



Εξόρυξη από το Web

12: Web mining

Διαφορές από το Data Mining

- Το web δεν είναι πίνακας
 - Οι ιστοσελίδες έχουν περιεχόμενο, έχουν συνδέσμους
- Τα δεδομένα χρήσης είναι τεράστια και αυξάνονται ταχύτατα
 - Τα δεδομένα χρήσης της Google είναι περισσότερα από τα δεδομένα που φέρνει ο crawler
 - Τα δεδομένα που παράγονται καθημερινά είναι συγκρίσιμα με τα αυτά που διατηρούν μεγάλοι οργανισμοί στις ΒΔ τους
- Συχνά οι παραγόμενες πληροφορίες (π.χ. usage patterns) αξιοποιούνται σε πραγματικό χρόνο (χωρίς να παρεμβάλεται άνθρωπος)

Πόσο μεγάλο είναι το Web;

- Αριθμός σελίδων
 - Τεχνικά, άπειρος
 - Λόγω των δυναμικά παραγόμενων σελίδων
 - Πολλά διπλότυπα (30-40%)
 - Εκτιμήσεις για τις διαφορετικές σταθερές (static) HTML σελίδες έρχονται από τις μηχανές αναζήτησης
 - Google = 8 billion, Yahoo = 20 billion
 - Αλλά δεν είναι αξιόπιστες

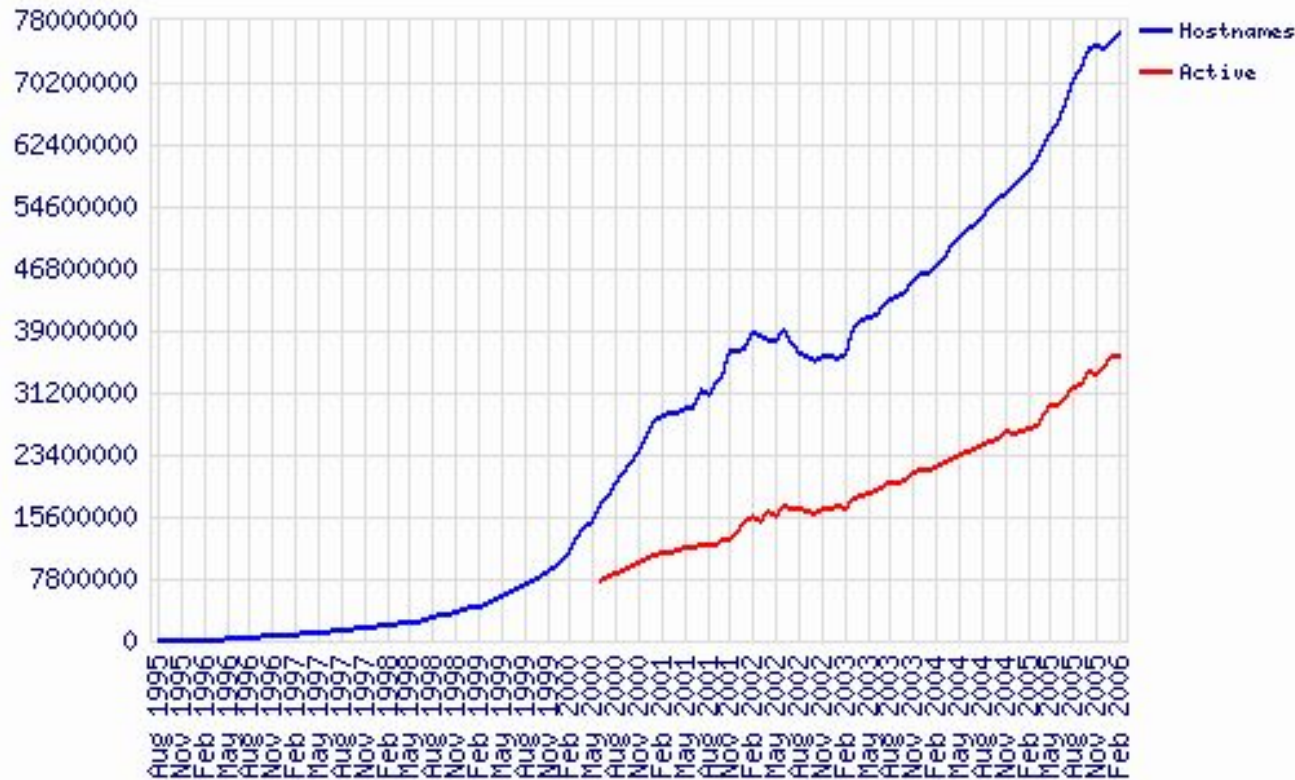
Gregory Piatetsky-Shapiro

"Web Mining: An Introduction" -KDnuggets

76,184,000 web sites (Feb 2006)

Netcraft survey

Total Sites Across All Domains August 1995 - February 2006



http://news.netcraft.com/archives/web_server_survey.html

Τι είναι το Web Mining;

- Η προσπάθεια ανακάλυψης ενδιαφέρουσας και χρήσιμης πληροφορίας από το Web: από το περιεχόμενό του (content) τη δομή του (structure) και τα δεδομένα χρήσης του (usage)
- Παραδείγματα:
 - Αναζήτηση στο Web και στο Hidden Web: π.χ. Google, Yahoo, MSN, Ask,
 - Εξειδικευμένη αναζήτηση: π.χ. Froogle (comparison shopping), job ads (Flipdog)
 - Συστάσεις (Recommendations): π.χ. Netflix, Amazon
 - Διαφήμιση: π.χ. Google Adsense
 - Ανίχνευση απάτης: π.χ. click fraud detection,
 - Βελτίωση της δομής ενός Web site

Θέματα Web Mining

- Crawling the web
- Web graph analysis
- Structured data extraction
- Classification and vertical search
- Deep Web search
- Collaborative filtering
- Web advertising and optimization
- Mining web logs
- Systems Issues

Web Data

WEB MINING

```
graph TD; WM[WEB MINING] --> S[STRUCTURE]; WM --> C[CONTENT]; WM --> U[USAGE]; S --> LA[Link analysis]; C --> IR[IR techniques on web content]; U --> WLA[web log analysis];
```

The diagram illustrates the components of Web Mining. At the top is a gold box labeled 'WEB MINING'. Three lines branch out from its bottom to three colored boxes below: a blue box for 'STRUCTURE', a cyan box for 'CONTENT', and a pink box for 'USAGE'. Each of these boxes has a corresponding white box below it containing specific techniques: 'Link analysis' under Structure, 'IR techniques on web content' under Content, and 'web log analysis' under Usage.

STRUCTURE

Link analysis

CONTENT

IR techniques on
web content

USAGE

web log analysis



Εξόρυξη Δεδομένων

Ανάκτηση Πληροφορίας

Εξόρυξη από κείμενα

Τι κρύβεται πίσω από τις μηχανές αναζήτησης

Βασισμένο στις διαφάνειες του Bing Liu για το
“Web Data Mining”

A diagram illustrating the concept of a web crawler. It features a light blue cloud with the text "WWW" inside. A black arrow points from the cloud down to a small, light blue rectangular box. Inside this box is a small image of a black spider with blue legs, representing a web crawler. The box is partially overlapping a larger, light blue rectangular area that contains the text "crawling".



Γενικά

- **Text Mining**: περιλαμβάνει διάφορες διαδικασίες εξόρυξης γνώσης από δεδομένα. Αυτή τη φορά τα δεδομένα είναι συλλογές κειμένων
- Οι περισσότερες διαδικασίες text mining χρησιμοποιούν μεθόδους ανάκτησης πληροφορίας (**Information Retrieval** - IR) κατά την προ-επεξεργασία των συλλογών
- Πίσω από τις μηχανές αναζήτησης κρύβονται πολλές διαδικασίες text mining και information retrieval
 - Στην αναζήτηση στο web δεν μπορούν να εφαρμοστούν απευθείας τα διάφορα IR μοντέλα,
 - Παρόλα αυτά το web search εισήγαγε πολλά ενδιαφέροντα προβλήματα στο IR

IR για Web Search

- Υπάρχουν διαφορές μεταξύ του παραδοσιακού IR (π.χ. σε μια τοπική συλλογή εγγράφων) και του IR στο Web
 - Τα συστήματα IR δεν λαμβάνουν πάντοτε υπόψη την πολυπλοκότητα, την αποτελεσματικότητα και τις χρονικές επιδόσεις
 - Το Web είναι τεράστιο!
 - Τα IR συστήματα φτιάχτηκαν για κείμενα
 - Οι ιστοσελίδες έχουν **hyperlinks** και **anchor texts**.
 - Οι ιστοσελίδες είναι ημιδομημένες
 - Στο Web υπάρχει καλόβουλο και κακόβουλο περιεχόμενο (spamming)

Περιεχόμενα

- Βασικές έννοιες
- Μοντέλα IR
- Προεπεξεργασία κειμένων και ιστοσελίδων
- Embeddings
- Ανεστραμένα ευρετήρια
- LSI (Latent Semantic Indexing)
- Μέθοδοι αξιολόγησης του IR
- Αναζήτηση στο Web
- Meta-search

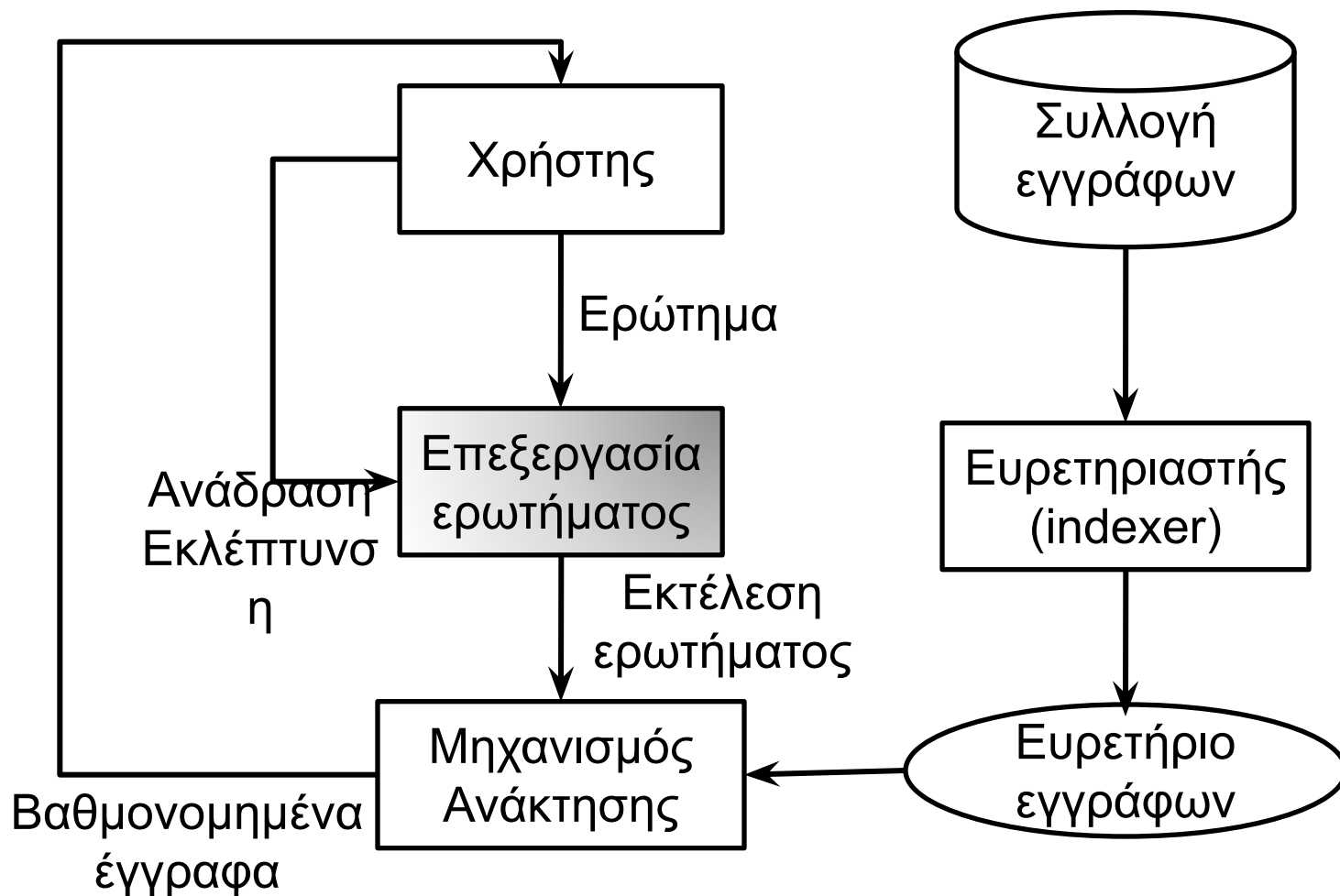
Ανάκτηση πληροφορίας (IR)

- Το IR αφορά στον εντοπισμό εκείνης της πληροφορίας που ικανοποιεί την πληροφοριακή ανάγκη του χρήστη
 - Πληροφοριακή ανάγκη: query
 - Πληροφορία: Κείμενα, έγγραφα, εικόνες, αποσπάσματα, περιλήψεις, κλπ
- Ιστορικά το IR εστιάζει στην ανάκτηση εγγράφων (document retrieval)
 - Στόχος είναι να βρώ έγγραφα που είναι σχετικά με την ερώτηση του χρήστη
- Τεχνικά το IR ασχολείται με τη συλλογή, οργάνωση, αποθήκευση, ανάκτηση και διανομή της πληροφορίας

Ερωτήσεις - queries

- Keyword queries
- Boolean queries (using AND, OR, NOT)
- Phrase queries
- Proximity queries
- Full document queries
- Natural language questions

