ΑΡΙΣΤΟΤΕΛΕΙΟ
ΠΑΝΕΠΙΣΤΗΜΙΟ
ΘΕΣΣΑΛΟΝΙΚΗΣ

ΑΝΟΙΚΤΑ
ΑΚΑΔΗΜΑΪΚΑ
ΜΑΘΗΜΑΤΑ

Ανάκτηση πληροφορίας

Ενότητα 6: Ο Αντεστραμμένος Κατάλογος

Απόστολος Παπαδόπουλος
Τμήμα Πληροφορικής

Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο
ΕΠΙΧΕΙΡΗΣΙΑΚΟ ΠΡΟΓΡΑΜΜΑ
ΕΚΠΑΙΔΕΥΣΗ ΚΑΙ ΔΙΑ ΒΙΟΥ ΜΑΘΗΣΗ
επένδυση στην κοινωνία της γνώσης
ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ ΚΑΙ ΘΡΗΣΚΕΥΜΑΤΩΝ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ
ΕΣΠΑ 2007-2013
πρόγραμμα για την ανάπτυξη
ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ
Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

Άδειες Χρήσης

- Το παρόν εκπαιδευτικό υλικό υπόκειται σε άδειες χρήσης Creative Commons.
- Για εκπαιδευτικό υλικό, όπως εικόνες, που υπόκειται σε άλλου τύπου άδειας χρήσης, η άδεια χρήσης αναφέρεται ρητώς.

Χρηματοδότηση

- Το παρόν εκπαιδευτικό υλικό έχει αναπτυχθεί στα πλαίσια του εκπαιδευτικού έργου του διδάσκοντα.
- Το έργο «Ανοικτά Ακαδημαϊκά Μαθήματα στο Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης» έχει χρηματοδοτήσει μόνο την αναδιαμόρφωση του εκπαιδευτικού υλικού.
- Το έργο υλοποιείται στο πλαίσιο του Επιχειρησιακού Προγράμματος «Εκπαίδευση και Δια Βίου Μάθηση» και συγχρηματοδοτείται από την Ευρωπαϊκή Ένωση (Ευρωπαϊκό Κοινωνικό Ταμείο) και από εθνικούς πόρους.

Ο Αντεστραμμένος Κατάλογος (Inverted Index)

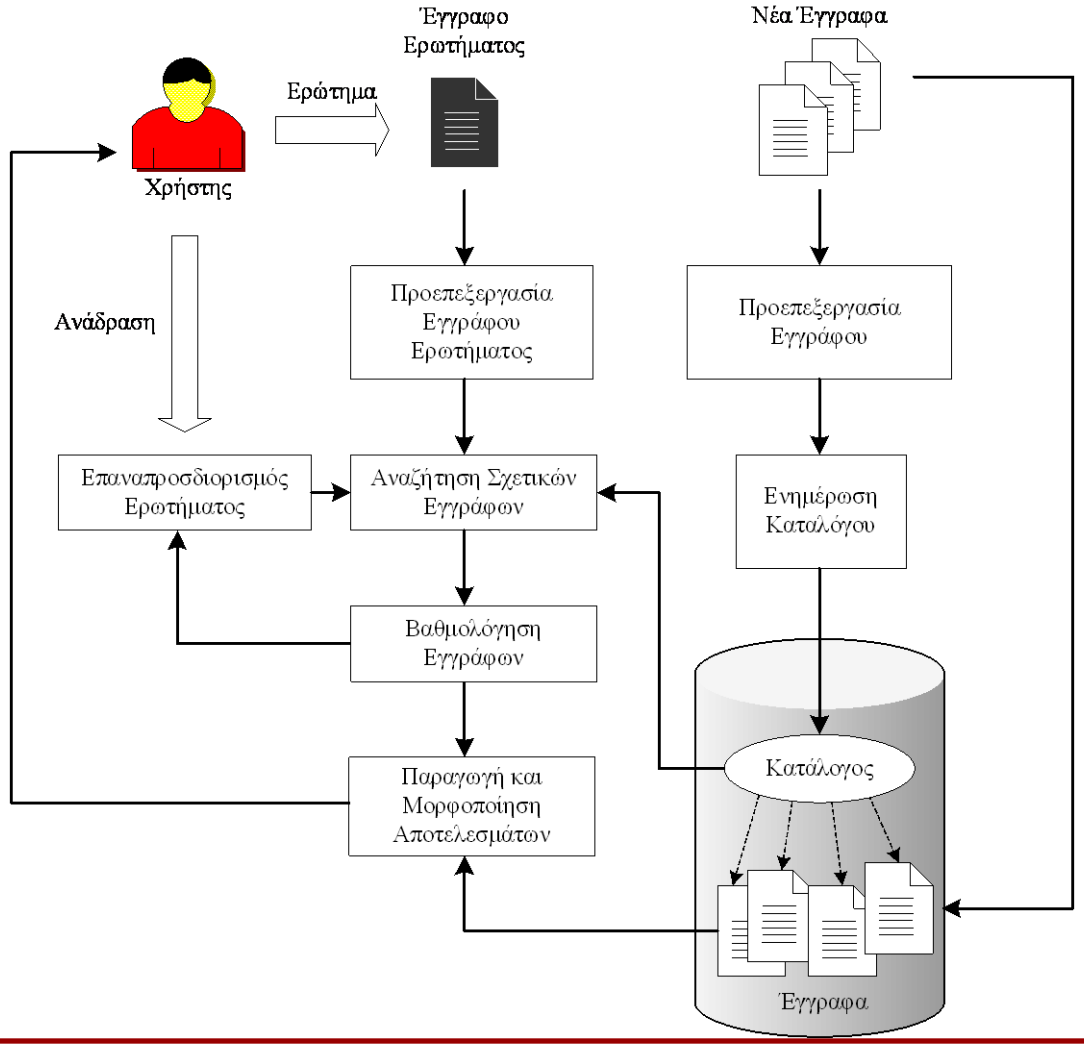
Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο
ΕΠΙΧΕΙΡΗΣΙΑΚΟ ΠΡΟΓΡΑΜΜΑ
ΕΚΠΑΙΔΕΥΣΗ ΚΑΙ ΔΙΑ ΒΙΟΥ ΜΑΘΗΣΗ
επένδυση στην κοινωνία της γνώσης
ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ ΚΑΙ ΘΡΗΣΚΕΥΜΑΤΩΝ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ
ΕΣΠΑ 2007-2013
πρόγραμμα για την ανάπτυξη
ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ
Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

Περιεχόμενα ενότητας

1. Χρησιμότητα καταλόγων.
2. Δομή του Αντεστραμμένου Καταλόγου.
3. Επεξεργασία ερωτημάτων.
4. Κατασκευή καταλόγου.
5. Συντήρηση (μετά από εισαγωγή εγγράφων).
6. Συμπίεση.

