} f(S_k, q, w(\cdot), d(\cdot,\cdot))$ με $|S_k| = k$.

2.2 Μοντελοποίηση προβλήματος

Ορίζεται ένα σύνολο αξιωμάτων, τα οποία αναμένουμε να ικανοποιούνται από ένα σύστημα διαφοροποίησης. Έπειτα, ορίζουμε συναρτήσεις διαφοροποίησης, καθεμία από τις οποίες ικανοποιεί διαφορετικό υποσύνολο των ορισθέντων αξιωμάτων. Επομένως, τα αξιώματα αποτελούν μία «βάση» για σύγκριση μεταξύ διαφορετικών συναρτήσεων που ορίζουμε.

Αποδεικνύεται ότι δεν υπάρχει συνάρτηση που ικανοποιεί όλα τα αξιώματα που ορίζονται παρακάτω. Επομένως, κάθε σύστημα επιλέγει συνάρτηση διαφοροποίησης που ικανοποιεί ιδιότητες-αξιώματα που τα θεωρεί απαραίτητα.

Διατυπώνουμε τα εξής αξιώματα:

- **scale invariance**: Η f δεν επηρεάζεται από κλιμάκωση(κατά σταθερά α>0) στη σχετικότητα και απόσταση. Δηλαδή, η f μεγιστοποιείται για το ίδιο ακριβώς σύνολο, άρα: $S_k^* = \text{argmax}_{S_k \subseteq U} f(S_k, q, \alpha \cdot w(\cdot), \alpha \cdot d(\cdot,\cdot))$
- **consistency:** Δοθέντων συναρτήσεων $\alpha : U \to \mathbf{R}^+$ and $\beta : U \times U \to \mathbf{R}^+$, αλλάζουμε τις w και d:

  $$w(u) = \begin{cases} w(u) + \alpha(u) & , u \in S_k^* \\ w(u) - \alpha(u) & , \text{otherwise} \end{cases}$$

  $$d(u,v) = \begin{cases} d(u,v) + \beta(u,v) & , u,v \in S_k^* \\ d(u,v) - \beta(u,v) & , \text{otherwise} \end{cases}$$

  Η f πρέπει να μεγιστοποιείται πάλι από το ίδιο σύνολο. Δηλαδή, να μην επηρεάζεται αν αυξήσουμε σχετικότητα και απόσταση αποτελεσμάτων που παίρνουμε και μειώσουμε σχετικότητα και απόσταση υπόλοιπων εγγράφων.
- **richness**: Πρέπει να υπάρχουν συναρτήσεις $w(\cdot)$ και $d(\cdot,\cdot)$ ώστε, για οποιοδήποτε k>=2, να υπάρχει μοναδικό S_k^* που μεγιστοποιεί την f.
- **stability**: Η f πρέπει να οριστεί έτσι ώστε $S_k^* \subset S_{k+1}^*$.
- **independence of irrelevant attributes**: Δοθέντος συνόλου S, θέλουμε η f να είναι τέτοια ώστε το f(S) να είναι ανεξάρτητο από:
  $w(u)$ for all $u \notin S$ και $d(u,v)$ for all $u,v \notin S$
- **monotonicity**: Δοθέντων $w(\cdot)$, $d(\cdot,\cdot)$, f and $S \subseteq U$, έχουμε ότι $f(S \cup \{x\}) \geq f(S)$ για κάθε $x \notin S$.
- **strength of relevance**: Η f δεν αγνοεί τη συνάρτηση σχετικότητας.
  Δοθέντων $w(\cdot)$, $d(\cdot,\cdot)$, f and S, πρέπει για κάθε $x \in S$:

  a)Υπάρχουν $\delta_0 > 0$ and $a_0 > 0$ ώστε:

  $f(S, w'(\cdot), d(\cdot,\cdot), k) = f(S, w(\cdot), d(\cdot,\cdot), k) + \delta_0$

  Η w' προέκυψε από w με την αλλαγή $w'(x) = a_0 > w(x)$

  b)Αν $f(S \setminus \{x\}) < f(S)$, υπάρχουν $\delta_1 > 0$ and $a_1 > 0$ ώστε:

  $f(S, w'(\cdot), d(\cdot,\cdot), k) = f(S, w(\cdot), d(\cdot,\cdot), k) - \delta_1$

  Η w' προέκυψε από w με την αλλαγή $w'(x) = a_1 < w(x)$

- **strength of similarity**: Η f δεν αγνοεί τη συνάρτηση ομοιότητας(απόσταση). Δοθέντων $w(\cdot)$, $d(\cdot,\cdot)$, f and S, πρέπει για κάθε $x \in S$:

  a)Υπάρχουν $\delta_0 > 0$ and $b_0 > 0$ ώστε:

  $f(S, w(\cdot), d'(\cdot,\cdot), k) = f(S, w(\cdot), d(\cdot,\cdot), k) + \delta_0$