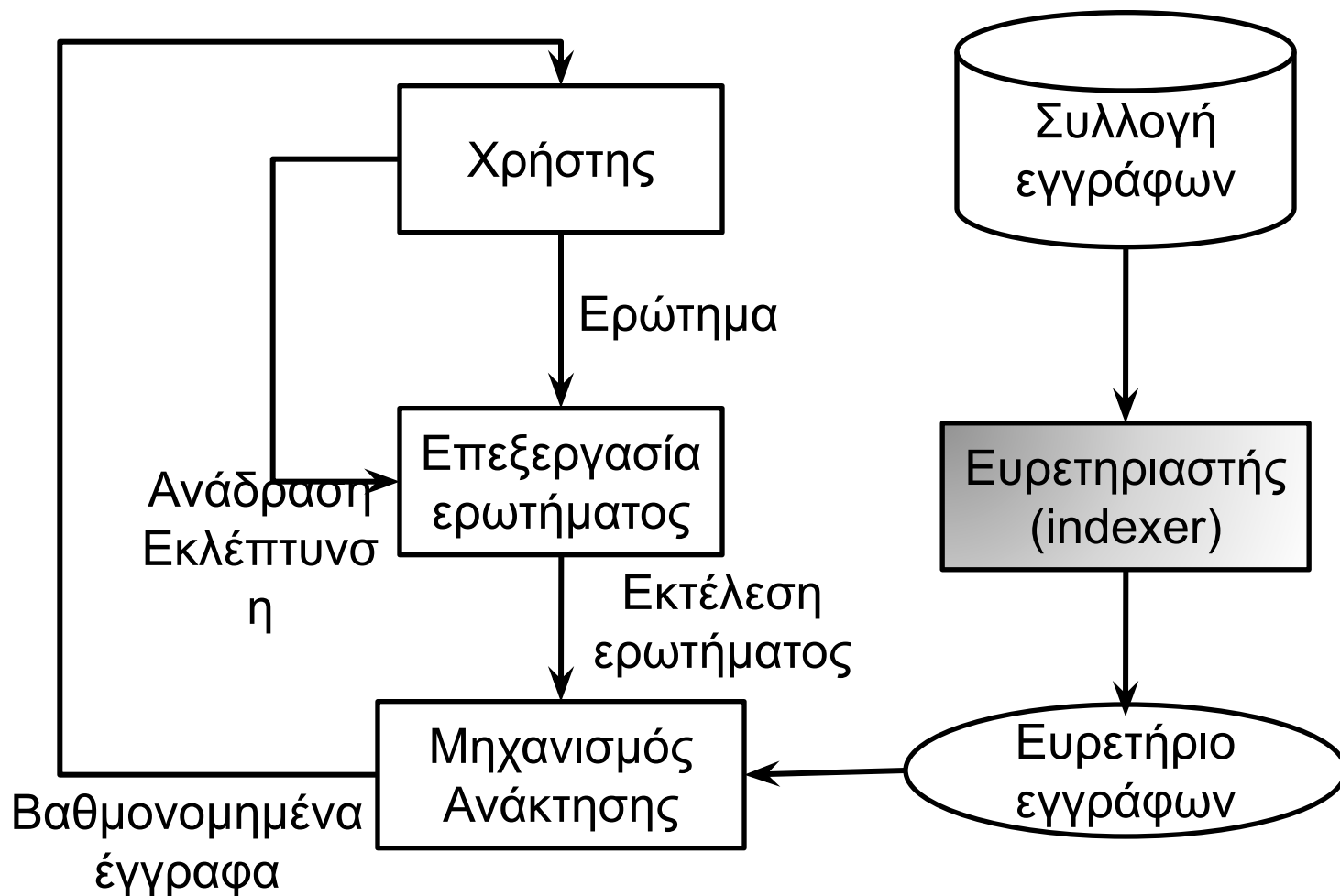
Αρχιτεκτονική ενός συστήματος IR



Επεξεργασία ερωτήματος

- Πολύ απλή: απλά πέρασε το ερώτημα στο μηχανισμό ανάκτησης
 - Προεπεξεργασίας (π.χ. αφαίρεσε τα **stop-words**)
- Πιο σύνθετη
 - Μετέτρεψε ένα ερώτημα φυσικής γλώσσας σε ερώτημα προς το μηχανισμό ανάκτησης
 - Χρησιμοποίησε user feedback για να βελτιώσεις (επεκτείνεις/εκλεπτύνεις) το ερώτημα (**relevance feedback**)
 - κλπ..

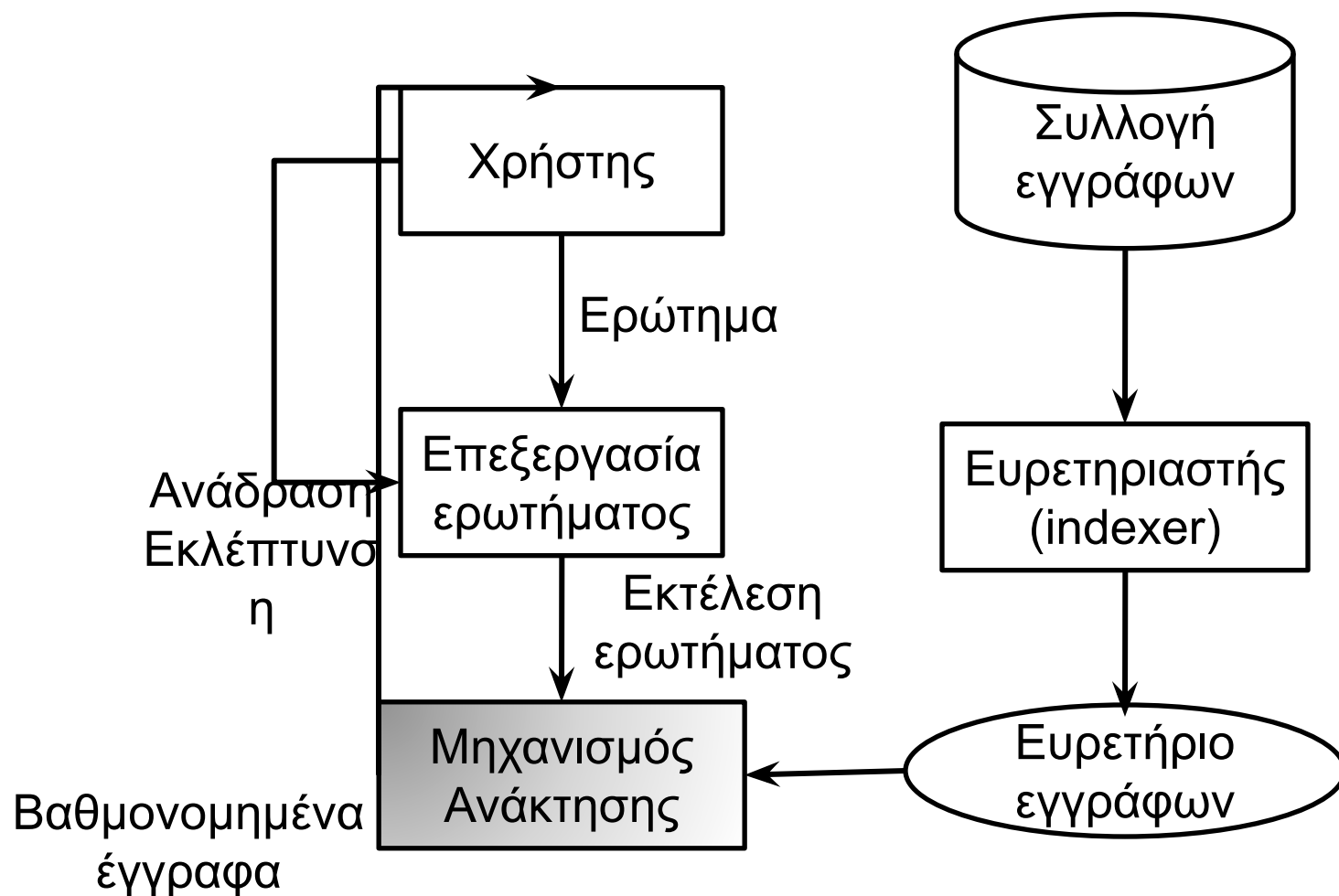
Αρχιτεκτονική ενός συστήματος IR



Μηχανισμός ευρετηρίασης

- Indexer
- Ευρετηριάζει τα αρχικά έγγραφα σε κατάλληλες δομές δεδομένων ώστε να επιτρέπει την αποτελεσματική και γρήγορη ανάκτηση
- Οι δομές αυτές ονομάζονται ευρετήρια εγγράφων (**document index**)
 - Το πιο διαδεδομένο είναι το ανεστραμμένο ευρετήριο: **inverted index**

Αρχιτεκτονική ενός συστήματος IR



Μηχανισμός ανάκτησης

- Υπολογίζει ένα βαθμό σχετικότητας (**relevance score**)
 - Για κάθε έγγραφο
 - Ως προς το ερώτημα
- Οι βαθμοί αυτοί χρησιμοποιούνται για να βαθμονομήσουμε (**rank**) τα έγγραφα πριν τα δείξουμε στο χρήστη
- Συνήθως μόνο ένα μικρό υποσύνολο εγγράφων συγκρίνεται (και βαθμολογείται) ως προς το ερώτημα

Περιεχόμενα

- Βασικές έννοιες
- Μοντέλα IR
- Προεπεξεργασία κειμένων και ιστοσελίδων
- Ανεστραμένα ευρετήρια
- LSI (Latent Semantic Indexing)
- Embeddings
- Μέθοδοι αξιολόγησης του IR
- Αναζήτηση στο Web
- Meta-search

Μοντέλα IR

- Καθορίζουν
 - Την αναπαράσταση ενός εγγράφου ή ενός ερωτήματος
 - Τον τρόπο ορισμού της σχετικότητας μεταξύ ενός εγγράφου και ενός ερωτήματος
- Κυριότερα μοντέλα
 - **Boolean model**
 - **Vector space model**
 - Probabilistic model
 - κλπ

Γενικό πλαίσιο των IR Models

- Κάθε έγγραφο/ερώτημα είναι ένα τσουβάλι λέξεις (“bag” of words)
 - Η διάταξη και η θέση των λέξεων στην πρόταση ή στο έγγραφο αγνοείται
- Κάθε λέξη (όρος-term) σε κάθε έγγραφο συνδέεται με ένα βάρος (weight)
 - Αν ο όρος t_i δεν υπάρχει σε ένα έγγραφο τότε το βάρος του είναι $w_i = 0$

Διανυσματική αναπαράσταση

- **Vector representation**
- Για μια συλλογή εγγράφων D , και το σύνολο $V = \{t_1, t_2, \dots, t_{|V|}\}$ των $|V|$ διακριτών όρων που εμφανίζονται στη συλλογή. Το V είναι το λεξικό (**vocabulary**).
- Ένα βάρος $w_{ij} > 0$ μπαίνει για τον όρο t_i που εμφανίζεται στο έγγραφο $\mathbf{d}_j \in D$.
- Αν ο ίδιος όρος δεν εμφανίζεται στο έγγραφο $\mathbf{d}_j$ τότε $w_{ij} = 0$.
$$\mathbf{d}_j = (w_{1j}, w_{2j}, \dots, w_{|V|j})$$
- Κάθε συλλογή εγγράφων αναπαρίσταται ως πίνακας

Boolean model

- Αναπαράσταση εγγράφου:

- $w_{ij} = 1$, αν το t_i εμφανίζεται στο d_j
- $w_{ij} = 0$, διαφορετικά

- Ερωτήματα

- Οι όροι του ερωτήματος συνδυάζονται λογικά με Boolean τελεστές **AND**, **OR**, **NOT**.
 - π.χ. $((data \text{ AND } mining) \text{ AND } (NOT \text{ text}))$

Ανάκτηση στο Boolean model

- Για ένα Boolean ερώτημα, το σύστημα ανακτά κάθε έγγραφο που κάνει το ερώτημα λογικά αληθές.
- Η μέθοδος καλείται **exact match**.
- Το Boolean model δεν είναι αποδοτικό
 - Τα αποτελέσματα που ανακτώνται δεν είναι καλά γιατί αγνοούνται πληροφορίες όπως η εγγύτητα, η συχνότητα των λέξεων κλπ.
 - Ορισμένες μηχανές αναζήτησης υποστηρίζουν λογικούς τελεστές (**inclusion** ή **exclusion**)
 - http://www.google.com/advanced_search?hl=en

Διανυσματικό μοντέλο

- Vector space model
- Αναπαράσταση εγγράφων
 - Και πάλι αναπαρίστανται ως “bag” of words ή terms
 - Κάθε έγγραφο αντιστοιχεί σε ένα διάνυσμα βαρών
 - Τα βάρη των όρων δεν είναι πλέον 0 ή 1
 - Χρησιμοποιούνται διάφορες παραλλαγές της συχνότητας εμφάνισης ενός όρου στο κείμενο (TF) και της συχνότητας εμφάνισης ενός όρους στα έγγραφα της συλλογής (IDF)

Term Frequency (TF)

- Το βάρος ενός όρου t_i στο έγγραφο $\mathbf{d}_j$ είναι ο αριθμός εμφανίσεων του όρου t_i στο $\mathbf{d}_j$, και συμβολίζεται f_{ij}
- Μπορούμε επίσης να κανονικοποιήσουμε τα βάρη
 - π.χ. διαιρώντας με το πλήθος των όρων του $\mathbf{d}_j$
 - διαιρώντας με το μέγιστο f_{ij} για το εκάστοτε $\mathbf{d}_j$
- Προβλήματα
 - Ορισμένες λέξεις είναι πολύ κοινές και δεν έχουν καμία πληροφοριακή αξία (π.χ. άρθρα, σύνδεσμοι κλπ – stopwords)
 - Ορισμένες λέξεις είναι πολύ κοινές για μια συλλογή (π.χ. η λέξη gene σε μια συλλογή βιολογικών κειμένων).