Χρησιμότητα καταλόγων

Δομή ενός ΣΑΠ



Χρήση Καταλόγων-1

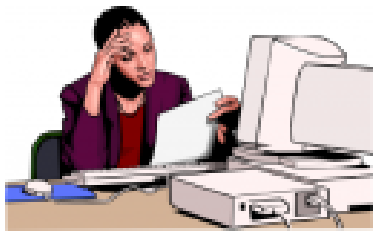
- Με ποιους τρόπους μπορούμε να αναζητήσουμε πληροφορία από μία συλλογή κειμένων;
- Ο πιο απλός και εύκολα υλοποιήσιμος τρόπος είναι να ψάξουμε σειριακά όλα τα κείμενα της συλλογής.
- Ένας άλλος τρόπος είναι να χτίσουμε ειδικές δομές δεδομένων (index structures) ώστε να επιταχύνουμε τη διαδικασία αναζήτησης.

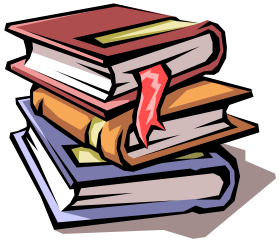
Χρήση Καταλόγων-2

- Η χρήση καταλόγων είναι ευρεία στα συστήματα βάσεων δεδομένων (π.χ. Oracle, MySQL, SQLserver).
- Οι κατάλογοι έχουν την ικανότητα να απορρίπτουν ένα μεγάλο τμήμα των δεδομένων το οποίο δεν συμμετέχει στην απάντηση.
- Παραδείγματα καταλόγων: Β-δένδρα, Κατακερματισμός (hashing).

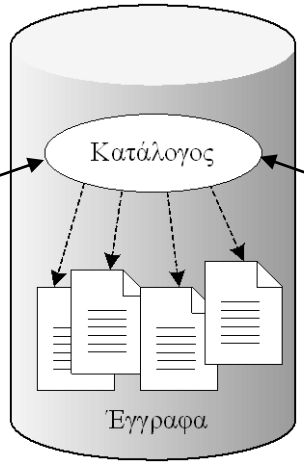
Χρήση Καταλόγων-3

- Τα συστήματα ανάκτησης σπάνια αναζητούν την πληροφορία απευθείας στη συλλογή εγγράφων. Συνήθως, χρησιμοποιούνται **κατάλογοι** οι οποίοι **επιταχύνουν** τη διαδικασία αναζήτησης.





συλλογή εγγράφων



αναζήτηση καταλόγου

δημιουργία καταλόγου

Συλλογή Εγγράφων

- *d1* Ο κομήτης του Χάλλεϋ μας επισκέπτεται περίπου κάθε εβδομήντα έξι χρόνια.
- *d2* Ο κομήτης του Χάλλεϋ ανακαλύφθηκε από τον αστρονόμο Έντμοντ Χάλλεϋ.
- *d3* Ένας κομήτης διαγράφει ελλειπτική τροχιά.
- *d4* Ο πλανήτης Άρης έχει δύο φυσικούς δορυφόρους, το Δείμο και το Φόβο.
- *d5* Ο πλανήτης Δίας έχει εξήντα τρεις γνωστούς φυσικούς δορυφόρους.
- *d6* Ο Άρης είναι ένας πλανήτης του ηλιακού μας συστήματος.

Δομή του Αντεστραμμένου Καταλόγου

Αντεστραμμένος Κατάλογος-1

- Αποτελείται από δύο βασικά τμήματα:
  - Το **λεξικό όρων**, που απαρτίζεται από τους διαφορετικούς όρους που εμφανίζονται στα έγγραφα.
  - Τις **λίστες εμφανίσεων** των όρων, που αναφέρουν για κάθε όρο σε ποια έγγραφα εμφανίζεται.

Αντεστραμμένος Κατάλογος-2

λεξικό όρων		λίστες εμφανίσεων
ο	→	[5: d_1, d_2, d_4, d_5, d_6]
κομήτης	→	[3: d_1, d_2, d_3]
του	→	[3: d_1, d_2, d_6]
Χάλλεϋ	→	[2: d_1, d_2]
μας	→	[2: d_1, d_6]
επισκέπτεται	→	[1: d_1]
περίπου	→	[1: d_1]
κάθε	→	[1: d_1]
εβδομήντα	→	[1: d_1]
έξι	→	[1: d_1]
χρόνια	→	[1: d_1]

αντεστραμμένος κατάλογος επιπέδου εγγράφων

Αντεστραμμένος Κατάλογος-3

ο	→	[5: (d_1, 1), (d_2, 1), (d_4, 1), (d_5, 1), (d_6, 1)]
κομήτης	→	[3: (d_1, 2), (d_2, 2), (d_3, 2)]
του	→	[3: (d_1, 3), (d_2, 3), (d_6, 6)]
Χάλλεϋ	→	[2: (d_1, 4), (d_2, 4, 10)]
μας	→	[2: (d_1, 5), (d_6, 8)]
επισκέπτεται	→	[1: (d_1, 6)]
περίπου	→	[1: (d_1, 7)]
κάθε	→	[1: (d_1, 8)]
εβδομήντα	→	[1: (d_1, 9)]
έξι	→	[1: (d_1, 10)]
χρόνια	→	[1: (d_1, 11)]

αντεστραμμένος κατάλογος επιπέδου όρων

Επεξεργασία ερωτημάτων

Επεξεργασία Λογικών Εκφράσεων

- Ο αντεστραμμένος κατάλογος προσφέρει σημαντική βοήθεια στην επεξεργασία λογικών εκφράσεων (Boolean model).
- Με βάση τον προηγούμενο κατάλογο να βρεθούν τα έγγραφα που ικανοποιούν τη λογική έκφραση

  κομήτης AND Χάλλεϋ

  κομήτης OR Χάλλεϋ

Υποστήριξη Διανυσματικού Μοντέλου

- Με βάση το Διανυσματικό μοντέλο, μία από τις συναρτήσεις ομοιότητας είναι η:
- $S(q,d)= \frac{1}{Lq \cdot Ld} \cdot \sum_{t \in T_{q,d}} (1 + \ln(f_{t,d})) \cdot \ln(1 + \frac{N}{N_t})$

L_q, L_d είναι τα μέτρα των διανυσμάτων.
$T_{q,d}$είναι το σύνολο των όρων κοινών σε *q* και *d*.

n_t είναι το πλήθος των εγγράφων που περιέχουν τον όρο *t*.

N είναι το πλήθος των εγγράφων της συλλογής.

$f_{t,d}$ είναι η συχνότητα εμφάνισης του όρου *t* στο έγγραφο *d*.