  Η d' προέκυψε από d με την αλλαγή ότι αυξάνουμε τα $d(x,u)$ ώστε $\min_{u \in S} d(x,u) = b_0$

  b)Αν $f(S \setminus \{x\}) < f(S)$, υπάρχουν $\delta_1 > 0$ and $b_1 > 0$ ώστε:

  $f(S, w(\cdot), d'(\cdot,\cdot), k) = f(S, w(\cdot), d(\cdot,\cdot), k) - \delta_1$

  Η d' προέκυψε από d με την αλλαγή ότι μειώνουμε τα $d(x,u)$ ώστε $\max_{u \in S} d(x,u) = b_1$

2.3 Αλγόριθμοι

Αναφέρουμε 3 συναρτήσεις διαφοροποίησης, όπου καθεμία ικανοποιεί ένα διαφορετικό υποσύνολο των 8 αξιωμάτων. Επίσης, αναφέρουμε αλγορίθμους βελτιστοποίησης των συναρτήσεων. Στις 2 πρώτες περιπτώσεις, οι αλγόριθμοι προκύπτουν με μετασχηματισμούς στο γνωστό πρόβλημα συνδυαστικής βελτιστοποίησης «**facility dispersion**»(βλέπε Παράρτημα). Συγκεκριμένα:

a) **Max-sum diversification**

Θέλουμε να μεγιστοποιήσουμε το άθροισμα της σχετικότητας και απόστασης του επιλεγμένου υποσυνόλου.

$$f(S) = (k-1)\sum_{u \in S} w(u) + 2\lambda \sum_{u,v \in S} d(u,v)$$

Στην εξίσωση, $|S| = k$ και $\lambda > 0$ παράμετρος(όσο μεγαλύτερη είναι, τόσο περισσότερη σημασία δίνουμε στην απόσταση)

Χαρακτηρισμός: Η συνάρτηση ικανοποιεί όλα τα αξιώματα, εκτός από το «**stability**».

Ο αλγόριθμος βελτιστοποίησης προκύπτει από το MaxSumDispersion πρόβλημα του «facility dispersion». Αυτό το πρόβλημα βελτιστοποιεί την: $f(S) = \sum_{u,v \in S} d'(u,v)$. Το πρόβλημά μας ανάγεται σε αυτό, αν θεωρήσουμε:

$d'(u,v) = w(u) + w(v) + 2\lambda d(u,v)$

Επομένως, επιλύουμε το πρόβλημά μας με αναγωγή στο MaxSumDispersion. Υπάρχει ένας 2-προσεγγιστικός αλγόριθμος για το MaxSumDispersion, καθώς το συγκεκριμένο πρόβλημα είναι NP-hard. Ο αλγόριθμος περιγράφεται παρακάτω και ισχύει στην περίπτωση που η d' είναι «**metric**»(βλέπε Παράρτημα).

Input : Universe U, k
Output: Set S ($|S| = k$) that maximizes $f(S)$

Initialize the set $S = \emptyset$
for $i \leftarrow 1$ **to** $\lfloor \frac{k}{2} \rfloor$ **do**
  Find $(u,v) = \text{argmax}_{x,y \in U}\, d(x,y)$
  Set $S = S \cup \{u,v\}$
  Delete all edges from E that are incident to u or v
end
If k is odd, add an arbitrary document to S

Algorithm 1: Algorithm for MAXSUMDISPERSION

b) **Max-min diversification**

Θέλουμε να μεγιστοποιήσουμε την ελάχιστη σχετικότητα και απόσταση του επιλεγμένου υποσυνόλου.

$$f(S) = \min_{u \in S} w(u) + \lambda \min_{u,v \in S} d(u,v)$$

Στην εξίσωση, $|S| = k$ και $\lambda > 0$ παράμετρος(όσο μεγαλύτερη είναι, τόσο περισσότερη σημασία δίνουμε στην απόσταση)

Χαρακτηρισμός: Η συνάρτηση ικανοποιεί όλα τα αξιώματα, εκτός από το «**consistency**» και «**stability**».

Ο αλγόριθμος βελτιστοποίησης προκύπτει από το MaxMinDispersion πρόβλημα του «facility dispersion». Αυτό το πρόβλημα βελτιστοποιεί την: $g(P) = \min_{v_i, v_j \in P} d(v_i, v_j)$. Το πρόβλημά μας ανάγεται σε αυτό, αν θεωρήσουμε:

$d'(u,v) = \frac{1}{2}(w(u) + w(v)) + \lambda d(u,v)$. Τότε έχουμε:

$$\min_{u,v \in S} d'(u,v) = \min_{u \in S} w(u) + \lambda \min_{u,v \in S} d(u,v) = f(S)$$

Επομένως, επιλύουμε το πρόβλημά μας με αναγωγή στο MaxMinDispersion. Υπάρχει ένας 2-προσεγγιστικός αλγόριθμος για το MaxMinDispersion, καθώς το συγκεκριμένο πρόβλημα είναι NP-hard. Ο αλγόριθμος περιγράφεται παρακάτω και ισχύει στην περίπτωση που η d' είναι «**metric**»(βλέπε Παράρτημα).