TF-IDF weighting scheme

- Το πιο δημοφιλές μοντέλο υπολογισμού βαρών

- TF: καθαρή συχνότητα εμφάνισης όρου στο κείμενο (**term frequency**)

$$tf_{ij} = \frac{f_{ij}}{\max\{f_{1j}, f_{2j}, \dots, f_{|V|j}\}}$$

- IDF: αντίστροφη συχνότητα εμφάνισης στα έγγραφα της συλλογής (**inverse document frequency**)

$$idf_i = \log \frac{N}{df_i}$$

N : πλήθος εγγράφων στη συλλογή

df_i : πλήθος εγγράφων που περιέχουν το t_i

- Τελικά το TF-IDF βάρος είναι:

$$w_{ij} = tf_{ij} \times idf_i$$

Ανάκτηση στο vector space model

- Το **ερώτημα** q αναπαρίσταιται με παρόμοιο τρόπο
- Η σχετικότητα (ομοιότητα) του d_j ως προς το q :
- Ομοιότητα συνημιτόνου - Cosine similarity
 - το συνημίτονο της γωνίας μεταξύ των δύο διανυσμάτων (d_j και q)

$$\text{cosine}(\mathbf{d}_j, \mathbf{q}) = \frac{\langle \mathbf{d}_j \bullet \mathbf{q} \rangle}{\|\mathbf{d}_j\| \times \|\mathbf{q}\|} = \frac{\sum_{i=1}^{|V|} w_{ij} \times w_{iq}}{\sqrt{\sum_{i=1}^{|V|} w_{ij}^2} \times \sqrt{\sum_{i=1}^{|V|} w_{iq}^2}}$$

- Το Cosine χρησιμοποιείται και στο text clustering
- Βαθμονόμηση – Ranking: Με βάση τα σκορ ομοιότητας

Παράδειγμα

- Ένα λεξικό με τρεις όρους:
 - $V = \{\text{hardware, software, users}\}$
- Μια συλλογή 9 κειμένων:
 - $A1=(1, 0, 0), \quad A2=(0, 1, 0), \quad A3=(0, 0, 1)$
 - $A4=(1, 1, 0), \quad A5=(1, 0, 1), \quad A6=(0, 1, 1)$
 - $A7=(1, 1, 1) \quad A8=(1, 0, 1). \quad A9=(0, 1, 1)$
- Αν το ερώτημα είναι “hardware και software” ποια κείμενα θα ανακτηθούν;
 - Boolean query matching
 - Vector space matching

Παράδειγμα

- Boolean query matching:
 - Τα έγγραφα A4, A7 θα ανακτηθούν ως τα μόνα που έχουν ΚΑΙ τους δύο όρους
- Similarity matching (cosine):
 - $q=(1, 1, 0)$
 - $S(q, A1)=0.71$
 - $S(q, A2)=0.71$
 - $S(q, A3)=0$
 - $S(q, A4)=1$
 - $S(q, A5)=0.5$
 - $S(q, A6)=0.5$
 - $S(q, A7)=0.82$
 - $S(q, A8)=0.5$
 - $S(q, A9)=0.5$
 - Το βαθμονομημένο σύνολο αποτελεσμάτων θα περιλαμβάνει=
 - $\{A4, A7, A1, A2, A5, A6, A8, A9\}$

Περιεχόμενα

- Βασικές έννοιες
- Μοντέλα IR
- Προεπεξεργασία κειμένων και ιστοσελίδων
- Ανεστραμένα ευρετήρια
- LSI (Latent Semantic Indexing)
- Embeddings
- Μέθοδοι αξιολόγησης του IR
- Αναζήτηση στο Web
- Meta-search

Προεπεξεργασία κειμένου

- Εντοπισμός λέξεων: tokenization
- Εντοπισμός όρων
- Αφαίρεση stopwords
- Αποκοπή ρίζας (stemming)
- Υπολογισμός συχνοτήτων εμφάνισης και TF-IDF βαρών.

Stopwords

- Λέξεις που δεν μεταφέρουν κάποια έννοια, αποτελούν θόρυβο και πρέπει να αφαιρεθούν (“stopped”) πριν ευρετηριαστούν
 - π.χ. : και, το, ο, η, στο, ή, σε, για, κλπ...
- Ο ρόλος τους στο κείμενο είναι καθαρά συντακτικός
- Κρατούνται σε ένα αρνητικό λεξικό
 - 10–1000 λέξεις
 - Π.χ. http://ir.dcs.gla.ac.uk/resources/linguistic_utils/stop_words
- Για ένα συγκεκριμένο πεδίο εφαρμογής μπορεί να χρειαστούμε και μια επιπλέον εξειδικευμένη λίστα stopwords

Αφαίρεση stopwords

- Κατά την ανάγνωση και διάσπαση του κειμένου σε τμήματα (tokenization) τα ανιχνεύουμε και τα αφαιρούμε
- Γιατί τα αφαιρούμε;
 - Μειώνουμε το μέγεθος του ευρετηρίου
 - Τα stopwords αντιστοιχούν στο 20-30% του συνολικού αριθμού εμφανίσεων λέξεων
 - Βελτιώνουμε την απόδοση του συστήματος ανάκτησης
 - Τα stopwords δεν είναι χρήσιμα και μπορεί να μπερδέψουν το σύστημα ανάκτησης

Εύρεση ρίζας - Stemming

- Η βασική δομή μιας λέξης στη Νέα Ελληνική Γλώσσα αποτελείται από τα ακόλουθα τμήματα,
Πρόθεμα + Ρίζα(Stem) + Κατάληξη + Τονισμό
- Σκοπός μας είναι να βρούμε την ρίζα κάθε λέξης
 - ✓ Δηλαδή η ορθή αφαίρεση της κατάληξης κάθε λέξης (*suffix stripping*)
 - ✓ Με τη Χρήση γλωσσικών μορφολογικών κανόνων
- Στόχος είναι να εντοπίσουμε τη ρίζα (**root/stem**) κάθε λέξης. π.χ.,
 - χρήστες
 - χρήστης
 - χρηστικός
 -

απέχω
απέχουμε
αποχή

stem: χρηστ

απεχ

Βασικές μέθοδοι stemming

Μια σειρά από κανόνες: π.χ. για αγγλικά

- Αφαίρεση κατάληξης

- if a word ends with a consonant other than s, followed by an s, then delete s.
- if a word ends in es, drop the s.
- if a word ends in ing, delete the ing unless the remaining word consists only of one letter or of th.
- If a word ends with ed, preceded by a consonant, delete the ed unless this leaves only a single letter.

- Μετασχηματισμός λέξης

- if a word ends with “ies” but not “eies” or “aies” then “ies --> y.”

Μορφολογικοί Κανόνες

- Στη Νέα Ελληνική Γλώσσα συναντάμε: Ρήματα, Επίθετα, Ουσιαστικά, Επιρρήματα, Πρόθεσεις, Σύνδεσμοι, Επιφωνήματα κλπ.
- Το μεγαλύτερο πλήθος των λέξεων υπάγεται στα ρήματα – επίθετα – ουσιαστικά

Ρήματα	Επίπεδο 1	ώ/ω, ά, έ, ί, ύ, ή, εί, αί, άω
	Επίπεδο 2	β-, γ-, ζ-, φ-, χ-, θ-, κ-, λ-αίν-, (ακολουθεί μια κατάληξη από το επίπεδο 1) -νω, -ήθηκα, -ύθηκα, -αίνομαι, -ημένος, -άνθηκα, -άχτηκα, -έρνω, -άνων, -αίω, -αίξα -αίχτηκα, -μένος, -όμαι, -άμαι, -ούμαι, -ιέμαι
Ουσιαστικά	Επίπεδο 1	ός/ος, άς/ας, ές/ες, ής/ης, ούς/ους, ως, οί/οι, ά/α, έ/ε, ή/η, ό/ο, ί/ι
	Επίπεδο 2	ιμ, ουδ, ατ, ηδ, εδ, αδ
Επίθετα	Επίπεδο 1	ος, ης, ούι άς, ων, ά/α, έ/ε, ή/η, ό/ο, ί/ι, υ/ύ
	Επίπεδο 2	ικ, άδικ, ούδικ, ότερ, ότατ, ώτερ, ώτατ, ύτερ, ύτατ, έστερ, έστατ, ούς/ους, ονας, ούλ, άσι, ιάρ, άρ, υμέν, υσμέν, αγμέν

Οφέλη από το Stemming

- Βελτίωση της αποτελεσματικότητας του IR
 - Μειώνεται ο χώρος λέξεων (μέγεθος λεξικού) και αυξάνεται η πιθανότητα ταιριάσματος λέξεων
 - Αυξάνει την ανάκληση (recall)
- Μείωση του μεγέθους του ευρετηρίου
 - Μπορεί να οδηγήσει σε μείωση 40-50%.
- Μείωση την ακρίβειας: πολλά κείμενα που δεν περιέχουν τη λέξη που θέλουμε, αλλά μια διαφορετική λέξη που όμως δίνει το ίδιο stem επιστρέφονται.

Άλλες ενέργειες προεπεξεργασίας

● Αριθμοί

- Στο IR: τους αφαιρούμε εκτός από πολύ συγκεκριμένους τύπους (χρηματικά ποσά, ημερομηνίες)
- Στο Web: τους ευρετηριάζουμε

● Hyphens

- Κάνουμε διάσπαση των λέξεων;
 - π.χ. «προ-επεξεργασία»
- Συνήθως έχουμε ένα γενικό κανόνα και συγκεκριμένες εξαιρέσεις

● Punctuation Marks

- Το ίδιο με τα hyphens

● Κεφαλαία/Μικρά

- Συνήθως τα μετατρέπουμε όλα σε μια από τις δύο μορφές

Προεπεξεργασία ιστοσελίδων

- **Εντοπισμός διαφορετικών πεδίων**
 - π.χ. HTML: title, metadata, body
- **Εντοπισμός του anchor text ενός link**
 - Συχνά αυτό που λένε οι άλλοι για μια σελίδα είναι πιο περιγραφικό από αυτό που λέει η ίδια η σελίδα
- **Αφαίρεση των HTML tags**
 - Όμοια με το punctuation
- **Εντοπισμός των μπλοκ περιεχομένου**
 - Τμηματοποίηση οπτική με βάση τη δομή
 - Θέλει εκπαίδευση και χειροκίνητη επισήμανση
 - Ταίριασμα υποδέντρων
 - Πολλές ιστοσελίδες βασίζονται σε templates
 - Βρίσκει τα κρυμμένα templates

Υπολογισμός του TF-IDF

- TF: Υπολογίζουμε τον αριθμό εμφανίσεων ενός όρου στο έγγραφο
 - Η συχνότητα εμφάνισης είναι ανάλογη της σημαντικότητας του όρου για το έγγραφο
 - Αν ο όρος εμφανίζεται συχνά στο έγγραφο, τότε το έγγραφο μάλλον ασχολείται με θέματα σχετικά με τον όρο
- IDF: Μετρά τον αριθμό των εγγράφων της συλλογής που περιέχουν τον κάθε όρο
 - Προϋποθέτει ότι έχουμε στη διάθεσή μας όλη τη συλλογή από την αρχή
 - Αυτό δεν ισχύει στο web

Το Web Search ως ένα σύστημα IR

- Ένας Web crawler περιηγείται το Web και συλλέγει τις σελίδες που βρίσκει
- Τα αποτελέσματα αποθηκεύονται σε δομές γρήγορης αναζήτησης που λέγονται **ανεστραμμένα ευρετήρια**
- Την ώρα που υποβάλλεται ένα ερώτημα η μηχανή αναζήτησης κάνει μια σειρά από συγκρίσεις διανυσμάτων
- Οι σελίδες βαθμονομούνται με βάση
 - Την ομοιότητα περιεχομένου προς το ερώτημα (Similarity to query)
 - Τη δημοτικότητα τους Popularity (“Authority”)

Περιεχόμενα

- Βασικές έννοιες
- Μοντέλα IR
- Προεπεξεργασία κειμένων και ιστοσελίδων
- **Ανεστραμένα ευρετήρια**
- LSI (Latent Semantic Indexing)
- Embeddings
- Μέθοδοι αξιολόγησης του IR
- Αναζήτηση στο Web
- Meta-search

Αποθήκευση εγγράφων

- Απλή προσέγγιση: Αποθηκεύω κάθε έγγραφο ως μια λίστα όρων/λέξεων
 - Προβληματική προσέγγιση για το Web
- Ανεστραμμένα ευρετήρια
 - Για μια συλλογή εγγράφων διατηρούν για κάθε διαφορετικό όρο μια λίστα από όλα τα έγγραφα που τον περιέχουν
 - Κατά την ανάκτηση, ο χρόνος για τον εντοπισμό των εγγράφων που περιέχουν έναν όρο είναι σταθερός
 - Όμοια μπορούμε να χειριστούμε ερωτήματα πολλαπλών όρων

Ανεστραμμένο ευρετήριο

- Για ένα:
 - Σύνολο εγγράφων $D = \{d_1, d_2, \dots, d_N\}$
 - Κάθε έγγραφο έχει μοναδικό ID id_j
- Ένα ανεστραμμένο ευρετήριο αποτελείται από:
 - Ένα λεξιλόγιο V που περιέχει όλους τους διακριτούς όρους t_i στο D
 - Και μια ανεστραμμένη λίστα με αναφορές(postings) για κάθε t_i
- Κάθε αναφορά περιέχει το id_j του d_j και επιπλέον πληροφορίες, π.χ. συχνότητα, θέση του όρου στο έγγραφο κλπ

Παράδειγμα

- Έχουμε 3 έγγραφα id1, id2, id3:

- id1: Web mining is useful.

1 2 3 4

- id2: Usage mining applications.

1 2 3

- id3: Web structure mining studies the Web hyperlink structure.

1 2 3 4 5 6 7 8

Applications: id2

Hyperlink: id3

Mining: id1,id2,id3

Structure: id3

Studies: id3

Usage: id2

Useful: id1

Web: id1,id3

Απλό
ανεστραμμένο
ευρετήριο

Applications: <id2,1,[3]>

Hyperlink: <id3,1,[7]>

Mining: <id1,1,[2]>, <id2,1,[2]>, <id3,1,[3]>

Structure: <id3,2,[2,8]>

Studies: <id3,1,[4]>

Usage: <id2,1,[1]>

Useful: <id1,1,[4]>

Web: <id1,1,[1]>, <id3,2,[1,6]>

Σύνθετο
ανεστραμμένο
ευρετήριο

Αναζήτηση με ανεστραμμένο ευρετήριο

- Για το σύνολο των όρων ενός ερωτήματος:
 1. **Αναζήτηση στο λεξιλόγιο:** βρίσκω για κάθε όρο του ερωτήματος την λίστα αναφορών στο λεξιλόγιο.
 - Το λεξιλόγιο συνήθως χωρά στη μνήμη
 - Αν έχω κάποιο ευρετήριο επιταχύνω την αναζήτηση
 - Αν έχω λεξικογραφική διάταξη κάνω δυαδική αναζήτηση
 - Αν το ερώτημα έχει 1 όρο, πάω στο βήμα 3, αλλιώς πάω στο βήμα 2
 2. **Συγχώνευση αποτελεσμάτων:** συγχωνεύω τις λίστες κάθε όρου και κρατώ την τομή τους
 - Έτσι βρίσκω τα έγγραφα που περιέχουν όλους τους όρους του ερωτήματος
 - Ευριστικό: Ξεκινώ από τη μικρότερη λίστα
 - Ενδεχομένως να δεχτώ μερικό ταίριασμα στους όρους
 3. **Υπολογισμός βαθμού σχετικότητας:** για κάθε έγγραφο που περιέχει τον όρο (τους όρους) ως προς το ερώτημα
 - Χρήση συνάρτησης ομοιότητας (π.χ. cosine)
 - Μπορεί να λάβει υπόψη την εγγύτητα των όρων στο έγγραφο και το ερώτημα αντίστοιχα

Παράδειγμα

id1: Web mining is useful.