Εύρεση Top-k χωρίς Κατάλογο

Αλγόριθμος Top-k-exhaustive ($\mathcal{D}$, q, k)
$\mathcal{D}$: συλλογή εγγράφων
q: ερώτημα
k: πλήθος εγγράφων της απάντησης

---

1. για κάθε όρο $t \in \mathcal{T}_q$ υπολογισμός της ποσότητας $IDF_t = \ln(1 + N/n_t)$
2. για κάθε έγγραφο $d \in \mathcal{D}$
   - 2.1. αρχικοποίηση $score_d = 0$
   - 2.2. για κάθε όρο $t \in \mathcal{T}_q$
     υπολογισμός της ποσότητας $TF_{t,d} = 1 + \ln(f_{t,d})$
     ενημέρωση $score_d = score_d + TF_{t,d} \cdot IDF_t$
   - 2.3. υπολογισμός της ποσότητας L_d
   - 2.4. ενημέρωση $score_d = score_d / L_d$
3. επιστροφή των k εγγράφων με τη μεγαλύτερη τιμή $score_d$

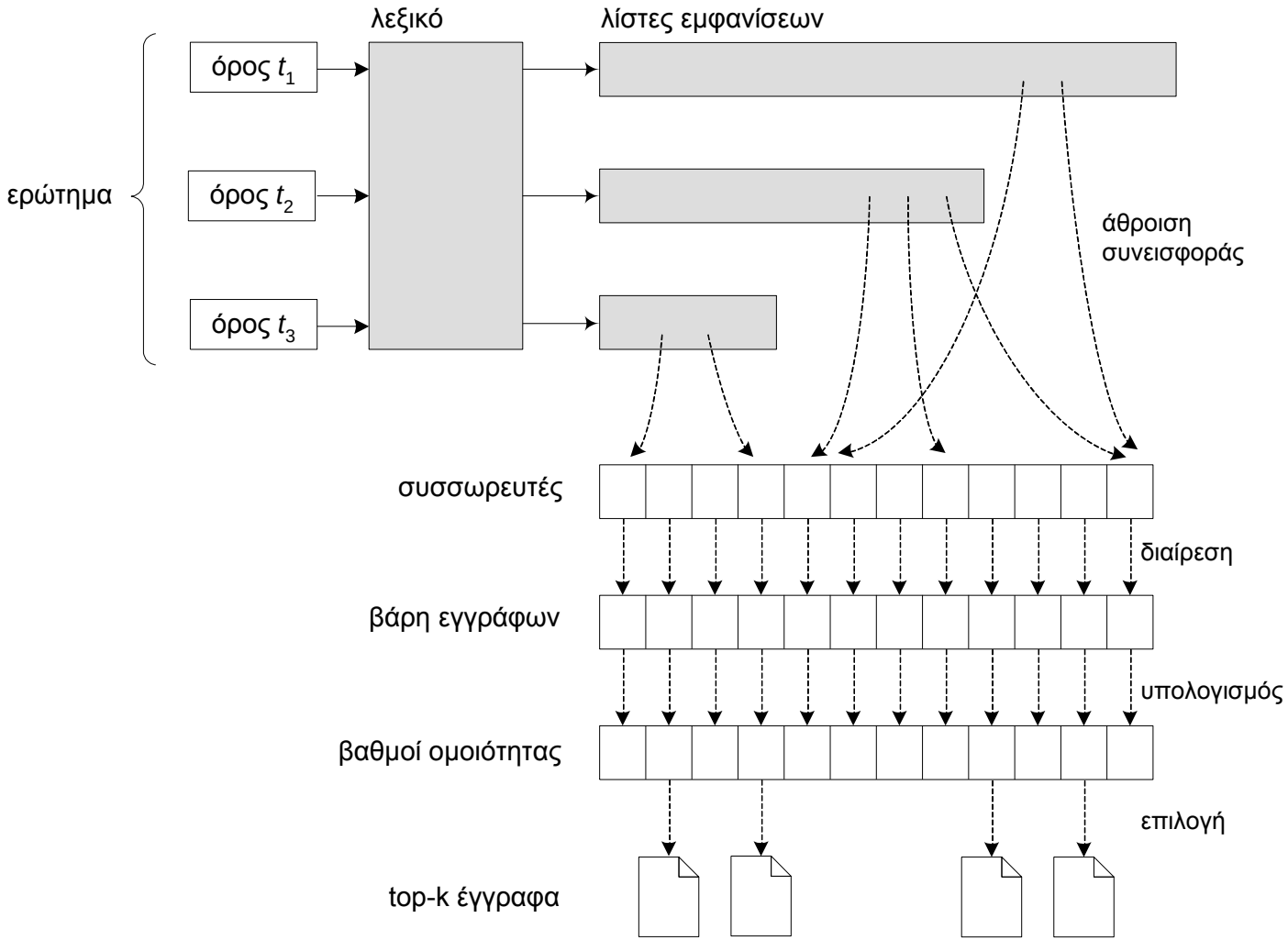
Εύρεση Top-k με Κατάλογο-1

Αλγόριθμος Top-k-inverted ($\mathcal{D}$, q, k)
$\mathcal{D}$: συλλογή εγγράφων
q: ερώτημα
k: πλήθος εγγράφων της απάντησης

---

1. αρχικοποίηση $\Sigma = \emptyset$ (σύνολο σωσσωρευτών)
2. για κάθε όρο $t \in \mathcal{T}_q$
   - 2.1. αναζήτηση του t στο λεξικό
   - 2.2. ανάγνωση της τιμής n_t
   - 2.3. υπολογισμός της ποσότητας $IDF_t = \ln(1 + N/n_t)$
   - 2.4. για κάθε ζεύγος $(d, f_{t,d})$ της λίστας εμφανίσεων του t
     - 2.4.1. αν δεν υπάρχει ο συσσωρευτής Σ_d τότε δημιουργείται
     - 2.4.2. υπολογισμός της ποσότητας $TF_{t,d} = 1 + \ln(f_{t,d})$
     - 2.4.3. ενημέρωση του συσσωρευτή $\Sigma_d = \Sigma_d + TF_{t,d} \cdot IDF_t$
3. για κάθε συσσωρευτή Σ_d ενημέρωση $\Sigma_d = \Sigma_d / L_d$
4. επιλογή των k μεγαλύτερων συσσωσρευτών και επιστροφή των αποτελεσμάτων

Εύρεση Top-k με Κατάλογο-2



Κατασκευή καταλόγου

Δημιουργία Καταλόγου-1

- Έως τώρα, θεωρήσαμε δεδομένη την ύπαρξη του καταλόγου και μελετήσαμε μεθόδους επεξεργασίας ερωτημάτων.
- Η δημιουργία του καταλόγου αποτελεί ενδιαφέρον πρόβλημα που απαιτεί αποδοτική λύση. Εδώ θα εξετάσουμε την περίπτωση της δημιουργίας του αντεστραμμένου καταλόγου για μία δεδομένη συλλογή εγγράφων. Η διαδικασία αυτή καλείται και αντιστροφή (inversion) της συλλογής.
- Η αντιστροφή χρησιμοποιείται σε περιπτώσεις όπου τα περισσότερα έγγραφα της συλλογής είναι γνωστά. Περαιτέρω αλλαγές στον κατάλογο είναι εφικτές χρησιμοποιώντας τις λειτουργίες συντήρησης (εισαγωγή νέου εγγράφου, διαγραφή υπάρχοντος εγγράφου).