Input : Universe U, k
Output: Set S ($|S| = k$) that maximizes $f(S)$

Initialize the set $S = \emptyset$; Find $(u, v) = \text{argmax}_{x,y \in U} d(x, y)$ and set $S = \{u, v\}$; For any $x \in U \setminus S$, define $d(x, S) = \min_{u \in S} d(x, u)$;
while $|S| < k$ **do**
  Find $x \in U \setminus S$ such that $x = \text{argmax}_{x \in U \setminus S} d(x, S)$;
  Set $S = S \cup \{x\}$;
end

Algorithm 2: Algorithm for MAXMINDISPERSION

c) **Mono-objective formulation**

Θέλουμε να μεγιστοποιήσουμε την:

$f(S) = \sum_{u \in S} w'(u)$, όπου $w'(u) = w(u) + \frac{\lambda}{|U| - 1} \sum_{v \in U} d(u, v)$ και $\lambda > 0$ παράμετρος, όπως και πριν.

Αυτή η συνάρτηση διαφέρει από τις προηγούμενες, καθώς ορίζει μία τιμή για κάθε έγγραφο, η οποία(τιμή) συνδυάζει τις τιμές σχετικότητας και απόστασης. Παρατηρούμε ότι η w' εκφράζει τη «συνολική» σημασία ενός εγγράφου στη συλλογή.

Χαρακτηρισμός: Η συνάρτηση ικανοποιεί όλα τα αξιώματα, εκτός από το «**consistency**».

Ο αλγόριθμος βελτιστοποίησης προκύπτει ως εξής: Υπολογίζουμε την τιμή $w'(u)$ για όλα τα $u \in U$. Τότε έχουμε επακριβώς τη βέλτιστη λύση, επιλέγοντας τα k έγγραφα της συλλογής με τις μεγαλύτερες τιμές $w'(u)$(γιατί έτσι μεγιστοποιείται το άθροισμα $f(S) = \sum_{u \in S} w'(u)$).

d) **Other objective functions**

Υπάρχουν πολλές άλλα προβλήματα-συναρτήσεις που ανάγονται σε κάποιο από τα προβλήματα του «facility dispersion» που περιγράφηκαν. Για παράδειγμα, το πρόβλημα **MaxMSTDispersion** μεγιστοποιεί το βάρος του ελάχιστου σκελετικού δέντρου του επιλεγμένου υποσυνόλου. Αποδεικνύεται ότι ο δεύτερος αλγόριθμος είναι ο καλύτερος με παράγοντα προσέγγισης, όμως, 4.

Επίσης, οι συναρτήσεις για το **DIVERSIFY** πρόβλημα και το **MinQueryAbandonment** πρόβλημα παραβιάζουν τα αξιώματα «**stability**» και «**independence of irrelevant attributes**».

2.4 Ειδικές περιπτώσεις συναρτήσεων απόστασης

Αναφέρουμε 2 περιπτώσεις συναρτήσεων απόστασης, οι οποίες θα χρησιμοποιηθούν στα πειράματα. Για τις συναρτήσεις σχετικότητας θα γίνει αναφορά στα πειράματα. Δίνεται έμφαση στην απόσταση, καθώς εξαρτάται από το είδος των δεδομένων που χρησιμοποιούνται.

a) **Semantic distance**

Στην περίπτωση των web σελίδων, υπολογίζουμε την απόστασή τους ως: $d(u,v) = 1 - sim(u,v)$, όπου u,v σελίδες. Το sim(u,v) υπολογίζεται χρησιμοποιώντας την Jaccard ομοιότητα. Δηλαδή: $sim(u,v) = \frac{|S(u) \cap S(v)|}{|S(u) \cup S(v)|}$.

Για το S, ορίζουμε τα εξής: Έστω συνάρτηση hash(h) που απεικονίζει τιμές από το U ομοιόμορφα στο [0,1]. Τότε το min-hash του συνόλου A=U ορίζεται: $MH_h(A) = \text{argmin}_x\{h(x)|x \in A\}$. Ο ορισμός αυτός επεκτείνεται εύκολα και για πολυσύνολα ως εξής: $MH(A) = \underset{x}{\text{argmin}}\{h(x,i)|x \in A, 1 \leq i \leq c_x\}$, όπου c_x είναι η συχνότητα ενός στοιχείου x του A. Επομένως, έχοντας k hash συναρτήσεις, το sketch(σκιαγράφηση περιεχομένου) του εγγράφου-σελίδας d είναι: $S(d) = \{MH_{h_1}(d), MH_{h_2}(d), \ldots, MH_{h_k}(d)\}$.

b) **Categorical distance**

Στην περίπτωση της ιεραρχίας προϊόντων, υπολογίζουμε την απόστασή τους, αφού τα έχουμε ταξινομήσει σε κατηγορίες. Μετά το τέλος της ταξινόμησης, έχει δημιουργηθεί ένα δέντρο κατηγοριών, όπου σε κάθε υποδέντρο βρίσκονται, αντίστοιχα, οι υποκατηγορίες.