1 2 3 4

id2: Usage mining applications.

1 2 3

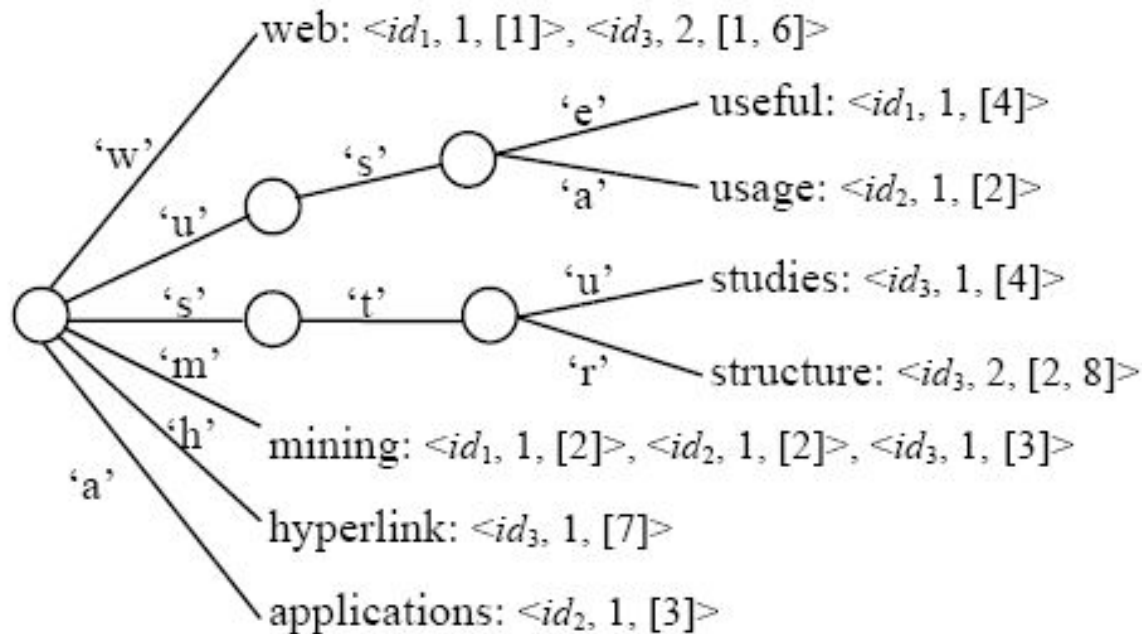
id3: Web structure mining studies the Web hyperlink structure.

1 2 3 4 5 6 7 8

- Ερώτημα: “web mining”
- Βήμα 1: Mining: $\langle id_1, 1, [2] \rangle$, $\langle id_2, 1, [2] \rangle$, $\langle id_3, 1, [3] \rangle$
Web: $\langle id_1, 1, [1] \rangle$, $\langle id_3, 2, [1,6] \rangle$
- Βήμα 2: Τα id_1, id_3 περιέχουν και τις δύο λέξεις
- Βήμα 3: (διαισθητικά) το id_1 πρέπει να λάβει μεγαλύτερη βαθμολογία από το id_3
 - Γιατί;
 - Με ποιο άλλο τρόπο μπορούμε να υπολογίσουμε τη βαθμολογία;

Δημιουργία ευρετηρίου

- Ειδική δομή (trie structure):
- id_1 : Web mining is useful
- id_2 : Usage mining applications
- id_3 : Web structure mining studies the Web hyperlink structure



Ευρετήρια μηχανών αναζήτησης

- Το ευρετήριο του Web δεν χωρά στη μνήμη
- Δημιουργούμε μερικά ευρετήρια που χωρούν στη μνήμη και τα αποθηκεύουμε στο δίσκο
- Μόλις φτιάξουμε όλα τα ευρετήρια τα συγχωνεύουμε ιεραρχικά
- Εναλλακτικά χρησιμοποιούμε ένα hash-table (ή κάποια άλλη δομή) ή συγχωνεύουμε τα ευρετήρια αυξητικά
- Συγχωνεύουμε τα ευρετήρια

Περιεχόμενα

- Βασικές έννοιες
- Μοντέλα IR
- Προεπεξεργασία κειμένων και ιστοσελίδων
- Ανεστραμένα ευρετήρια
- LSI (Latent Semantic Indexing)
- Embeddings
- Μέθοδοι αξιολόγησης του IR
- Αναζήτηση στο Web
- Meta-search

Λέξεις ή Έννοιες

- Το ακριβές λεκτικό ταίριασμα (exact keyword/term matching) δεν δουλεύει πάντοτε καλά
 - Κυρίως όταν είναι μικρά τα κείμενα ή τα ερωτήματα
 - Οι άνθρωποι χρησιμοποιούν διαφορετικούς όρους για να περιγράψουν τα ίδια πράγματα (τις ίδιες έννοιες) π.χ. τρένο, αμαξοστοιχία
- Μια λύση είναι να έχουμε ένα θησαυρό που θα μας λέει ποιες λέξεις συνδέονται σημασιολογικά με ποιες άλλες λέξεις (π.χ. ότι τρένο και αμαξοστοιχία είναι συνώνυμα, ή ότι αμάξι και αυτοκίνητο είναι συνώνυμα και ότι το όχημα είναι υπέρνυμο του αυτοκινήτου)

Latent Semantic Indexing (LSI)

- Η μέθοδος ευρετηρίασης της υποκρυπτόμενης σημασιολογικής πληροφορίας (indexing of latent semantics - LSI) χρησιμοποιεί τις στατιστικές συσχετίσεις μεταξύ λέξεων
- Σε πολλά κείμενα που μιλάνε για τρένα εμφανίζεται η λέξη αμαξοστοιχία, άρα οι δύο λέξεις έχουν μια συσχέτιση
- Το LSI χρησιμοποιεί μια στατιστική τεχνική, το **Singular Value Decomposition (SVD)** to για να υπολογίσει την κρυμμένη (λανθάνουσα/latent) σημασιολογική δομή των λέξεων
- **Κρυμμένος χώρος «εννοιών»:** Συσχετίζει συντακτικά διαφορετικούς αλλά σημασιολογικά όμοιους όρους και έγγραφα

Παράδειγμα

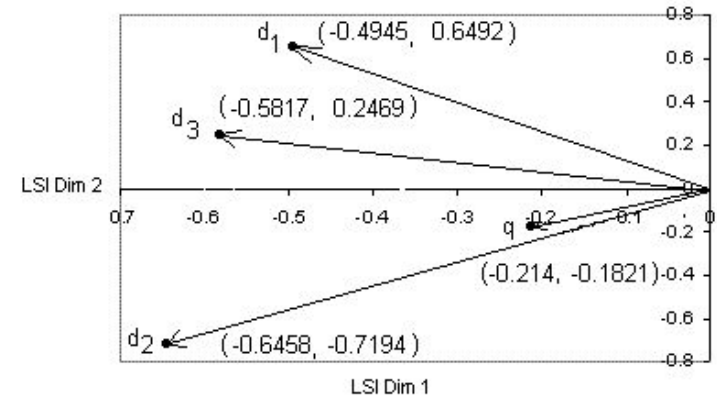
- Βαθμονόμησε τα έγγραφα d1,d2,d3 ως προς το ερώτημα q (gold silver truck)

Terms ↓	d1 ↓	d2 ↓	d3 ↓	q ↓
a	1	1	1	0
arrived	0	1	1	0
damaged	1	0	0	0
delivery	0	1	0	0
fire	1	0	0	0
gold	1	0	1	1
in	1	1	1	0
of	1	1	1	0
shipment	1	0	1	0
silver	0	2	0	1
truck	0	1	1	1

$A =$

$q =$

Αποσύνθεσε τον πίνακα A
και βρες τους πίνακες U, S
και V, ώστε $A = USV^T$



- Δουλεύει καλύτερα από τους παραδοσιακούς αλγορίθμους που βασίζονται σε λέξεις
- Αλλά έχει μεγάλη πολυπλοκότητα (κακό για το Web)

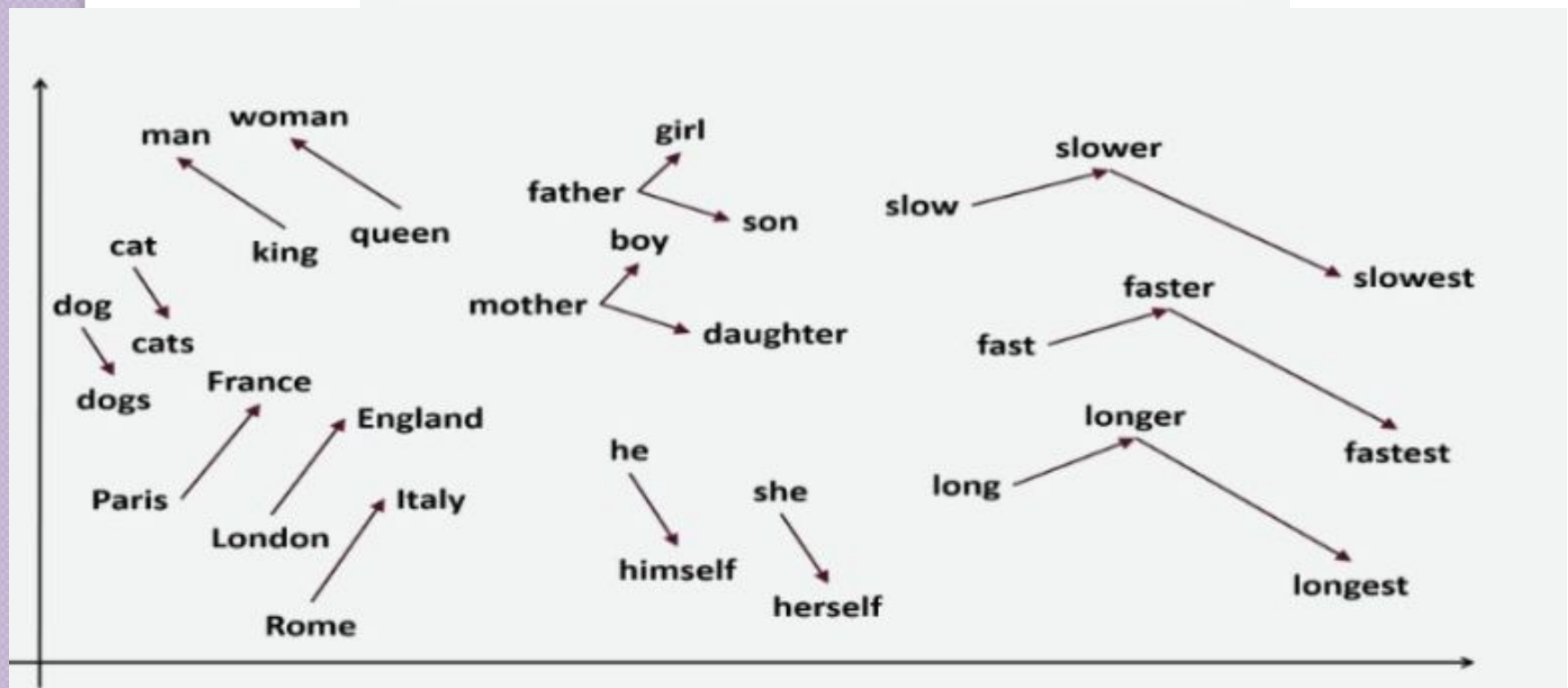
Περιεχόμενα

- Βασικές έννοιες
- Μοντέλα IR
- Προεπεξεργασία κειμένων και ιστοσελίδων
- Ανεστραμένα ευρετήρια
- LSI (Latent Semantic Indexing)
- **Embeddings**
- Μέθοδοι αξιολόγησης του IR
- Αναζήτηση στο Web
- Meta-search