Δημιουργία Καταλόγου-2

- Τρεις είναι οι βασικές μέθοδοι:
  - αντιστροφή **κύριας μνήμης.**
  - αντιστροφή με **ταξινόμηση.**
  - αντιστροφή με **συγχώνευση.**

Δημιουργία Καταλόγου στη RAM-1

- Η πρώτη τεχνική αντιστροφής βασίζεται αποκλειστικά στην **κύρια μνήμη** (RAM) του συστήματος.
- Επομένως, για πολύ μεγάλες συλλογές εγγράφων, αυτή η μέθοδος αντιστροφής μπορεί να είναι προβληματική στην περίπτωση που η κύρια μνήμη δεν είναι αρκετή για την αποθήκευση του λεξικού και των λιστών εμφανίσεων των όρων.
- Ο αλγόριθμος αντιστροφής στην κύρια μνήμη διαβάζει δύο φορές τη συλλογή των εγγράφων D. Κατά την πρώτη ανάγνωση υπολογίζονται οι συχνότητες εμφάνισης των όρων στα έγγραφα, ενώ κατά το δεύτερο πέρασμα τοποθετούνται οι δείκτες των λιστών εμφανίσεων στις κατάλληλες θέσεις, χρησιμοποιώντας τη δυνατότητα της τυχαίας προσπέλασης που προσφέρει η κύρια μνήμη.

Δημιουργία Καταλόγου στη RAM-2

Αλγόριθμος BuildInvertedIndex-InMemory ($\mathcal{D}$)
$\mathcal{D}$: συλλογή εγγράφων

---

1. πρώτη ανάγνωση της συλλογής $\mathcal{D}$, όπου γιά κάθε όρο t
   1.1. υπολογίζεται το πλήθος των εγγράφων που τον περιέχουν (n_t)
   1.2. υπολογίζεται ένα άνω όριο (u_t) για το μήκος της λίστας εμφανίσεων
2. δεσμεύεται στη μνήμη χώρος μεγέθους $\sum u_t$
3. για κάθε t δημιουργείται ένα δείκτης p_t που δείχνει στη λίστα εμφανίσεων
4. δεύτερη ανάγνωση της συλλογής $\mathcal{D}$
   4.1. για κάθε όρο t και κάθε έγγραφο d
      4.1.1 υπολογίζεται η συχνότητα εμφάνισης $f_{t,d}$
      4.1.2 αποθηκεύεται ο κωδικός του εγγράφου και ο αριθμός $f_{t,d}$ στη θέση p_t
      4.1.3 αποθηκεύονται οι θέσεις του όρου t στο έγγραφο d
      4.1.4 μετακινείται ο δείκτης p_t μία θέση δεξιά
5. ανάγνωση του καταλόγου, συμπίεση και αποθήκευση στο δίσκο

Δημιουργία Καταλόγου στη RAM-3

- Τα δύο βασικά προβλήματα της προηγούμενης μεθόδου είναι τα ακόλουθα:
  - (α) απαιτούνται δύο αναγνώσεις της συλλογής εγγράφων, ενώ θα πρέπει να αναζητούμε κάθε φορά τους όρους στα έγγραφα και
  - (β) για μεγάλες συλλογές εγγράφων το μέγεθος του καταλόγου ενδέχεται να δημιουργήσει προβλήματα επάρκειας μνήμης.
- Τα προβλήματα της αντιστροφής εξ' ολοκλήρου στην κύρια μνήμη μπορούν να επιλυθούν χρησιμοποιώντας **εναλλακτικές μεθόδους αντιστροφής**.

Αντιστροφή με Ταξινόμηση-1

- Πραγματοποιείται μία μόνο ανάγνωση της συλλογής, κατά την οποία δημιουργείται ένα ενδιάμεσο αρχείο στο δίσκο που περιέχει εγγραφές της μορφής (t, d, ft,d).
- Αρχικά, το αρχείο είναι ταξινομημένο ως προς τους κωδικούς των εγγράφων, αφού επεξεργαζόμαστε τα αρχεία σειριακά.
- Στη συνέχεια, το ενδιάμεσο αρχείο ταξινομείται ως προς τους όρους και για κάθε όρο t υπολογίζεται το πλήθος των εγγράφων nt που τον περιέχουν.
- Η ταξινόμηση μπορεί να πραγματοποιηθεί χρησιμοποιώντας τη μέθοδο ταξινόμησης με συγχώνευση. Το ενδιάμεσο αρχείο χωρίζεται σε τμήματα, στη συνέχεια κάθε τμήμα ταξινομείται στην κύρια μνήμη και τέλος εγγράφεται στο δίσκο.
- Ακολουθεί η διαδικασία της συγχώνευσης, αποτέλεσμα της οποίας είναι το ταξινομημένο ενδιάμεσο αρχείο.
- Τέλος, κατασκευάζεται ο αντεστραμμένος κατάλογος. Με προσεκτικό σχεδιασμό, η μέθοδος αυτή μπορεί να λειτουργήσει χωρίς να δημιουργηθεί θέμα επάρκειας της κύριας μνήμης.

Αντιστροφή με Ταξινόμηση-2

Αλγόριθμος BuildInvertedIndex-Sorting ($\mathcal{D}$)
$\mathcal{D}$: συλλογή εγγράφων

---

1. για κάθε έγγραφο $d \in \mathcal{D}$
   1.1. για κάθε όρο $t \in d$ εύρεση της ποσότητας $f_{t,d}$
   1.2. δημιουργία της εγγραφής $(t, d, f_{t,d})$
2. για κάθε τμήμα του ενδιάμεσου αρχείου
   2.1. ανάγνωση του τμήματος στην κύρια μνήμη
   2.2. ταξινόμηση του τμήματος
   2.3. εγγραφή του ταξινομημένου τμήματος στο δίσκο
3. συγχώνευση των ταξινομημένων τμημάτων
4. ανάγνωση του ταξινομημένου αρχείου και σταδιακή κατασκευή του καταλόγου

Αντιστροφή με Ταξινόμηση: παράδειγμα-1

- Συλλογή εγγράφων

  d1 : Ο κομήτης του Χάλεϋ μας επισκέπτεται περίπου κάθε εβδομήντα έξι χρόνια.

  d2 : Ο κομήτης του Χάλεϋ πήρε το όνομά του από τον αστρονόμο Έντμοντ Χάλεϋ.

  d3 : Ένας κομήτης διαγράφει ελλειπτική τροχιά.

  d4 : Ο πλανήτης Άρης έχει δύο φυσικούς δορυφόρους, το Δείμο και το Φόβο.

  d5 : Ο πλανήτης Δίας έχει 63 γνωστούς φυσικούς δορυφόρους.

  d6 : Ένας κομήτης έχει μικρότερη διάμετρο από ότι ένας πλανήτης.

  d7 : Ο Άρης είναι ένας πλανήτης του ηλιακού μας συστήματος.

Θα εξετάσουμε τη μέθοδο για τα δύο πρώτα έγγραφα της συλλογής.

Αντιστροφή με Ταξινόμηση: παράδειγμα-2

- Οι τριάδες που παράγονται από το d1 είναι:

(ο, d1, 1), (κομήτης, d1, 1), (του, d1, 1), (Χάλλεϋ, d1, 1), (μας, d1, 1),

(επισκέπτεται, d1, 1), (περίπου, d1, 1), (κάθε, d1, 1),

(εβδομήντα, d1, 1), (έξι, d1, 1), (χρόνια, d1, 1)

- και από το d2:

(ο, d2, 1), (κομήτης, d2, 1), (του, d2, 2), **(Χάλλεϋ, d2, 2)**, (πήρε, d2, 1),

(το, d2, 1), (όνομα, d2, 1), (από, d2, 1), (τον, d2, 1),

(αστρονόμο, d2, 1), (Έντμοντ, d2, 1)

- Αν ενώσουμε τα δύο τμήματα τότε οι τριάδες είναι ταξινομημένες ως προς το document id (δεύτερο πεδίο). Χρησιμοποιώντας έναν αλγόριθμο εξωτερικής ταξινόμησης παίρνουμε το αρχείο ταξινομημένο ως προς το πρώτο πεδίο (όρος).