Για τον υπολογισμό της απόστασης, επομένως, βρίσκουμε μία βεβαρημένη δεντρική απόσταση μεταξύ των κατηγοριών των προϊόντων. Η απόσταση αυτή ορίζεται ως: $d(u,v) = \sum_{i=1}^{l(u)} \frac{1}{2^{e(i-1)}} + \sum_{i=1}^{l(v)} \frac{1}{2^{e(i-1)}}$, όπου u,v κόμβοι-κατηγορίες, e>=0 και $l(\cdot)$ το βάθος του κόμβου-κατηγορίας στην ταξινομία-δέντρο. Αν e=0, τότε η απόσταση αυτή ανάγεται στο μήκος της διαδρομής μέχρι τον ελάχιστο κοινό προκάτοχο($lca(u,v)$).

Αν θεωρήσουμε ότι κάθε προϊόν μπορεί να ανήκει σε περισσότερες κατηγορίες(με διαφορετικές πιθανότητες), τότε ορίζουμε την απόστασή τους ως: $d_c(x,y) = \sum_{u \in C_x, v \in C_y} \min(C_x(u), C_y(v)) \operatorname{argmin}_v d(u,v)$, όπου C_x,C_y η κατηγορική πληροφορία(διάνυσμα με τιμές τις πιθανότητες το προϊόν να ανήκει στις αντίστοιχες κατηγορίες) για τα προϊόντα x,y, αντίστοιχα.

Κεφάλαιο 3: Πειραματική αποτίμηση

3.1 Μέτρα εκτίμησης

Θέλουμε να χαρακτηρίσουμε την επιλογή συνάρτησης διαφοροποίησης, χρησιμοποιώντας 2 μέτρα: σχετικότητα(relevance) και πρωτοτυπία(novelty). Προηγουμένως, χαρακτηρίσαμε συναρτήσεις διαφοροποίησης με βάση αξιώματα που ικανοποιούσαν.

Διενεργούμε 2 πειράματα. Στο πρώτο πείραμα, αποτιμούμε την επίδοση των 3 συναρτήσεων που ορίστηκαν στο προηγούμενο κεφάλαιο, έχοντας ως βάση το σύνολο των αμφίσημων σελίδων της wikipedia. Στο δεύτερο πείραμα, αποτιμούμε πάλι τις 3 συναρτήσεις, όσον αφορά την κατηγορική διαφοροποίηση προϊόντων.

3.2 Semantic disambiguation(πρώτο πείραμα)

Έστω Q οι αμφίσημες σελίδες της wikipedia και $\mathcal{S}_q$ τα θέματα(topics) που σχετίζονται με κάθε τέτοια σελίδα q. Ιδανικά θέλουμε τα αποτελέσματα, για μία ερώτηση q, να περιέχουν πολλά θέματα του $\mathcal{S}_q$. Χρησιμοποιώντας την **semantic distance**, υπολογίζουμε την πιθανότητα $p_q(x, s)$ ένα αποτέλεσμα-έγγραφο x να αναπαριστά ένα θέμα $s \in S_q$, δεδομένης ερώτησης q.

Υπενθυμίζεται ότι η διαδικασία διαφοροποίησης γίνεται ως εξής: Ανακτούνται από τη μηχανή αναζήτησης τα top n έγγραφα R(q) για μία ερώτηση $q \in \mathcal{Q}$. Μετά εφαρμόζουμε σε αυτά τον αλγόριθμο διαφοροποίησης και παίρνουμε τα top k διαφοροποιημένα έγγραφα D(q).

Συμβολίζοντας τα πρώτα k έγγραφα της R(q) ως $R_k(q)$, συγκρίνουμε τα $D(q)$ και $R_k(q)$ με τα μέτρα novelty και relevance.

3.2.1 Novelty

Θέλουμε να εκτιμήσουμε πόσα θέματα, ερώτησης q, καλύπτονται σε μία λίστα αποτελεσμάτων L. Υπολογίζουμε για την L το:

$\text{Novelty}_q(L) = \frac{1}{|S_q|} \sum_{s \in S_q} \mathbf{I}\left(\sum_{x \in L} p_q(x, s) > \theta\right)$, όπου θ κατώφλι ικανοποίησης και I συνάρτηση ίση με 1, αν η συνθήκη είναι αληθής, αλλιώς ίση με 0.