Word2Vec

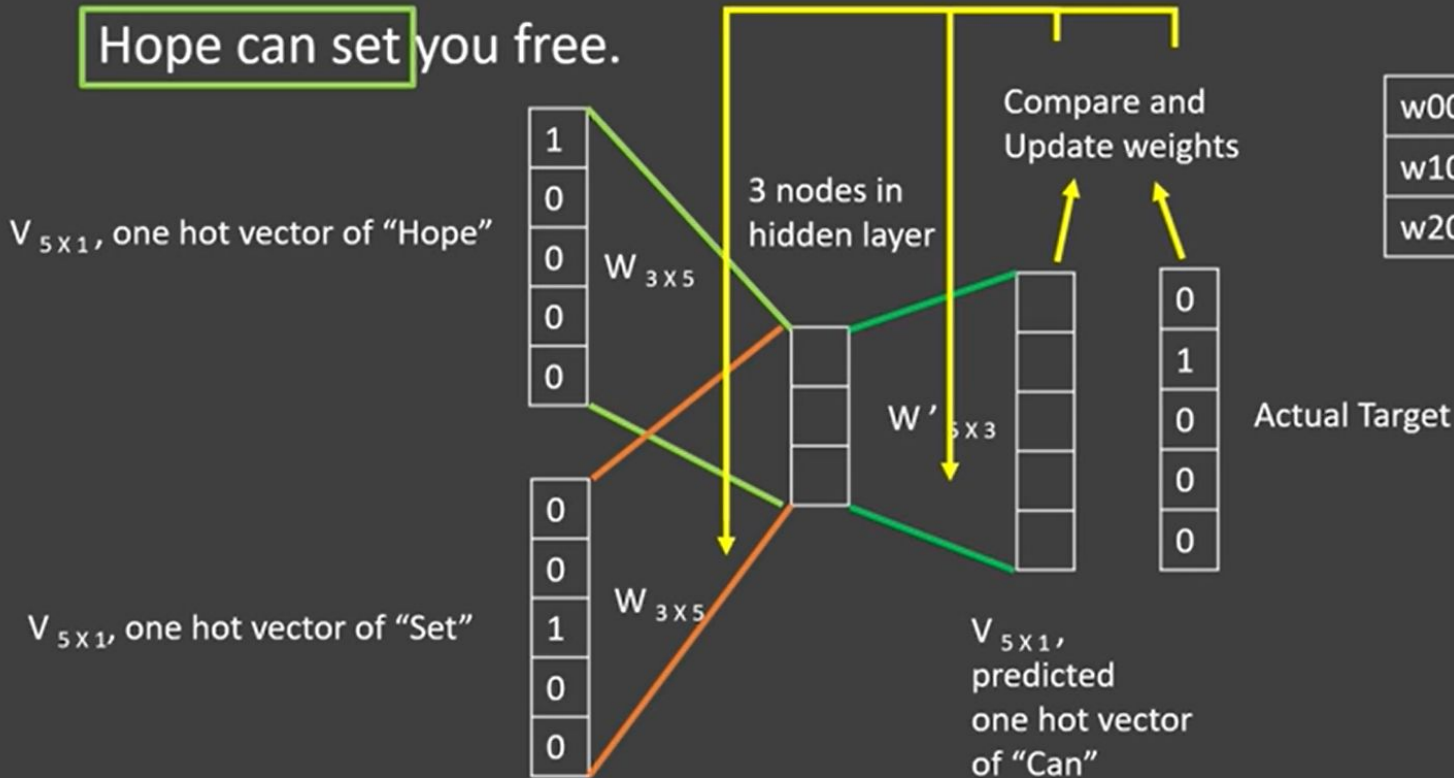
- Προτάθηκαν από τον Tomas Mikolov της Google το 2013
- Βασίζονται στην αναπαράσταση κάθε λέξης με τη μορφή ενός διανύσματος.
- Η διανυσματική αναπαράσταση προκύπτει από το περιβάλλον της λέξης στο σώμα κειμένων.
- Ένα απλό νευρωνικό εκπαιδεύεται με τις λέξεις και το περιβάλλον τους σε ένα μεγάλο σώμα κειμένων.

One hot vectors - Word embeddings



CBOW (continuous bag of words)

Hope can set you free.

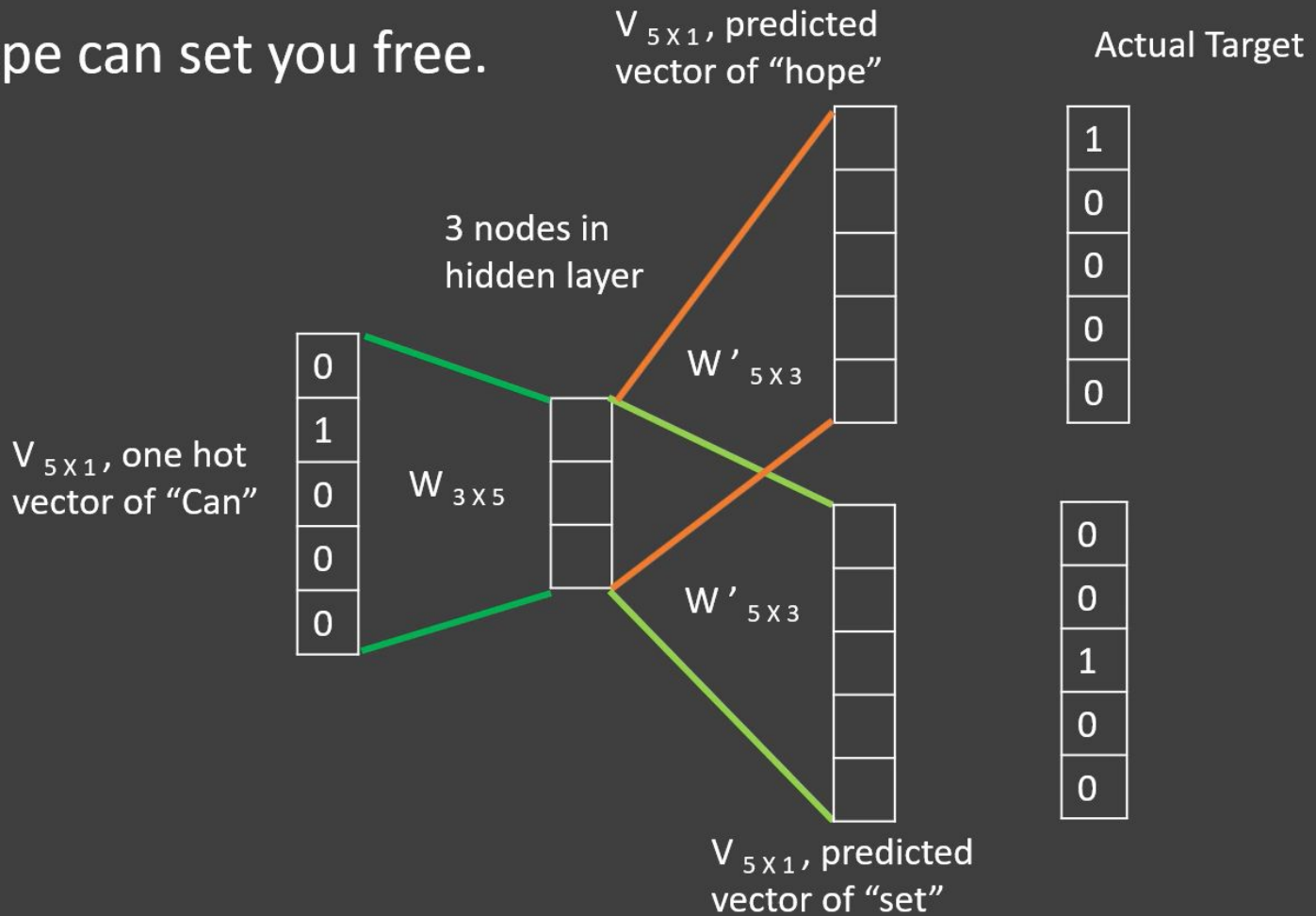


w00	w01	w02	w03	w04
w10	w11	w12	w13	w14
w20	w21	w22	w23	w24

www.youtube.com/thesemicolon

Skip-grams

Hope can set you free.



www.youtube.com/thesemicolon

Παραγωγή vector embeddings

Weights after training

$W_{3 \times 5}$

w00	w01	w02	w03	w04
w10	w11	w12	w13	w14
w20	w21	w22	w23	w24

One Hot vector of words

$V_{5 \times 1}$

1	0	0	0	0
0	1	0	0	0
0	0	1	0	0
0	0	0	1	0
0	0	0	0	1
Hope	can	set	you	free

Word Vector for hope = $W_{3 \times 5} \times V_{5 \times 1}$

w00	w01	w02	w03	w04
w10	w11	w12	w13	w14
w20	w21	w22	w23	w24

$\times$

1
0
0
0
0

=

$V_{3 \times 1}$

w00
w10
w20

Word Vector for Hope

www.youtube.com/thesemicolon

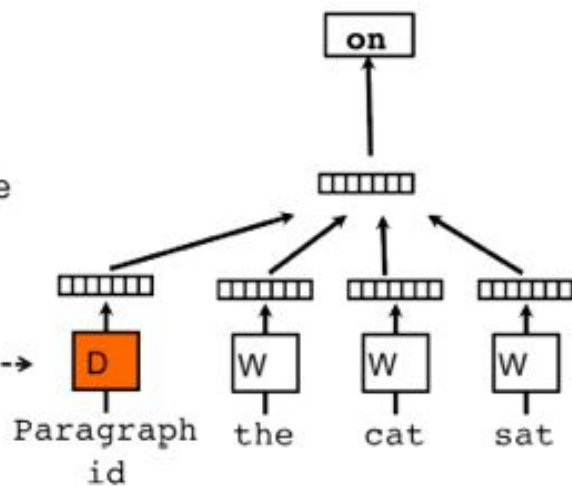
Doc2Vec

- Προτάθηκε από τους Le & Mikolov το 2014
- Βασίζεται στο Word2Vec στο οποίο προσθέτει 1 ακόμη vector (με το id του κειμένου)

Classifier

Average/Concatenate

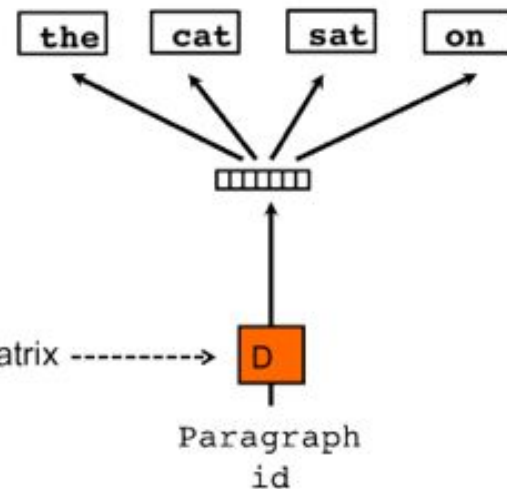
Paragraph Matrix



DM (Distributed Memory)

Classifier

Paragraph Matrix



DBOW (Distributed Bag of Words)

Περιεχόμενα

- Βασικές έννοιες
- Μοντέλα IR
- Προεπεξεργασία κειμένων και ιστοσελίδων
- Ανεστραμένα ευρετήρια
- LSI (Latent Semantic Indexing)
- Embeddings
- Μέθοδοι αξιολόγησης του IR
- Αναζήτηση στο Web
- Meta-search

Μέτρα αξιολόγησης του IR

- Αξιολόγηση των βαθμονομημένων αποτελεσμάτων
- Εξετάζουμε τα έγγραφα που ανακτήθηκαν για ένα ερώτημα
 - Είναι όλα τα ανακτημένα έγγραφα σχετικά με το ερώτημα; **Ακρίβεια - Precision**
 - Έχουν ανακτηθεί όλα τα σχετικά με το ερώτημα έγγραφα που υπάρχουν στη συλλογή;
Πληρότητα/Ανάκληση - Recall

Ορισμοί

- Για

- ένα ερώτημα q
- Μια βαθμονομημένη λίστα εγγράφων
 $R_q: \langle d_1, d_2, \dots, d_N \rangle$
- Και το σύνολο των πραγματικά σχετικών εγγράφων με το q να είναι το D_q

- Ορίζουμε

- Precision: Ποσοστό των ανακτημένων εγγράφων που είναι σχετικά

$$\text{Precision} = |\text{σχετικά έγγραφο στο } R_q| / |R_q|$$

- Recall: Ποσοστό των σχετικών εγγράφων που ανακτήθηκαν

$$\text{Recall} = |\text{σχετικά έγγραφα στο } R_q| / |D_q|$$

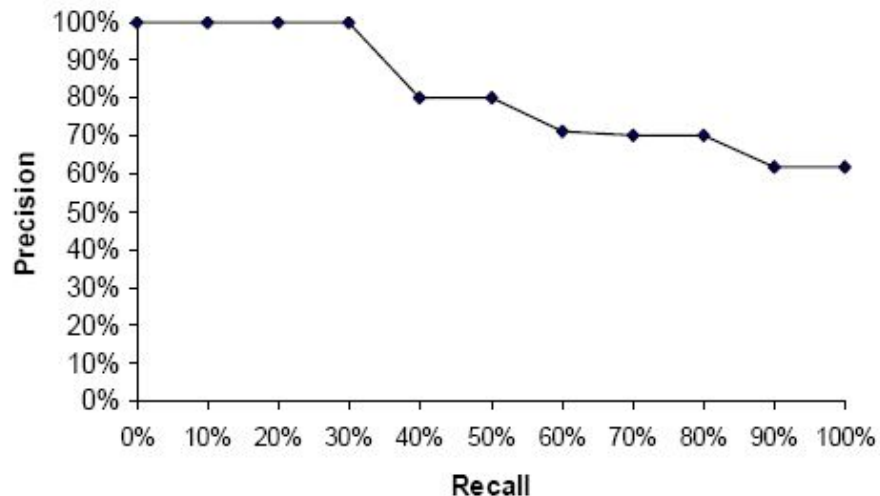
Ακρίβεια της βαθμονόμησης

- Rank precision
- Υπολογίζουμε τις τιμές της ακρίβειας (precision) σε επιλεγμένα μεγέθη της λίστας ανακτημένων εγγράφων
 - π.χ. Για μια μηχανή αναζήτησης, υπολογίζουμε ακρίβεια στις top 5, 10, 15, 20, 25 και 30 απαντήσεις
 - συνήθως ο χρήστης κοιτάζει το πολύ τις 30 πρώτες απαντήσεις
- Το Recall δεν είναι πολύ σημαντικό στο Web search
 - Συνήθως υπάρχουν πάρα πολλά έγγραφα που σχετίζονται με το data mining precision recall
 - Μας ενδιαφέρει κυρίως στα πρώτα έγγραφα που θα φέρουμε στο χρήστη να έχουμε όσο το δυνατόν περισσότερα σχετικά και όχι να φέρουμε όλα τα σχετικά έγγραφα που υπάρχουν στο web