Αντιστροφή με Ταξινόμηση: παράδειγμα-3

- Για τα δύο προηγούμενα έγγραφα έχουμε:

(**από**, d2, 1), (**αστρονόμο**, d2, 1), (**εβδομήντα**, d1, 1),

(**Έντμοντ**, d2, 1), (**έξι**, d1, 1), (**επισκέπτεται**, d1, 1), (**κάθε**, d1, 1),

(**κομήτης, d1, 1**), (**κομήτης, d2, 1**), (**μας**, d1, 1), (**ο, d1, 1**),

(**ο, d2, 1**), (**όνομα**, d2, 1), (**περίπου**, d1, 1), (**πήρε**, d2, 1), (**το**, d2, 1),

(**τον**, d2, 1), (**του, d1, 1**), (**του, d2, 2**), (**Χάλλεϋ, d1, 1**),

(**Χάλλεϋ, d2, 2**), (**χρόνια**, d1, 1)

- Είναι προφανές ότι τριάδες που αναφέρονται στους ίδιους όρους θα είναι γειτονικοί στην τελική διάταξη (π.χ. τα μπλε είναι δίπλα στα κόκκινα).

Αντιστροφή με Συγχώνευση-1

- Πραγματοποιείται και πάλι μία μόνο ανάγνωση της συλλογής εγγράφων, κατά την οποία δημιουργείται ο αντεστραμμένος κατάλογος στην κύρια μνήμη.
- Όταν η ελεύθερη εξαντληθεί, τότε το τμήμα του καταλόγου που έχει δημιουργηθεί μαζί με το τμήμα του λεξικού αποθηκεύονται στο δίσκο, ενώ διαγράφονται τα δεδομένα από την κύρια μνήμη.
- Επαναλαμβάνεται η ίδια διαδικασία μέχρι να ολοκληρωθεί η ανάγνωση και η επεξεργασία της συλλογής.
- Στη συνέχεια, τα τμήματα που έχουν αποθηκευθεί στο δίσκο συγχωνεύονται ώστε να παραχθεί ο τελικός συνολικός αντεστραμμένος κατάλογος.
- Σύμφωνα με πειραματικές μελέτες, η μέθοδος της αντιστροφής με συγχώνευση είναι πολύ αποδοτική ακόμη και για μεγάλες συλλογές εγγράφων.

Αντιστροφή με Συγχώνευση-2

Αλγόριθμος BuildInvertedIndex-Merging ($\mathcal{D}$)
$\mathcal{D}$: συλλογή εγγράφων

1. μέχρι να εξαντληθεί η ελεύθερη μνήμη
    1.1. ανάγνωση του επόμενου εγγράφου $d \in \mathcal{D}$
    1.2. ενημέρωση του καταλόγου για τους όρους $t \in d$
2. εγγραφή του τμήματος του καταλόγου στο δίσκο
3. αν υπάρχουν και άλλα αρχεία, επανάληψη από το βήμα 1.
4. συγχώνευση των τμημάτων καταλόγου που έχουν δημιουργηθεί

Μετά από εισαγωγή εγγράφων

Συντήρηση

Συντήρηση Καταλόγου-1

- Εάν δεν υπάρχουν αλλαγές στη συλλογή (δεν εισάγονται ούτε διαγράφονται έγγραφα) τότε η μέθοδος της αντιστροφής εκτελείται μία μόνο φορά.
- Ωστόσο, σε πολλές περιπτώσεις επιβάλλεται η υποστήριξη της ενημέρωσης του καταλόγου λόγω της εισαγωγής νέων εγγράφων (ή της διαγραφής παλαιών).
- Είναι προφανές, ότι σε μία τέτοια περίπτωση θα πρέπει ο κατάλογος να τροποποιηθεί ώστε να ανταποκρίνεται πλήρως στη συλλογή εγγράφων.
- Δυστυχώς, η άμεση ενημέρωση του καταλόγου με στόχο την προσθήκη ενός νέου εγγράφου είναι αρκετά χρονοβόρα, ιδιαίτερα για πολύ μεγάλα έγγραφα.
- Για το λόγο αυτό, έχουν προταθεί εναλλακτικές μέθοδοι ενημέρωσης του καταλόγου με σκοπό τη **μείωση του κόστους ενημέρωσης**.

Συντήρηση Καταλόγου-2

- Τρεις είναι οι βασικές τεχνικές:
  - αναδόμηση καταλόγου.
  - ενημέρωση με συγχώνευση.
  - σταδιακή ενημέρωση.

Συντήρηση Καταλόγου-3

- Αναδόμηση καταλόγου
  - Σύμφωνα με την τεχνική αυτή, συλλέγονται νέα έγγραφα προς εισαγωγή και στη συνέχεια πραγματοποιείται μία εκ νέου δημιουργία του καταλόγου.
  - Ο προηγούμενος κατάλογος διαγράφεται.

Συντήρηση Καταλόγου-4

- Ενημέρωση με συγχώνευση.
  - Για τα νέα έγγραφα δημιουργείται ένας προσωρινός κατάλογος που διατηρείται στην κύρια μνήμη του συστήματος.
  - Όταν εξαντληθεί ο ελεύθερος χώρος στη μνήμη, τότε εφαρμόζεται η τεχνική της συγχώνευσης κατά την οποία συγχωνεύονται τα περιεχόμενα του υπάρχοντος καταλόγου που βρίσκονται στο δίσκο με τα περιεχόμενα του προσωρινού καταλόγου της κύριας μνήμης.

Συντήρηση Καταλόγου-5

- Σταδιακή ενημέρωση
  - Σύμφωνα με τη μέθοδο αυτή, ο κατάλογος ενημερώνεται σταδιακά για κάθε όρο όταν αυτό είναι εφικτό. Για την ενημέρωση της λίστας εμφανίσεων ενός όρου t η λίστα διαβάζεται από το δίσκο στην κύρια μνήμη, ενημερώνεται και στη συνέχεια αποθηκεύεται πάλι στο δίσκο.
  - Η λειτουργία αυτή πρέπει να καθυστερεί όσο γίνεται περισσότερο ώστε να αποφεύγεται κατά το δυνατόν η πολλαπλή ανάγνωση και αποθήκευση της ίδιας λίστας. Ένας εύκολος τρόπος να γίνει αυτό είναι να συλλέγονται τα δεδομένα στην κύρια μνήμη, και όταν το μέγεθος των δεδομένων ξεπεράσει κάποιο όριο τότε να πραγματοποιείται η ενημέρωση των λιστών εμφάνισης.
  - Εναλλακτικά, μία λίστα μπορεί να ενημερωθεί όταν απαιτηθεί η ανάγνωσή της κατά τη διαδικασία επεξεργασίας κάποιου ερωτήματος.
  - Οι τεχνικές αυτές είναι λιγότερο αποδοτικές σε σχέση με την ενημέρωση με συγχώνευση.