Για να συγκρίνουμε τις $D(q)$ και $R_k(q)$, υπολογίζουμε το:

$$\text{FN}_q = \frac{\text{Novelty}_q(D(q)) - \text{Novelty}_q(R_k(q))}{\max\left(\text{Novelty}_q(D(q)), \text{Novelty}_q(R_k(q))\right)}$$

3.2.2 Relevance

Έχοντας την κατάταξη της wikipedia ως ιδανική κατάταξη για τη σχετικότητα των αποτελεσμάτων, υπολογίζουμε τη σχετικότητα λίστας S ως:

$R(S, q) = \sum_{s \in S'} \left| \frac{1}{r_s} - \frac{1}{r'_s} \right|$, όπου S' η ιδανική wikipedia κατάταξη και r_s,r'_s η διάταξη του εγγράφου s στις S και S', αντίστοιχα.

Για την $R_k(q)$ βρίσκουμε το r_s κάνοντας αναζήτηση περιορισμένη στα wikipedia sites. Χρησιμοποιώντας τη θέση κάθε $s \in S_q$ στη διάταξη, βρίσκουμε το r_s.

Για την $D(q)$ βρίσκουμε το r_s υπολογίζοντας το: $Rel(s, q) = \sum_{d \in D(q)} \frac{1}{pos(d)} p_q(d, s)$, όπου pos(d) η διάταξη του εγγράφου στην D(q). Ταξινομούμε τα έγγραφα με βάση το Rel(s,q) και, χρησιμοποιώντας τη θέση κάθε εγγράφου στη διάταξη, παίρνουμε το r_s.

Για να συγκρίνουμε τις $D(q)$ και $R_k(q)$, υπολογίζουμε το:

$\text{FR}_q = \frac{\text{Relevance}_q(D(q)) - \text{Relevance}_q(R_k(q))}{\max\left(\text{Relevance}_q(D(q)), \text{Relevance}_q(R_k(q))\right)}$, όπου $\text{Relevance}_q(S)=R(S,q)$

3.2.3 Results

Παρακάτω δίνονται γραφικές παραστάσεις που προέκυψαν μετά την εκτέλεση διάφορων πειραμάτων. Οι παράμετροι των πειραμάτων είναι n=30 και k=10.

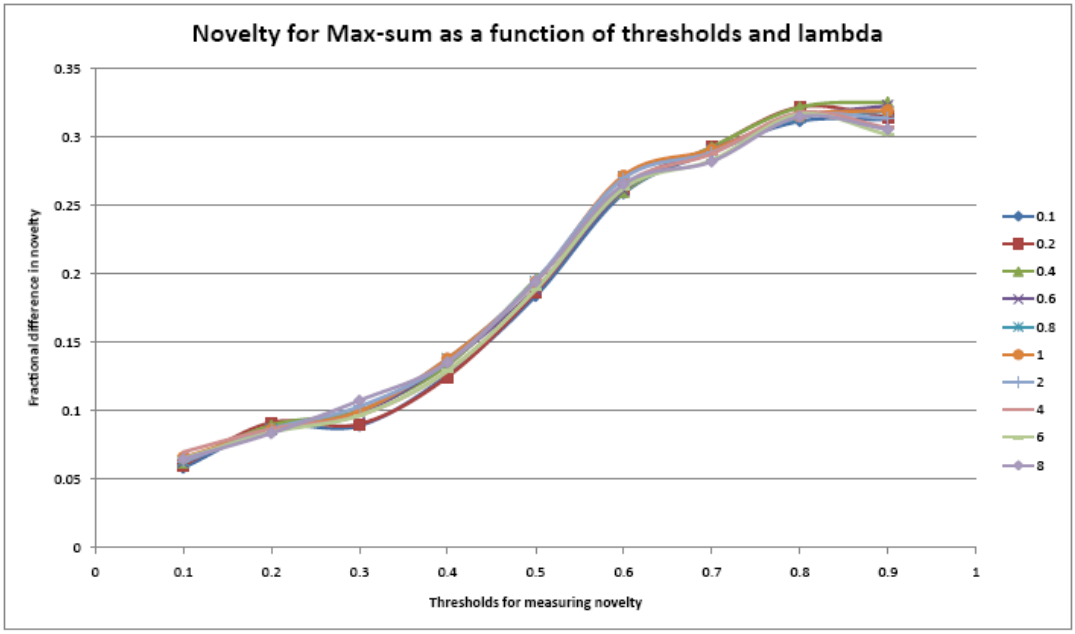
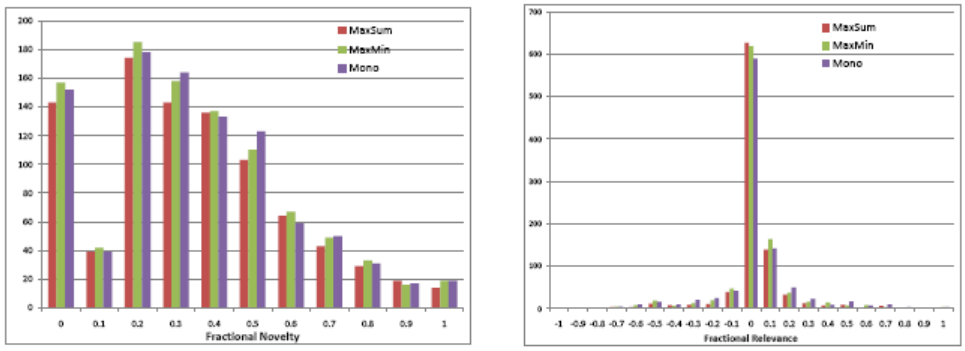
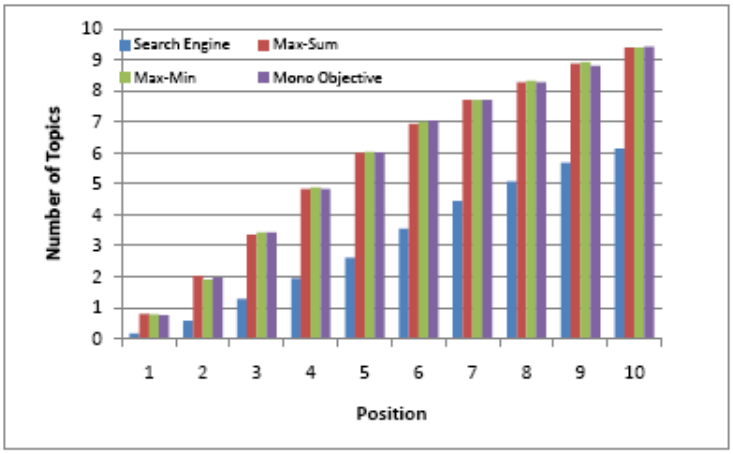