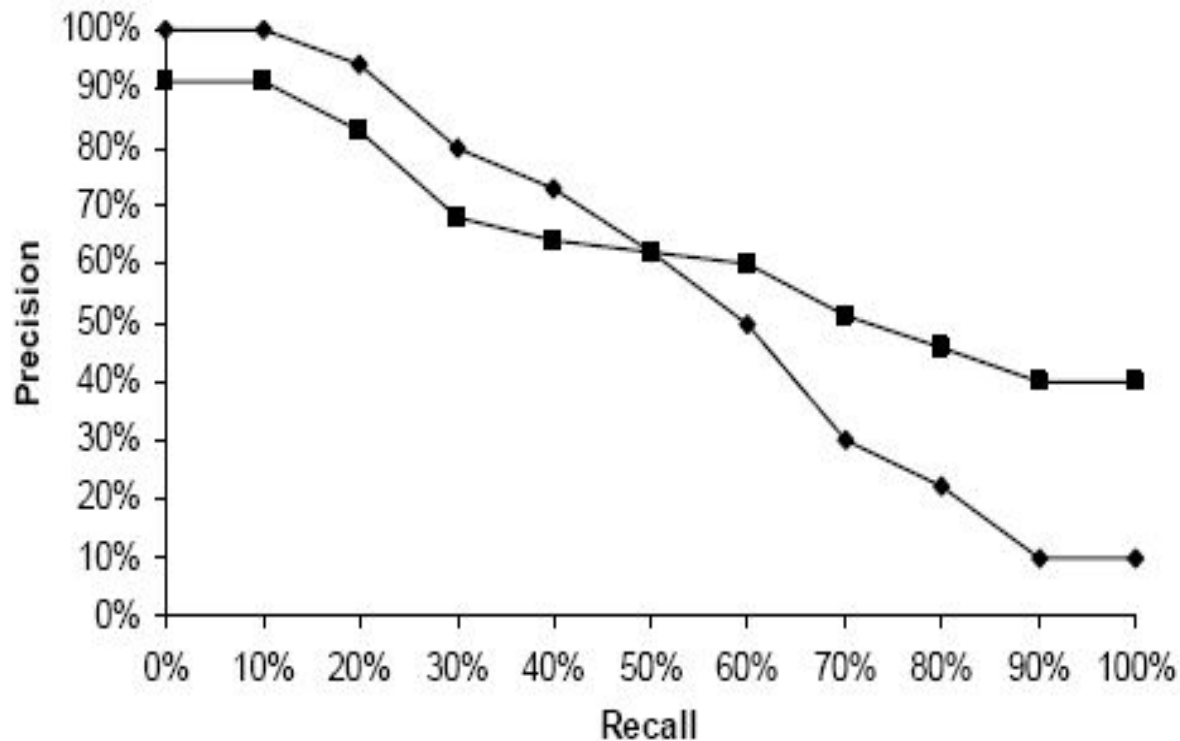
Καμπύλη precision-recall

- Όσο αυξάνει το recall, μειώνεται το precision
- Εμφανίζουμε τις τιμές του precision όχι για τα top-5, 10 κλπ (top- k_i) αποτελέσματα, αλλά για εκείνα τα k_i , στα οποία το recall είναι στο 10,20,...100%
- Εμφανίζουμε τα interpolated precisions στα 11 recall levels
- Στόχος μας να κρατάμε το precision ψηλό για όσο το δυνατόν μεγαλύτερα recall levels

i	$p(r_i)$	r_i
0	100%	0%
1	100%	10%
2	100%	20%
3	100%	30%
4	80%	40%
5	80%	50%
6	71%	60%
7	70%	70%
8	70%	80%
9	62%	90%
10	62%	100%



Ποια μηχανή είναι καλύτερη;





Ανάλυση συνδέσμων

Link Analysis

Περιεχόμενα

- Εισαγωγή
- Ανάλυση κοινωνικών δικτύων
- Βαθμονόμηση
 - HITS
 - PageRank
- Ομοιότητα
 - Συν-αναφορά (Co-citation) και βιβλιογραφική σύζευξη (coupling)
- Web Spamming

Εισαγωγή

- Οι πρώτες μηχανές αναζήτησης συνέκριναν την ομοιότητα του ερωτήματος με το περιεχόμενο των ευρετηριασμένων σελίδων
 - Χρησιμοποιούσαν τεχνικές IR: [cosine](#), [TF-IDF](#), ...
 - Το 1996 διαπιστώθηκε ότι η ομοιότητα περιεχομένου δεν αρκεί πλέον
 - Ο αριθμός των σελίδων που σχετίζονται με μια ερώτηση είναι τεράστιος
- 
- The screenshot shows the Google search interface. The search bar contains the text 'web mining'. To the right of the search bar is a 'Search' button. Below the search bar, it says 'About 13,700,000 results (0.14 seconds)'. There is also a link for 'Advanced search'.
- Πώς μπορούμε να οιαλεζουμε μονο 30-40 σελιδες, να τις βαθμονομήσουμε κατάλληλα και να τις δείξουμε στο χρήστη;
 - Η ομοιότητα περιεχομένου μπορεί εύκολα να εξαπατηθεί ([spam](#))
 - Ο συγγραφέας μιας ιστοσελίδας μπορεί να επαναλάβει πολλές φορές ορισμένες λέξεις κλειδιά ώστε να προωθήσει σημαντικά τη σελίδα του στις βαθμολογίες για πολλά ερωτήματα

Εισαγωγή

- Η έρευνα στράφηκε στους υπερσυνδέσμους (**hyperlinks**),
- Οι ιστοσελίδες συνδέονται με υπερσυνδέσμους που μεταφέρουν σημαντικές πληροφορίες
 - π.χ. το κείμενο που βλέπει ο χρήστης πάνω και γύρω από κάθε υπερσύνδεσμο
 - Ορισμένα hyperlinks οργανώνουν την πληροφορία σε ένα website (navigation links)
 - Άλλα hyperlinks δείχνουν σε σελίδες σε άλλα Web sites
 - Αυτά τα out-going hyperlinks υποδεικνύουν έμμεσα μια σημαντικότητα (**authority**) της σελίδας στην οποία δείχνουν
- Οι σελίδες που δείχνονται από πολλές άλλες σελίδες αποτελούν τις αυθεντίες (authoritaties) του web

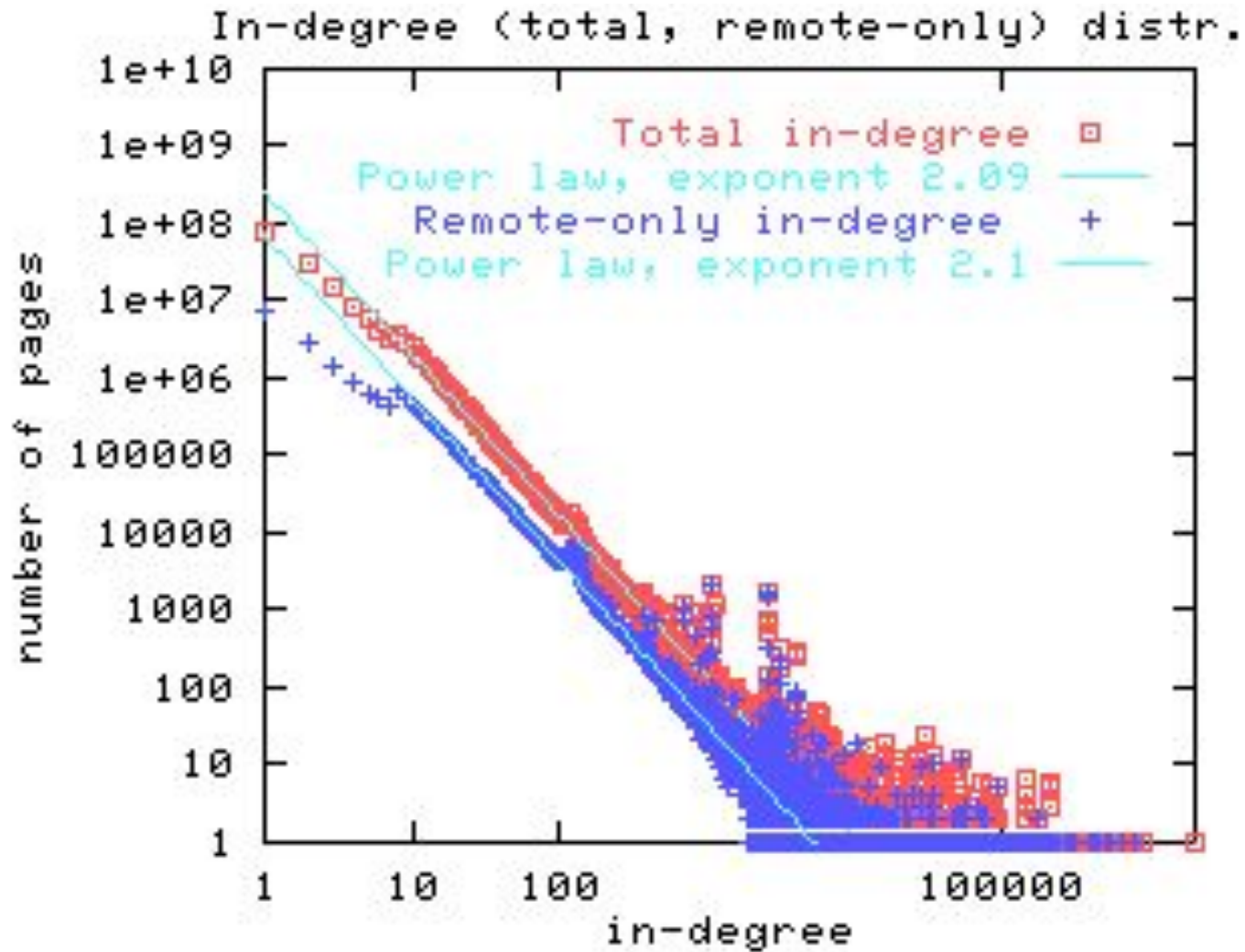
Εισαγωγή

- Το 1997-98 εμφανίστηκαν οι δύο πιο γνωστοί αλγόριθμοι για ανάλυση συνδέσμων: ο **PageRank** και ο **HITS**
- Και οι δύο σχετίζονται με τη θεωρία της **ανάλυσης κοινωνικών δικτύων**. Χρησιμοποιούν τους υπερσυνδέσμους για να βαθμονομήσουν τις σελίδες ως προς την αναγνώριση (“prestige”) ή την αυθεντία τους (“authority”).
 - **HITS**: Jon Kleinberg (Cornel University), at *Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*, January 1998
 - **PageRank**: Sergey Brin and Larry Page (PhD students from Stanford University), at *Seventh International World Wide Web Conference (WWW7)* in April, 1998.

Το web ως γράφος

- Σελίδες= κόμβοι, Σύνδεσμοι= ακμές
 - Αγνοούμε το περιεχόμενο
 - Κατευθυνόμενος γράφος
- Υψηλή συνδεσιμότητα
 - 8-10 σύνδεσμοι/σελίδα κατά μέσο όρο
 - Κατανομή power-law degree

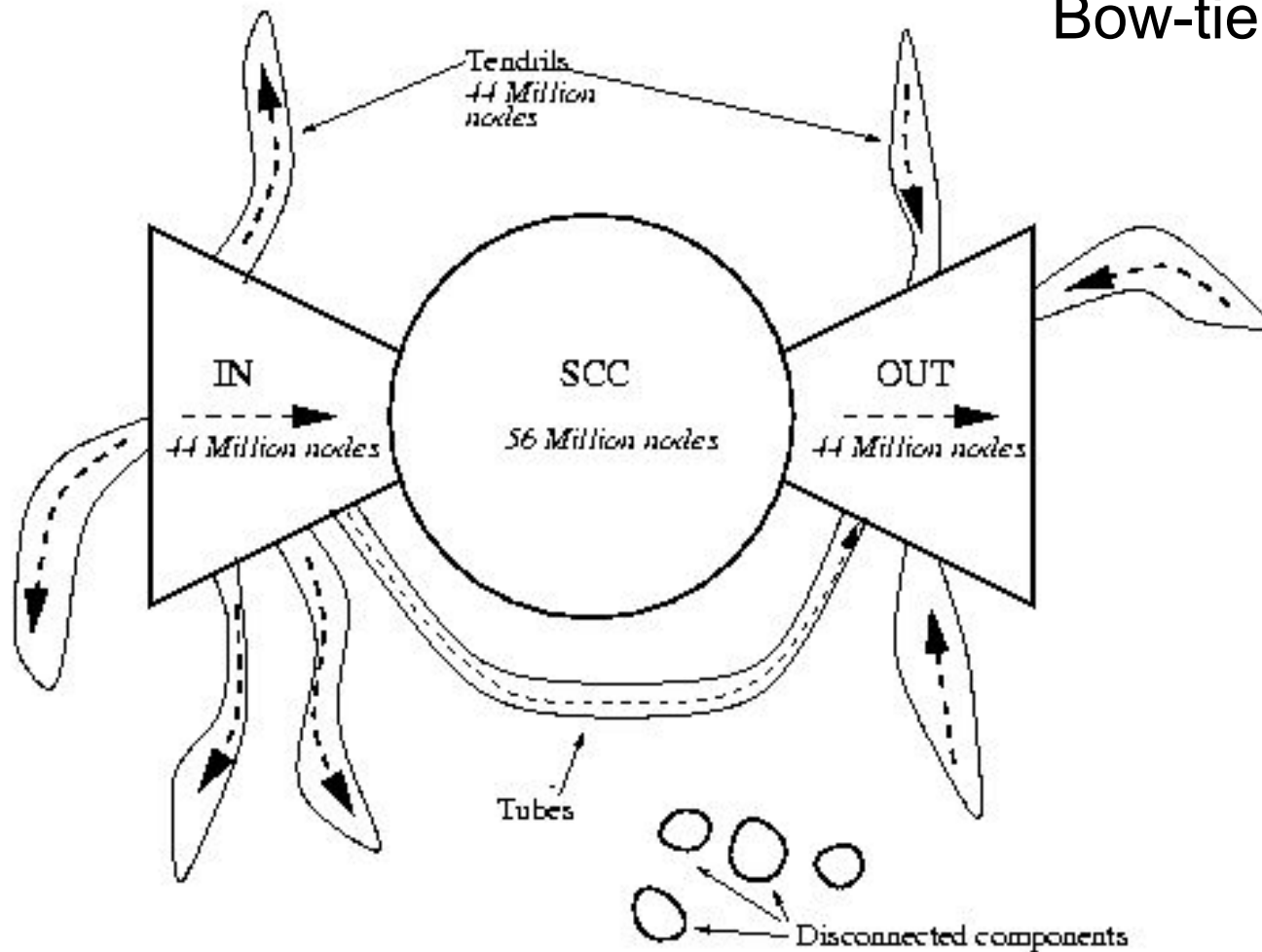
Power-law degree distribution



Source: Broder et al, 2000

Η δομή του Web

Bow-tie Structure



Source: Broder et al, 2000

Ανάλυση υπερσυνδέσμων

- Επιτρέπει να εντοπίσουμε:
 - Αυθεντίες (authorities): Σελίδες με πολλά inlinks
 - Πηγές (hubs): Σελίδες με πολλά outlinks
- Επιτρέπει να βαθμονομήσουμε τις σελίδες (ή τους δικτυακούς τόπους) ως προς τη σημαντικότητά τους
- Επιτρέπει να βρούμε κλίκες σελίδων (**Web communities**)
 - Μια Web community είναι μια ομάδα πυκνά συνδεδεμένων σελίδων που εκπροσωπεί ομάδες ανθρώπων με κοινά ενδιαφέροντα
- Οι εφαρμογές κοινωνικής δικτύωσης έδωσαν πληθώρα από “social” links