Συμπίεση

Συμπίεση-1

- Κίνητρο
  - Το μέγεθος που καταλαμβάνουν οι λίστες εμφανίσεων ενός αντεστραμμένου καταλόγου είναι πολύ σημαντικό και πολλές φορές ξεπερνά το μέγεθος της συλλογής εγγράφων.
- Πιθανή λύση: συμπίεση καταλόγου.

Συμπίεση-2

- Με τη συμπίεση της δομής πετυχαίνουμε δύο σημαντικούς στόχους:
  - εξοικονομείται πολύτιμος χώρος στην κύρια μνήμη και
  - μειώνεται ο αριθμός των προσπελάσεων στη δευτερεύουσα μνήμη.

Συμπίεση-3

- Εστιάζουμε στον κατάλογο επιπέδου εγγράφων για περισσότερη απλότητα.
- Μία λίστα εμφανίσεων έχει τη μορφή:
- $< x; d_{i1}, d_{i2}, \ldots, d_{ix} >$
- Όπου x είναι το πλήθος των εγγράφων που περιέχουν τον όρο και d_{ij} είναι ένας κωδικός εγγράφου.

Συμπίεση-4

- Επειδή η λίστα των κωδικών συνήθως αποθηκεύεται σε αύξουσα διάταξη, η λίστα μπορεί να αναπαρασταθεί με μία ακολουθία διαφορών.
- Π.χ.

  <7; 134, 45, 130, 20, 10, 100, 120>

  <7; 10, 20, 45, 100, 120, 130, 134>

  <7; 10, 10, 25, 55, 20, 10, 4>

Συμπίεση-5

- Είναι προφανές ότι αναμένονται μικρές διαφορές κωδικών για όρους που εμφανίζονται σε πολλά έγγραφα, ενώ αναμένονται μεγαλύτερες διαφορές για όρους που δεν είναι συχνοί.
- Αυτή η παρατήρηση οδηγεί στην ιδέα να χρησιμοποιηθούν κωδικοί μεταβλητού μήκους, οι οποίοι :
  - δίνουν λίγα δυαδικά ψηφία σε συχνά εμφανιζόμενους αριθμούς και
  - δίνουν περισσότερα δυαδικά ψηφία στην αντίθετη περίπτωση.

Συμπίεση-6

- Μοναδιαίος κώδικας (unary code)
  - Ένας ακέραιος x κωδικοποιείται με x-1 άσσους και ένα μηδενικό στο τέλος. Π.χ. αν x=5 τότε ο κώδικας είναι 11110

Συμπίεση-7

- Elias-γ (Elias 1975)
- Ο κώδικας ενός ακεραίου x αποτελείται από δύο τμήματα. Το πρώτο τμήμα είναι ο μοναδιαίος κωδικός του αριθμού

$$1 + \lfloor \log x \rfloor$$

- και το δεύτερο τμήμα είναι ένας κωδικός που αποτελείται από

$$\lfloor \log x \rfloor$$

- δυαδικά ψηφία και αναπαριστά στο δυαδικό σύστημα τον αριθμό

$$x - 2^{\log x}$$

Συμπίεση-8

- Elias-δ (Elias 1975)
- Το πρώτο τμήμα του κωδικού είναι ο αριθμός των δυαδικών ψηφίων που υπάρχουν στον κώδικα Elias-γ του αριθμού *x*.
- Το δεύτερο τμήμα παράγεται όπως και προηγουμένως.

Συμπίεση-9

- Elias-δ (Elias 1975)
- Το πρώτο τμήμα του κωδικού είναι ο αριθμός των δυαδικών ψηφίων που υπάρχουν στον κώδικα Elias-γ του αριθμού *x*.
- Το δεύτερο τμήμα παράγεται όπως και προηγουμένως.

Συμπίεση-10

- Παράδειγμα Elias-γ
- Έστω x=7.

$$1 + \lfloor \log x \rfloor = 1 + 2 = 3$$

- Ο μοναδιαίος κώδικας για το 3 είναι το **110**

$$x - 2^{\lfloor \log x \rfloor} = 7 - 2^2 = 3$$

- Στο δυαδικό σύστημα ο αριθμός αυτός είναι ο **11**
- Άρα ο Elias-γ του 7 είναι ο **11011**

Συμπίεση-11

- Παράδειγμα Elias-δ
- Έστω x=7.
- Το πρώτο τμήμα του κώδικα είναι η δυαδική αναπαράσταση του αριθμού των ψηφίων που έχει ο Elias-γ.
- Από τα προηγούμενα έχουμε ότι ο Elias-γ του 7 είναι ο 11011, άρα έχουμε 5 δυαδικά ψηφία, επομένως το πρώτο τμήμα του Elias-δ είναι **101**.
- Το δεύτερο τμήμα είναι το **11** όπως και προηγουμένως.
- Άρα τελικά **10111**

Συμπίεση-12

- Golomb (1966)
- Χρησιμοποιεί γεωμετρική κατανομή. Η πιθανότητα η διαφορά δύο κωδικών εγγράφων να είναι x δίνεται από τον τύπο, όπου p είναι η πιθανότητα εμφάνισης του όρου σε ένα έγγραφο:
- $P(x) = (1-p)^{x-1} \cdot p$ $\qquad p = \frac{K}{N \cdot M}$
  - *N*: πλήθος εγγράφων
  - *M*: πλήθος όρων
  - *K* = σύνολο κωδικών εγγράφων (σε όλες τις λίστες εμφανίσεων)

Συμπίεση-13

- Golomb
- Χρησιμοποιεί την παράμετρο b για να δημιουργήσει τους κωδικούς.
- Εάν b=1 τότε η μέθοδος Golomb ταυτίζεται με την κωδικοποίηση μοναδιαίου κωδικού.
- Η κωδικοποίηση ενός ακεραίου x πραγματοποιείται με τον προσδιορισμό δύο ποσοτήτων, του **πηλίκου** (quotient) που συμβολίζεται με q και του **υπολοίπου** (remainder) που συμβολίζεται με r.