Figure 1: [Best viewed in color] The effect of varying the value of the trade-off parameter λ, and the threshold for measuring novelty on the output of the search results from MaxSumDispersion.

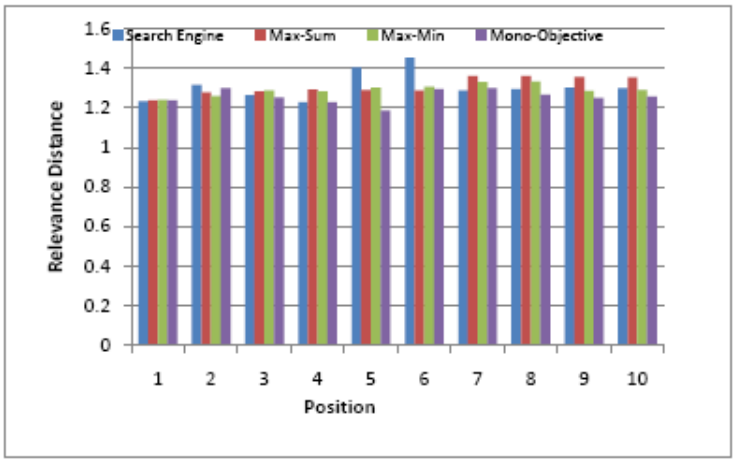


(a) Novelty (b) Relevance

Figure 2: [Best viewed in color] The histogram of fractional difference in novelty (a) and relevance (b) plotted over a 1000 ambiguous queries.



(a) Novelty



(b) Relevance

Figure 3: [Best viewed in color] The positional variation in the novelty (a) and relevance (b) of the result set.

Στο **figure 1** παρατηρούμε ότι FN_q είναι θετικό, που σημαίνει αύξηση της «πρωτοτυπίας». Αυξάνοντας θ ή λ, αυξάνεται το FN_q.

Στο **figure 2(a)** παρατηρούμε ότι ο MaxMinDispersion υπερτερεί στην «πρωτοτυπία». Γενικά, 75% των ερωτήσεων παράγουν διαφοροποιημένα αποτελέσματα σε σχέση με αναζήτηση. Στο **figure 2(b)** παρατηρούμε ότι το MonoObjective υπερτερεί στην «σχετικότητα», που είναι λογικό λόγω ορισμού του. Γενικά, έχουμε καλό ranking, αντίστοιχο με μηχανή αναζήτησης.

Στο **figure 3** χρησιμοποιήθηκε λ=1.0 και θ=0.5. Στο **figure 3(a)** παρατηρούμε ότι οι αλγόριθμοι υπερτερούν έναντι μηχανής αναζήτησης, ενώ μεταξύ τους ισάξιοι. Στο **figure 3(b)** παρατηρούμε ότι διαφορές υπάρχουν στις χαμηλές θέσεις. Βλέπουμε ότι ο MonoObjective υπερτερεί, καθώς απέχει λιγότερο από wikipedia σελίδες, ενώ ο MaxSumDispersion παράγει λιγότερο «σχετικά» αποτελέσματα, καθώς είναι πιο πιθανό να προσθέσει διαφοροποιημένο αποτέλεσμα.

3.3 Product disambiguation(δεύτερο πείραμα)

Το σύνολο δεδομένων αποτελείται από 100 ερωτήσεις προϊόντων και τα 50 πρώτα αποτελέσματα, βάσει της δημοτικότητας των προϊόντων. Η «σχετικότητα» βασίζεται στο πόσο διάσημο είναι ένα προϊόν και η «απόσταση» είναι η **categorical distance**.