Περιεχόμενα

- Εισαγωγή
- **Ανάλυση κοινωνικών δικτύων**
- Βαθμονόμηση
 - HITS
 - PageRank
- Ομοιότητα
 - Συν-αναφορά (Co-citation) και βιβλιογραφική σύζευξη (coupling)
- Web Spamming

Ανάλυση κοινωνικών δικτύων

- Η ανάλυση πραγματικών κοινωνικών δικτύων αναφέρεται στη μελέτη κοινωνικών οντοτήτων (ανθρώπων - **actors**), και των μεταξύ τους σχέσεων και δράσεων
- Οι δράσεις και σχέσεις αναπαρίστανται ως δίκτυο ή γράφος
 - Κάθε κορυφή (κόμβος) αντιστοιχεί σε έναν actor
 - Κάθε ακμή (σύνδεση) αναπαριστά μια σχέση
- Στο γράφο που προκύπτει
 - μελετούμε τη δομή του, το ρόλο κάθε actor, τη θέση του και τη φήμη του (prestige) στο γράφο,
 - Βρίσκουμε ενδιαφέροντες υπο-γράφους (κοινότητες – **communities**)

Κοινωνικά δίκτυα και Web

- Βλέπουμε το Web ως ένα εικονικό κοινωνικό δίκτυο,
 - Κάθε σελίδα (ή web site) είναι ένας actor
 - Κάθε σύνδεσμος (εμφανής ή υποκρυπτόμενο) είναι μια σχέση
- Τεχνικές ανάλυσης κοινωνικών δικτύων μπορούν να εφαρμοστούν στο Web
- Στη συνέχεια εστιάζουμε στο **prestige**, στη σημαντικότητα ενός κόμβου στο γράφο του Web
- Μπορεί κανείς να ορίσει σημαντικούς κόμβους σε ένα κοινωνικό δίκτυο
 - Αυτούς που είναι σημείο αναφοράς στο δίκτυο (θα δούμε σε λίγο)
 - Αυτούς που επηρεάζουν την εξέλιξη της δομής του δικτύου “influencers”)

Κεντρικότητα - Centrality

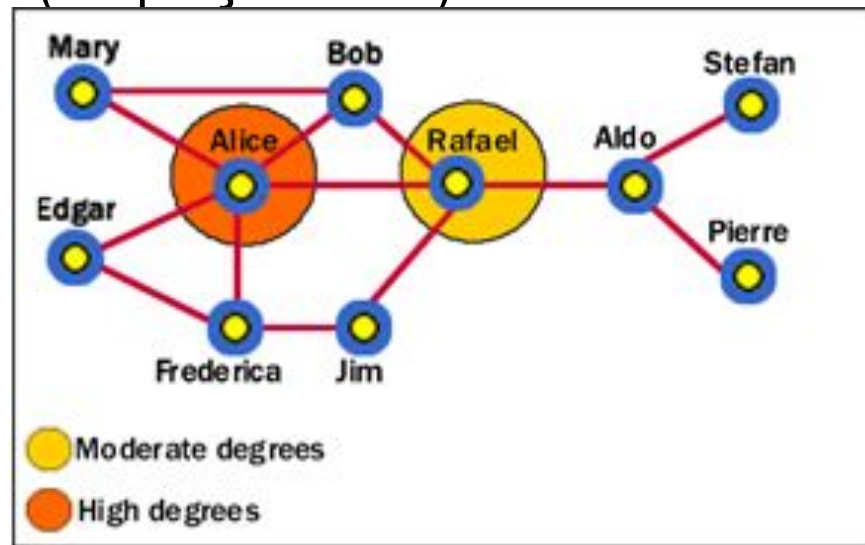
- **Σημαντικοί actors** είναι αυτοί που συνδέονται πυκνά με άλλους actors
- Ένας άνθρωπος που συνδέεται/επικοινωνεί πολύ με άλλους ανθρώπους στο ίδιο δίκτυο είναι πιο σημαντικός από έναν άλλο άνθρωπο με λίγες συνδέσεις
- Ένας κεντρικός actor έχει πολλούς συνδεσμούς (inlinks, outlinks)

Degree Centrality

- Όσο μεγαλύτερος ο βαθμός (degree) ενός κόμβου, τόσο πιο κεντρικός είναι ο κόμβος
- Αν έχουμε συνολικά n actors
- Σε μη κατευθυνόμενο γράφο: Το degree centrality ενός actor i ($C_D(i)$) είναι ο βαθμός του κόμβου (αριθμός ακμών στον κόμβο) προς το μέγιστο βαθμό στο γράφο
- Σε κατευθυνόμενο γράφο: Διακρίνουμε μεταξύ in-links και out-links. Το degree centrality ορίζεται μόνο βάση του out-degree (πλήθος out-links)

$$C_D(i) = \frac{d(i)}{n-1}$$

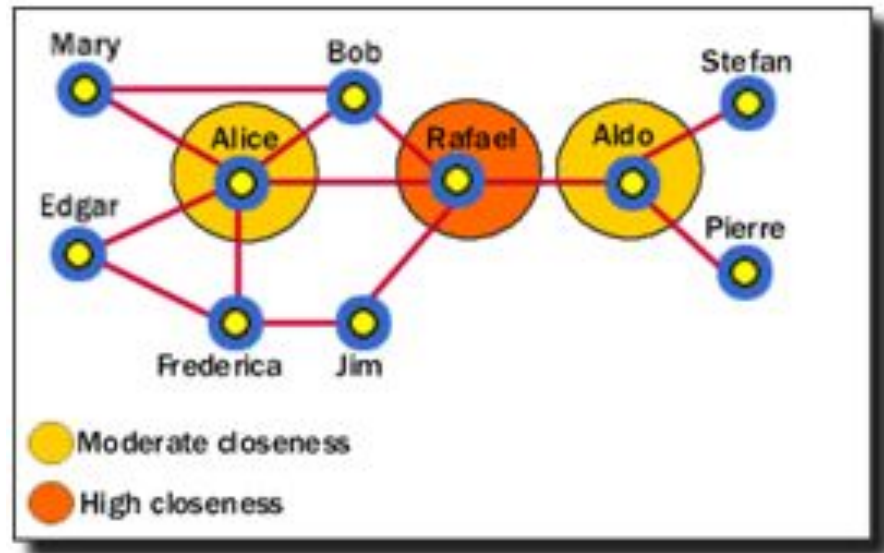
$$C'_D(i) = \frac{d_o(i)}{n-1}$$



Closeness Centrality

- Ο Actor x_i είναι κεντρικός αν μπορεί εύκολα να επικοινωνεί με άλλους actors
- Βασίζεται στην εγγύτητα (**closeness/distance**): Αν η συνολική απόστασή του από όλους τους άλλους actors είναι μικρή

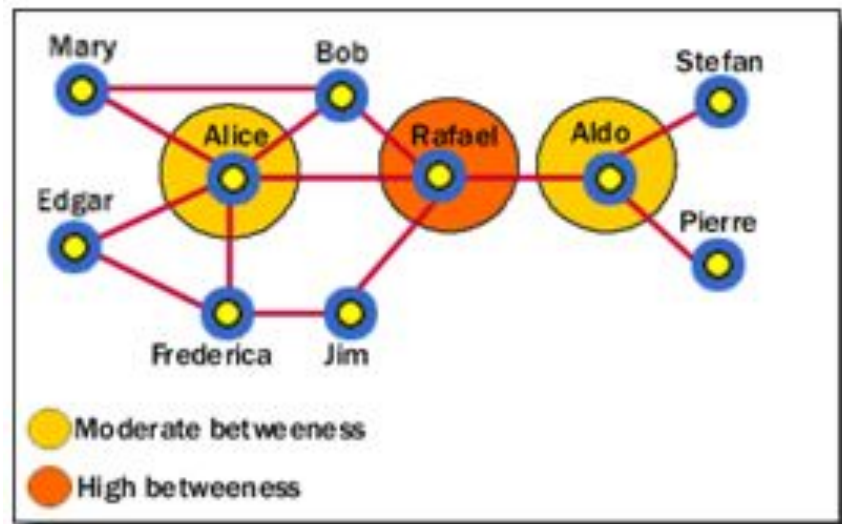
$$C_C(i) = \frac{n-1}{\sum_{j=1}^n d(i,j)}$$



[Image Source: <http://www.fmsasg.com/SocialNetworkAnalysis>]

Betweenness Centrality

- Αν δύο μη-γειτονικοί actors j και k θέλουν να επικοινωνήσουν και ο i βρίσκεται στο μονοπάτι μεταξύ των j και k , τότε ο i έχει έλεγχο στη μεταξύ τους επικοινωνία
- Αν ο i είναι σε πολλά τέτοια μονοπάτια, τότε ο i είναι ένας σημαντικός actor.
 - Δηλ. έχει μεγάλη επιρροή σε ότι συμβαίνει στο δίκτυο



[Image Source: <http://www.fmsasg.com/SocialNetworkAnalysis>]

Betweenness Centrality

- **Μη κατευθυνόμενος γράφος:** Αν p_{jk} είναι το πλήθος των συντομότερων μονοπατιών μεταξύ των actors j και k
- Το betweenness του actor i ορίζεται ως το πλήθος των συντομότερων μονοπατιών που περνούν από το i ($p_{jk}(i)$) προς το πλήθος όλων των συντομότερων μονοπατιών

$$C_B(i) = \sum_{j < k} \frac{p_{jk}(i)}{p_{jk}}$$

- Ο τύπος μπορεί να επεκταθεί για πολλαπλά συντομότερα μονοπάτια μεταξύ των actors j και k

Αναγνώριση - Prestige

- Καλύτερο μέτρο από την κεντρικότητα (centrality)
 - Διακρίνει μεταξύ των εξερχόμενων (**out-links**) και των εισερχόμενων (**in-links**) δεσμών
- Ένας actor με μεγάλη αναγνώριση είναι αυτός που έχει πολλούς εισερχόμενους δεσμούς
 - Το prestige χρησιμοποιεί μόνο in-links.
- **Διαφορά με το centrality:** Το centrality εστιάζει στα out-links (για κατευθυνόμενους γράφους), το prestige εστιάζει στα in-links.
- **Τρεις μετρικές για το prestige:**
 - **Degree prestige:** $P_D(i) = \frac{d_I(i)}{n-1}$, in-degree $d_I(i)$ του actor i
 $n =$ πλήθος κόμβων
 - **Proximity prestige**
 - **Rank prestige:** είναι η βάση για τους αλγορίθμους **PageRank** και **HITS**

Proximity prestige

- Το degree prestige του actor i εξετάζει μόνο τους άμεσους γείτονες του i
- Το **proximity prestige** γενικεύει, εξετάζοντας και τους άμεσους και τους έμμεσους γείτονες (αυτούς που μπορούν να προσεγγίσουν) του actor i .
- Η εγγύτητα (**proximity**) ορίζεται πάνω στην απόσταση των άλλων actors ($d(j,i)$) από τον i
- Χρησιμοποιούμε τη μέση απόσταση
- Το proximity prestige είναι το πλήθος όλων των γειτόνων το

$$P_P(i) = \frac{|I_i|/(n-1)}{\sum_{j \in I_i} d(j,i) / |I_i|}$$

Rank prestige

- Οι άλλες δύο μετρικές αγνοούν το επιμέρους prestige των actors που δείχνουν προς τον actor i
- Στην πραγματικότητα, αν κάποιος άνθρωπος i προτείνεται από έναν αναγνωρισμένο άνθρωπο j , τότε θεωρείται πιο αναγνωρισμένο απ' ό,τι αν προτείνεται από ένα λιγότερο αναγνωρισμένο άνθρωπο.
 - Καλές οι συστάσεις, αλλά εξαρτάται και ποιος σε συστήνει
- Αν η γειτονιά (*circle of influence*) κάποιου περιέχει πολλούς prestigious actors, τότε κι αυτός έχει μεγάλο prestige
 - Το prestige του επηρεάζεται από τα rank των γειτόνων του

Rank Prestige

- Με βάση τα προηγούμενα, το rank prestige $P_R(i)$ ορίζεται ως ο γραμμικός συνδυασμός των rank prestige όλων όσων δείχνουν με κάποιο link στον i :

$$P_R(i) = A_{1i}P_R(1) + A_{2i}P_R(2) + \dots + A_{ni}P_R(n)$$

όπου $A_{ji} = 1$ αν το j δείχνει στο i , αλλιώς 0

Περιεχόμενα

- Εισαγωγή
- **Ανάλυση κοινωνικών δικτύων**
- HITS
- PageRank
- Συν-αναφορά (Co-citation) και βιβλιογραφική σύζευξη (coupling)
- Web Spamming