Συμπίεση-14

- Golomb

$$q = \left\lfloor \frac{x-1}{b} \right\rfloor$$

$$r = x - 1 - q \cdot b$$

Συμπίεση-15

- Ο κώδικας Golomb απαρτίζεται από τον αριθμό q+1 στη μορφή μοναδιαίου κωδικού ακολουθούμενο από τον αριθμό r στη δυαδική αναπαράσταση.
- Ο αριθμός r απαιτεί $\lfloor \log b \rfloor$ δυαδικά ψηφία για την αναπαράστασή του σε περίπτωση που $r < 2^{\lfloor \log b \rfloor - 1}$
- Διαφορετικά απαιτεί $\lceil \log b \rceil$ δυαδικά ψηφία

Συμπίεση-16

- Παράδειγμα Golomb
- Έστω x=7 και b=3. $q = \lfloor (7-1)/3 \rfloor = 2$
- Το πρώτο τμήμα είναι ο μοναδιαίος κώδικας του q+1, δηλαδή του 3 που είναι το **110**.
- Το υπόλοιπο στην περίπτωση αυτή είναι

  r = x – 1 –qb = 0
- Άρα ο κώδικας για το 7 είναι ο **1100**.
- Με τον ίδιο τρόπο βρίσκουμε ότι:
- Golomb(7,4) = **1010**
- Golomb(7,5) = **1001**

Συμπίεση-17

- Golomb
- Σύμφωνα με τους Gallager και van Voorhis η κωδικοποίηση Golomb κατασκευάζει βέλτιστους κωδικούς για την γεωμετρική κατανομή χωρίς να υπάρχουν κοινά προθέματα (prefix-free) εάν η παράμετρος b επιλεγεί ώστε να ικανοποιεί την παρακάτω ανισότητα:

$$(1-p)^b + (1-p)^{b+1} \leq 1 < (1-p)^{b-1} + (1-p)^{\mathrm{b}}$$

Συμπίεση-18

- Golomb
- Για να ισχύει η ισότητα θα πρέπει

$$b = \left\lceil \frac{\log(2-p)}{-\log(1-p)} \right\rceil$$

Συμπίεση-19

- Golomb
- Η τιμή p είναι συνήθως κατά πολύ μικρότερη της μονάδας, οπότε μπορεί να χρησιμοποιηθεί ο ακόλουθος προσεγγιστικός τύπος για την τιμή του b.

$$b \approx 0.69 \cdot \frac{N \cdot M}{K}$$

Συμπίεση-20

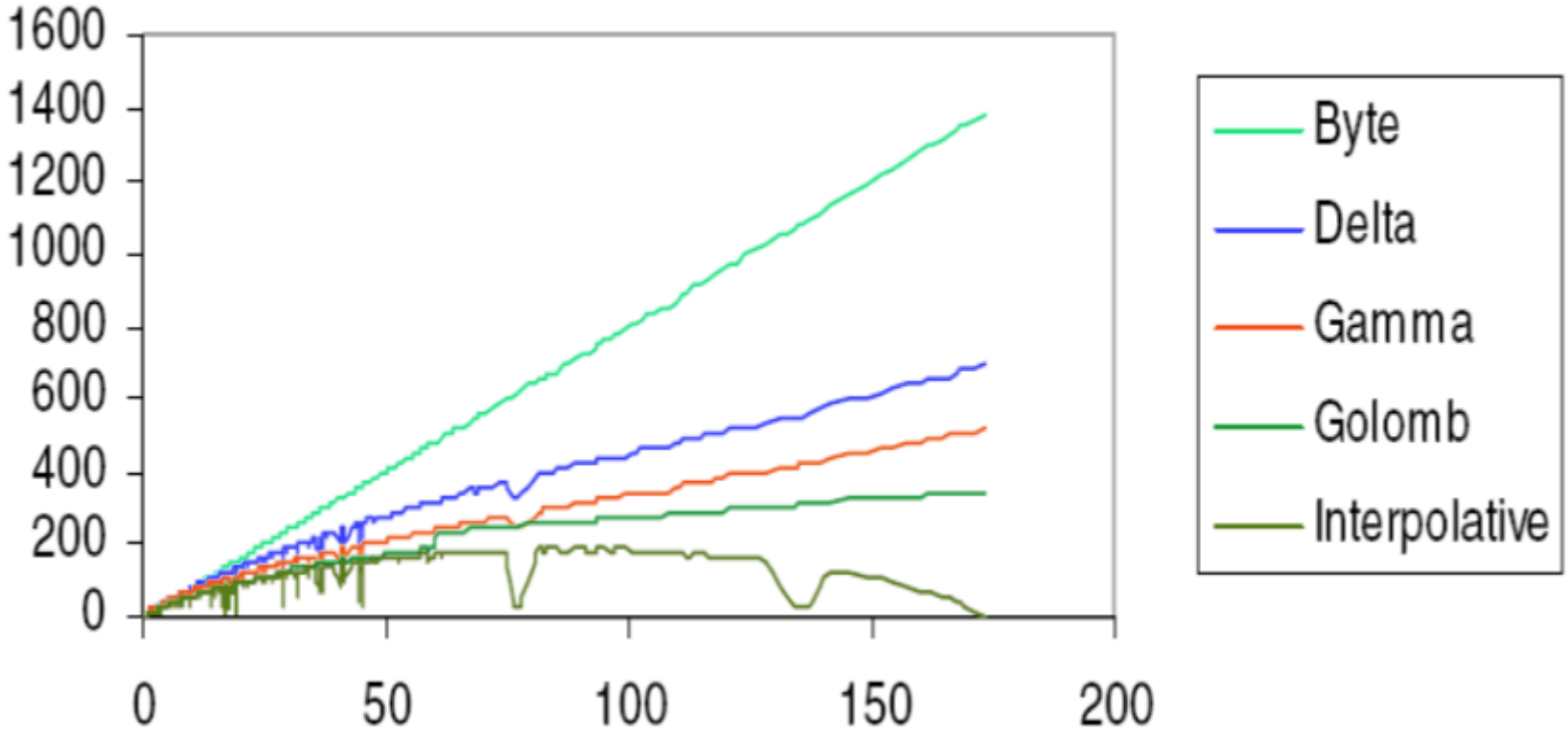
- Έως τώρα θεωρήσαμε ότι η μέθοδος Golomb λειτουργεί με τον ίδιο τρόπο για κάθε λίστα εμφανίσεων.
- Μπορεί να επιτευχθεί καλύτερη απόδοση στη συμπίεση του αντεστραμμένου καταλόγου εάν η κάθε λίστα εμφανίσεων συμπιεστεί ξεχωριστά, με διαφορετικές τιμές παραμέτρων.
- Για κάθε όρο t απαιτείται η γνώση της ποσότητας nt που συμβολίζει το πλήθος των εγγράφων που περιέχουν τον όρο t.
- Αν εστιάζουμε στη λίστα εμφανίσεων ενός όρου t, τότε προφανώς το πλήθος των όρων είναι 1 ενώ αντικαθιστούμε το πλήθος των δεικτών του καταλόγου με το πλήθος των δεικτών της λίστας τού όρου t.
- Αν bt την τιμή της παραμέτρου b για τη λίστα εμφανίσεων του όρου t τότε έχουμε:

$$b \approx 0.69 \cdot \frac{N}{N_t}$$

Συμπίεση-21

x	Μοναδιαίος	Elias-γ	Elias-δ	Golomb b=3
1	0	0	0	00
2	10	100	1000	010
3	110	101	1001	011
4	1110	11000	10100	100
5	11110	11001	10101	1010
6	111110	11010	10110	1011
7	1111110	11011	10111	1100
8	11111110	1110000	11000000	11010
9	111111110	1110001	11000001	11011
10	1111111110	1110010	11000010	11100