Συμβολίζοντας τα πρώτα k έγγραφα της R(q) ως $R_k(q)$, συγκρίνουμε τα $D(q)$ και $R_k(q)$ με τα μέτρα novelty και relevance.

3.3.1 Novelty

Θέλουμε να εκτιμήσουμε πόσες κατηγορίες προϊόντων, ερώτησης q, καλύπτονται σε μία λίστα αποτελεσμάτων L. Μία κατηγορία αναπαρίσταται στην L, αν δεν είναι «απόγονος» άλλης κατηγορίας. Υπολογίζουμε για την L το:

$\text{Novelty}_q(L) = \frac{2}{|L|(|L|-1)} \sum_{u,v \in L} \mathbf{I}(lca(u,v) \notin \{u,v\})$, όπου lca ο ελάχιστος κοινός προκάτοχος και I συνάρτηση ίση με 1, αν η συνθήκη είναι αληθής, αλλιώς ίση με 0.

3.3.2 Relevance

Επειδή δεν έχουμε κάποια ιδανική κατάταξη για να συγκρίνουμε διατάξεις αποτελεσμάτων, θεωρούμε ένα προϊόν σχετικό με ερώτηση ανάλογα με το πόσο σχετικές είναι οι αντίστοιχες κατηγορίες τους στην ταξινομία. Θεωρούμε σχετικές 2 κατηγορίες, όταν η μία εμπεριέχει την άλλη. Υπολογίζουμε για την L το:

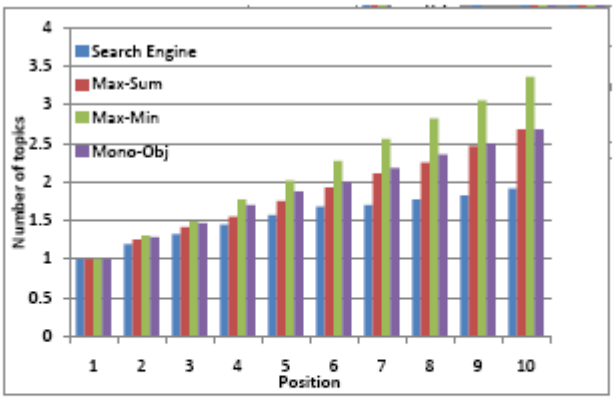
$\frac{1}{\text{Relevance}_q(L)} = \frac{1}{|L|} \sum_{u \in L} \frac{1 + d(lca(q,u),q)}{pos(u)}$, όπου $pos(\cdot)$ είναι η θέση του u στη διάταξη και $d(\cdot,\cdot)$ η categorical distance.

3.3.3 Results

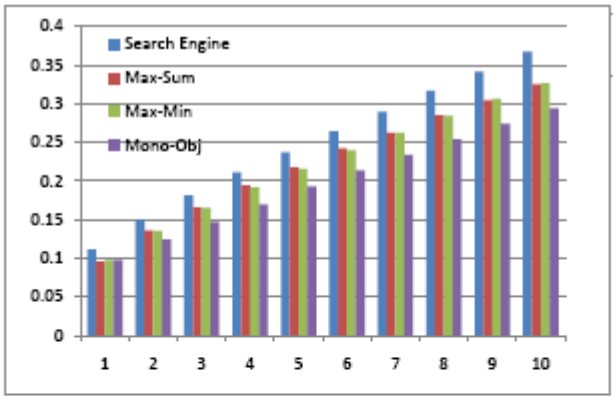
Παρακάτω δίνονται γραφικές παραστάσεις που προέκυψαν μετά την εκτέλεση διάφορων πειραμάτων.

Product Search Engine	Diversified Results
Sony SCD CE595-SACD changer	Sony SCD CE595-SACD changer
Sony CDP CE375-CD changer	Sony CDP CE375-CD changer
Sony CDP CX355-CD changer	Teac SR-L50-CD player/radio
Teac SR-L50- CD player/radio	Bose Wave Music System Multi-CD Changer
Bose Wave Music System Multi-CD Changer	Sony S2 Sports ATRAC3/MP3 CD Walkman D-NS505
Sony RCD-W500C-CD changer/CD recorder	COBY CX CD109-CD player
Sony CD Walkman D-EJ011-CD player	JVC XL PG3-CD player
Sony S2 Sports ATRAC3/MP3 CD Walkman D-NS505	Pioneer PD M426-CD changer
Sony Atrac3/MP3 CD Walkman D-NF430	Sony SCD XA9000ES-SACD player
COBY CX CD109-CD player	Yamaha CDR HD1500-CD recorder/HDD recorder

Table 1: The difference in the top 10 results for the query `cd player` from a commercial product search engine and the diversified results produced by running MAXSUMDISPERSION with $n = 30$ and $k = 10$



(a) Diversity



(b) Relevance

Figure 4: [Best viewed in color] The positional variation in the diversity (a) and relevance (b) of the product result set averaged over 100 queries with $n = 30$, $k = 10$, $\lambda = 1.0$ and $\theta = 0.5$.

Στο **table 1** παρατηρούμε ότι τα διαφοροποιημένα αποτελέσματα εμπεριέχουν περισσότερες μάρκες προϊόντων σε σχέση με αυτά της μηχανής αναζήτησης.

Στο **figure 4(a)** παρατηρούμε ότι ο MaxMinDispersion υπερτερεί έναντι των άλλων αλγορίθμων, ενώ και οι 3 υπερέχουν της μηχανής αναζήτησης. Στο **figure 4(b)** παρατηρούμε ότι ο MonoObjective παραδόξως μειονεκτεί, καθώς μειώνεται η τιμή συγκριτικά με τους άλλους αλγορίθμους.

Κεφάλαιο 4: Κριτική-επεκτάσεις εργασίας

4.1 Πλεονεκτήματα

- Ο τυχαίος χρήστης είναι πολύ πιο πιθανό να ικανοποιηθεί με τα αποτελέσματα, σε σχέση με την διάταξη που θα του επέστρεφε η μηχανή αναζήτησης.
- Η χρήση των αξιωμάτων επιτρέπει ένα θεωρητικό χαρακτηρισμό των συναρτήσεων διαφοροποίησης, ανεξάρτητα από τις συναρτήσεις «σχετικότητας» και «απόστασης» που χρησιμοποιούνται.
- Τα μέτρα πρωτοτυπίας και σχετικότητας, που χρησιμοποιούνται κατά την πειραματική ανάλυση, ποσοτικοποιούν κατάλληλα την απόδοση των συναρτήσεων.

4.2 Μειονεκτήματα

- Η πειραματική ανάλυση και αξιολόγηση εξαρτάται σημαντικά από τα αξιώματα, καθώς οι συναρτήσεις διαφοροποίησης ικανοποιούν διαφορετικά σύνολα αξιωμάτων.
- Οι λύσεις των 2 αλγορίθμων διαφοροποίησης είναι προσεγγιστικές, καθώς το γενικότερο πρόβλημα του «facility dispersion» είναι NP-hard.
- Δεν υπάρχει κάποια διάταξη των αξιωμάτων ως προς τη σημαντικότητά τους, αλλά μας ενδιαφέρει μόνο το ποιά ικανοποιούνται, όταν αξιολογούμε τις συναρτήσεις διαφοροποίησης.

4.3 Επεκτάσεις

- Έλεγχος ικανοποίησης και των 8 αξιωμάτων, στην περίπτωση που η απόσταση είναι «metric».
- «Χαλάρωση» κάποιων αξιωμάτων(π.χ. stability), προκειμένου να διευκολύνουμε την εύρεση κατάλληλων συναρτήσεων διαφοροποίησης.
- Εύρεση νέων συναρτήσεων διαφοροποίησης, που θα ανάγονται στο «facility dispersion» πρόβλημα και οι οποίες θα είναι βέλτιστες για συγκεκριμένες περιοχές ενδιαφέροντος(π.χ. web search).
- Εισαγωγή βαρών σημαντικότητας στα αξιώματα που ορίζονται. Έτσι γίνεται πληρέστερη η αξιολόγηση συναρτήσεων διαφοροποίησης, καθώς προτιμάται η ικανοποίηση αξιωμάτων με μεγαλύτερα βάρη(μεγάλο βάρος σημαίνει ότι το αξίωμα ικανοποιεί καλύτερα τους στόχους ενός συστήματος διαφοροποίησης).
- Αξιοποιώντας τα μέτρα novelty και relevance, εύρεση κατάλληλων μέτρων που προκύπτουν από το συνδυασμό τους. Τα νέα αυτά μέτρα θα έχουν στόχο να αξιολογούν καλύτερα την απόδοση των αλγορίθμων.

ΠΑΡΑΡΤΗΜΑ

1) Facility dispersion

Το πρόβλημα του «facility dispersion» ορίζεται ως η τοποθέτηση μονάδων σε ένα δίκτυο, έτσι ώστε να μεγιστοποιείται κάποια συνάρτηση απόστασης μεταξύ τους. Για παράδειγμα, το κριτήριο βελτιστοποίησης MAX-MIN μεγιστοποιεί την ελάχιστη απόσταση για οποιοδήποτε ζεύγος μονάδων και το κριτήριο βελτιστοποίησης MAX-SUM μεγιστοποιεί το άθροισμα των αποστάσεων για όλα τα ζεύγη μονάδων.

2) Metric distance

Μία απόσταση d είναι metric αν για οποιαδήποτε u,v,w συνόλου A:

- d(u,v)=0 αν και μόνο αν u=v
- d(u,v)= d(v,u)
- d(u,v) <= d(u,w) + d(w,v)