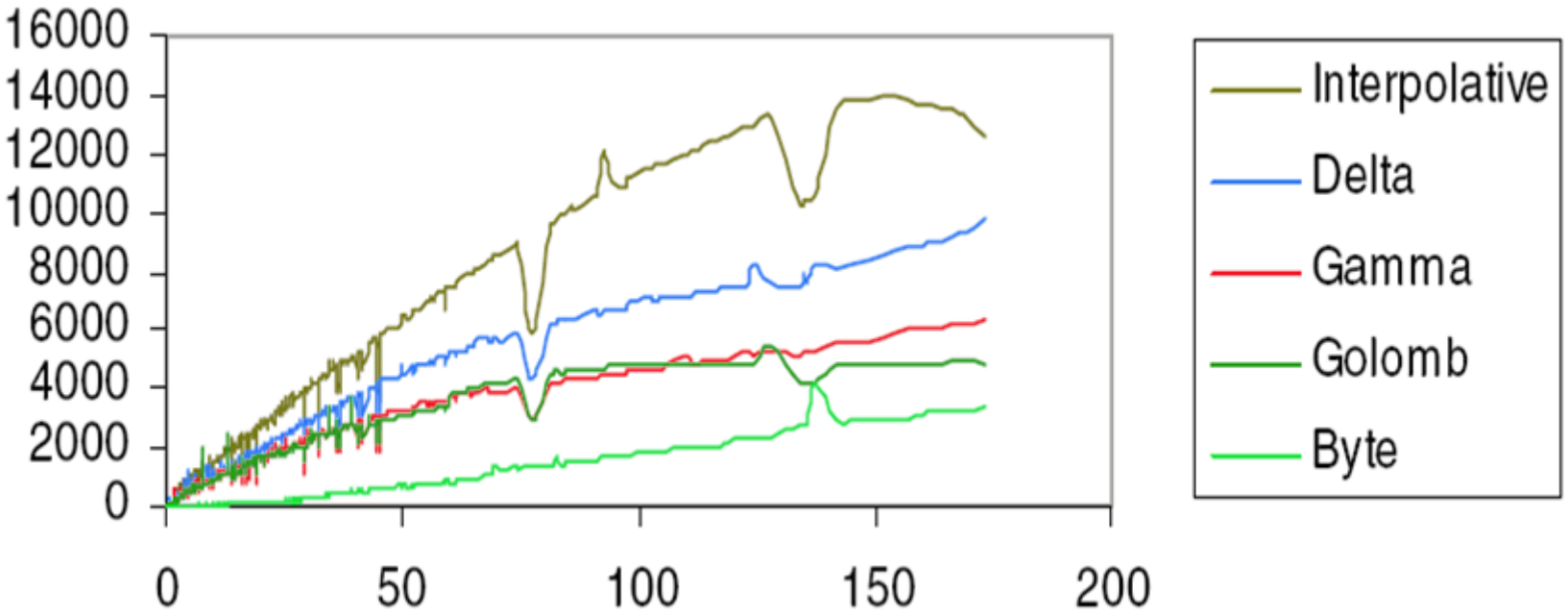
Συμπίεση-22



Αριθμός bits για τη συμπίεση των λιστών εμφάνισης.
Ο άξονας *x* δείχνει το πλήθος των doc IDs στις λίστες εμφάνισης

(*Πηγή: A. Trotman, Compressing Inverted Files, Information Retrieval, 2003*)

Συμπίεση-23



Κύκλοι CPU που απαιτούνται για αποκωδικοποίηση
Ο άξονας *x* δείχνει το πλήθος των doc IDs στις λίστες εμφάνισης.

(*Πηγή: A. Trotman, Compressing Inverted Files, Information Retrieval, 2003*)

Σύνοψη-1

- Ο αντεστραμμένος κατάλογος αποτελεί την πιο διαδεδομένη μέθοδο οργάνωσης μίας συλλογής εγγράφων.
- Αποτελείται από δύο βασικά τμήματα, το λεξικό όρων και τις λίστες εμφανίσεων. Στην απλούστερη μορφή της, κάθε λίστα εμφανίσεων καταγράφει τους κωδικούς των εγγράφων που περιέχουν τον όρο.
- Ωστόσο, στις λίστες εμφανίσεων μπορούν να καταγραφούν και άλλες πληροφορίες όπως το πλήθος των εμφανίσεων του όρου σε ένα έγγραφο ή το πλήθος των εγγράφων που περιέχουν έναν όρο.

Σύνοψη-2

- Κατασκευή.
- Χρήση για αναζήτηση.
- Συντήρηση.
- Συμπίεση.

Σημείωμα Αναφοράς

Σημείωμα Αδειοδότησης

Το παρόν υλικό διατίθεται με τους όρους της άδειας χρήσης Creative Commons Αναφορά - Μη Εμπορική Χρήση - Όχι Παράγωγα Έργα 4.0 [1] ή μεταγενέστερη, Διεθνής Έκδοση. Εξαιρούνται τα αυτοτελή έργα τρίτων π.χ. φωτογραφίες, διαγράμματα κ.λ.π., τα οποία εμπεριέχονται σε αυτό και τα οποία αναφέρονται μαζί με τους όρους χρήσης τους στο «Σημείωμα Χρήσης Έργων Τρίτων».

Ο δικαιούχος μπορεί να παρέχει στον αδειοδόχο ξεχωριστή άδεια να χρησιμοποιεί το έργο για εμπορική χρήση, εφόσον αυτό του ζητηθεί.

Ως **Μη Εμπορική** ορίζεται η χρήση:

- που δεν περιλαμβάνει άμεσο ή έμμεσο οικονομικό όφελος από την χρήση του έργου, για το διανομέα του έργου και αδειοδόχο
- που δεν περιλαμβάνει οικονομική συναλλαγή ως προϋπόθεση για τη χρήση ή πρόσβαση στο έργο
- που δεν προσπορίζει στο διανομέα του έργου και αδειοδόχο έμμεσο οικονομικό όφελος (π.χ. διαφημίσεις) από την προβολή του έργου σε διαδικτυακό τόπο

[1] http://creativecommons.org/licenses/by-nc-nd/4.0/

Τέλος ενότητας

Επεξεργασία: <Μαυρίδης Απόστολος>
Θεσσαλονίκη, <Εαρινό εξάμηνο 2013-2014>

Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο
ΕΠΙΧΕΙΡΗΣΙΑΚΟ ΠΡΟΓΡΑΜΜΑ
ΕΚΠΑΙΔΕΥΣΗ ΚΑΙ ΔΙΑ ΒΙΟΥ ΜΑΘΗΣΗ
επένδυση στην κοινωνία της γνώσης
ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ ΚΑΙ ΘΡΗΣΚΕΥΜΑΤΩΝ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ
ΕΣΠΑ 2007-2013
πρόγραμμα για την ανάπτυξη
ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ
Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

Σημειώματα

Διατήρηση Σημειωμάτων

Οποιαδήποτε αναπαραγωγή ή διασκευή του υλικού θα πρέπει να συμπεριλαμβάνει:

- το Σημείωμα Αναφοράς
- το Σημείωμα Αδειοδότησης
- τη δήλωση Διατήρησης Σημειωμάτων
- το Σημείωμα Χρήσης Έργων Τρίτων (εφόσον υπάρχει)

μαζί με τους συνοδευόμενους υπερσυνδέσμους.