Περιεχόμενα

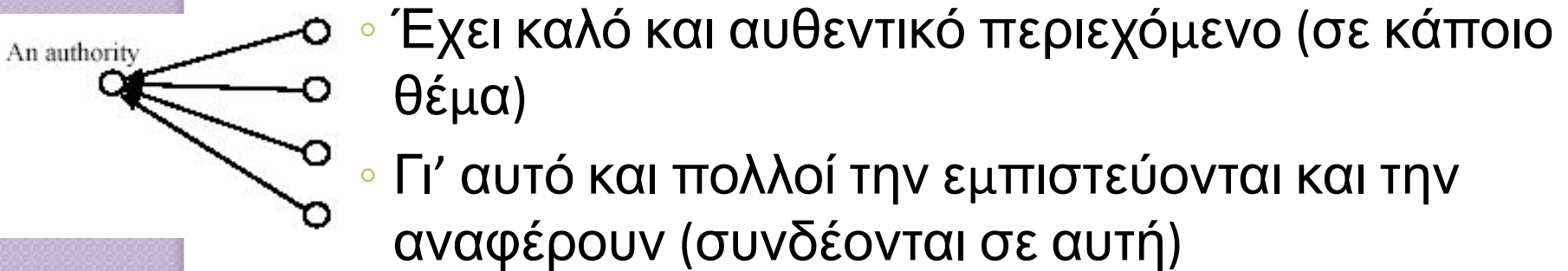
- Εισαγωγή
- Ανάλυση κοινωνικών δικτύων
- Βαθμονόμηση
 - HITS
 - PageRank
- Ομοιότητα
 - Συν-αναφορά (Co-citation) και βιβλιογραφική σύζευξη (coupling)
- Web Spamming

HITS

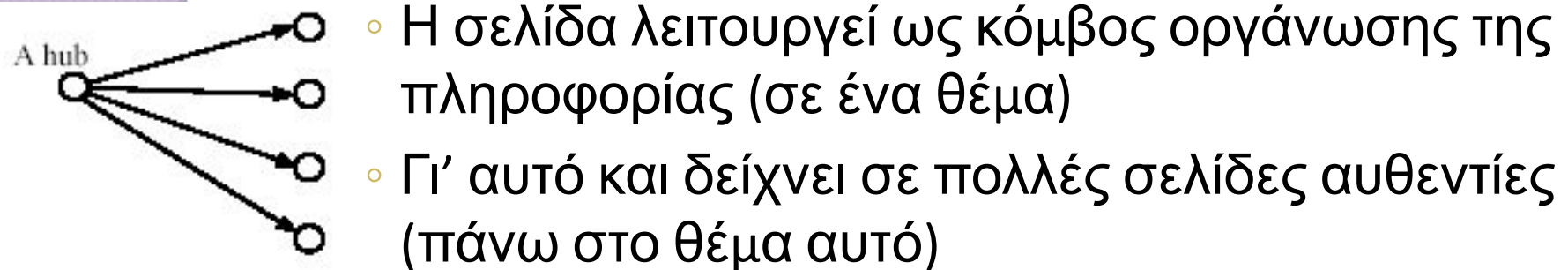
- **Hypertext Induced Topic Search.**
- Το σκορ που παράγει ο HITS για μια σελίδα αλλάζει με το ερώτημα (**search query-dependent**)
- Όταν ο χρήστης κάνει ένα ερώτημα
 - Ο HITS επεκτείνει τη λίστα με τις σχετικές σελίδες που επιστρέφει η μηχανή αναζήτησης
 - προσθέτοντας σελίδες που δείχνουν ή δείχνονται από τις σελίδες της λίστας
 - Στη συνέχεια παράγει δύο βαθμονομημένες λίστες για τις σελίδες του επεκταμένου συνόλου:
authority ranking και **hub ranking**.

Αυθεντίες και Πηγές

Αυθεντία (Authority): Μια σελίδα με πολλά in-links

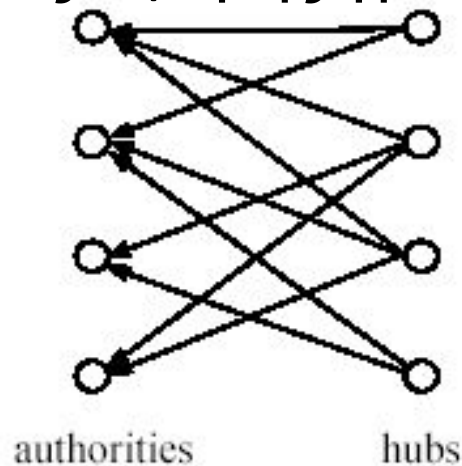


Πηγή (Hub): Μια σελίδα με πολλά out-links



Η βασική ιδέα του HITS

- Ένα καλό hub δείχνει σε πολλά καλά authorities, και
- Ένα καλό authority δείχνεται από πολλά καλά hubs
- Τα authorities και τα hubs έχουν μια σχέση **αμοιβαίας ενίσχυσης** (το ένα ενδυναμώνει το άλλο)
- Παράδειγμα: ένας διμερής γράφος



HITS: Ανάκτηση σημαντικών σελίδων

- Για ένα ερώτημα q που μπορεί να φέρει πολλές σελίδες ως απάντηση ο HITS:
 - Συλλέγει τις πρώτες σελίδες t (π.χ. $t=200$) που έρχονται ως απάντηση (βαθμονομημένες π.χ. με βάση το tf-idf): αρχικό σύνολο σελίδων W (root set)
 - Αυξάνουν το W προσθέτοντας κάθε σελίδα που δείχνεται από μια σελίδα του W και κάθε σελίδα που δείχνει σε μια σελίδα του W : Το διευρυμένο σύνολο σελίδων S (base set)
 - Με βάση τις σελίδες του S και τους μεταξύ τους υπερσυνδέσμους ο HITS δημιουργεί ένα γράφο $G = (V, E)$
 - Ορίζει τον πίνακα γειτνίασης L (adjacency matrix) του γράφου

$$L_{ij} = \begin{cases} 1 & \text{if } (i, j) \in E \\ 0 & \text{otherwise} \end{cases}$$

Υπολογισμός hub και authority

- Υπολογίζει ένα **authority score** και ένα **hub score** για κάθε σελίδα στο S
- Έστω για μια σελίδα i είναι $a(i)$ και $h(i)$ αντίστοιχα
- Η σχέση αμοιβαίας ενίσχυσης είναι:

$$a(i) = \sum_{(j,i) \in E} h(j)$$

$$h(i) = \sum_{(i,j) \in E} a(j)$$

Πλεονεκτήματα και μειονεκτήματα

- **Πλεονεκτήματα:** μπορεί να ταξινομεί τις σελίδες ως προς ένα θέμα (προσδιορίζεται από το ερώτημα) κι έτσι μπορεί να αναδείξει authorities και hubs για το θέμα αυτό
- **Αδυναμίες:**
 - **Εύκολα μπορεί να επηρεαστεί (spam).** Αν φτιάξω μια σελίδα που περιέχει πολλά out-links (σε authorities και σε δικές μου σελίδες), η σελίδα γίνεται καλό hub και οι υπόλοιπες δικές μου σελίδες αποκτούν μεγάλο authority score
 - **Μετατόπιση (αλλοίωση) θέματος.** Πολλές σελίδες στο εκτεταμένο σύνολο μπορεί να είναι εκτός θέματος
 - **Μειωμένη απόδοση στις αναζητήσεις:** Η διαδικασία εκτέλεσης ερωτήματος, ανάκτησης βασικού και εκτεταμένου συνόλου είναι χρονοβόρα

Περιεχόμενα

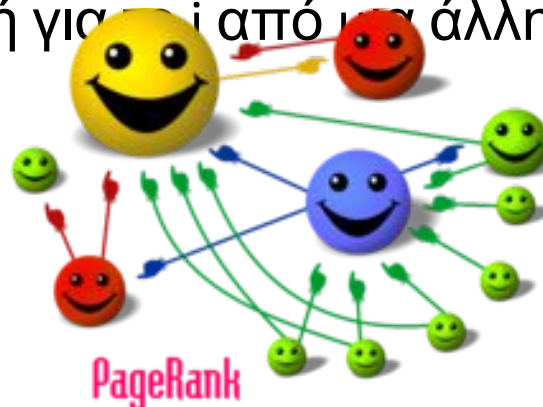
- Εισαγωγή
- Ανάλυση κοινωνικών δικτύων
- Βαθμονόμηση
 - HITS
 - PageRank
- Ομοιότητα
 - Συν-αναφορά (Co-citation) και βιβλιογραφική σύζευξη (coupling)
- Web Spamming

PageRank

- Ο PageRank επικράτησε έναντι του HITS χάρη:
 - Στην ανεξαρτησία του από το ερώτημα,
 - Στην ικανότητά του να αντιμετωπίζει το spamming
 - Στην επιτυχία της μηχανής αναζήτησης της Google
- Ο PageRank στηρίζεται στη «δημοκρατική φύση» του Web και αξιοποιεί το σύνολο των συνδέσμων για τον υπολογισμό της βαθμολογίας μιας σελίδας
- Ερμηνεύει κάθε σύνδεσμο από τη σελίδα x στη σελίδα y ως μια «ψήφο» της x για την y
- Παράλληλα, λαμβάνει υπόψη του τη «σημαντικότητα» της σελίδας x που έδωσε την ψήφο στην y
- Θυμίζει κάτι; ☐ **rank prestige.**
<http://infolab.stanford.edu/~backrub/google.html>

Συγκεκριμένα

- Ένας σύνδεσμος από μια σελίδα σε μια άλλη υποδηλώνει αυθεντία της σελίδα που δείχνεται.
 - Όσο πιο πολλά in-links έχει μια σελίδα i , τόσο μεγαλύτερο το prestige της
- Οι σελίδες που δείχνουν το i έχουν το δικό τους σκορ prestige
 - Μια σελίδα με υψηλό prestige που δείχνει στο i , είναι πιο σημαντική για το i από μια άλλη με μικρότερο prestige

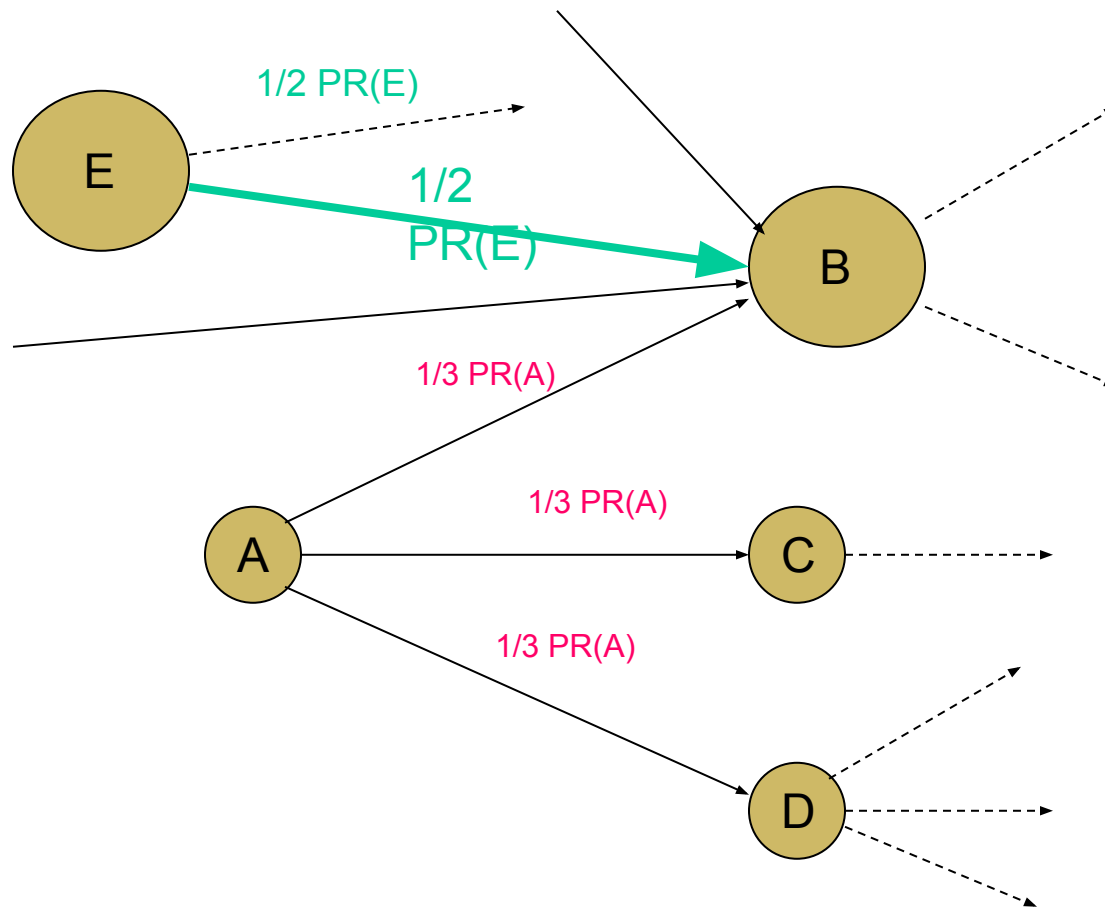


Ο αλγόριθμος του PageRank

- Το PageRank σκορ μιας σελίδας i είναι το άθροισμα των PageRank σκορ των σελίδων που δείχνουν στο i
- Καθώς μια σελίδα μπορεί να δείχνει σε πολλές άλλες, το PageRank σκορ της πρέπει να μοιράζεται
 - Η διαδικασία υπολογισμού ξεκινά με όλους τους κόμβους να έχουν το ίδιο σκορ
 - Ένας τυχαίος κόμβος επιλέγεται και μοιράζει το σκορ του σε όσους δείχνει
 - Η διαδικασία επαναλαμβάνεται μέχρις ότου όλοι οι κόμβοι να μοιράσουν το σκορ τους
 - Τα βήματα 2 και 3 επαναλαμβάνονται μέχρι τα PageRank σκορ των σελίδων να συγκλίνουν (αποδεικνύεται ότι στο web συγκλίνουν)

http://www.webworkshop.net/pagerank_calculator.php

Παράδειγμα



Το μοντέλο του τυχαίου περιηγητή (Random Surfer Model)

- Οι επισκέπτες μιας σελίδας στο web έχουν δύο επιλογές:
 - Να επιλέξουν ένα από τα out-links
 - Η πιθανότητα να επιλεγεί ένα συγκεκριμένο out-link στη σελίδα A είναι $1/m$
 - m ο αριθμός των out-links στη σελίδα A
 - Να μεταφερθούν σε μια άλλη σελίδα δίνοντας το URL της απευθείας (*teleport*)
 - Η πιθανότητα της απευθείας μεταφοράς είναι $1/n$
 - n ο αριθμός των σελίδων του web site (της συλλογής μας, ή του web γενικότερα)
- Το PageRank σκορ μιας σελίδας: η πιθανότητα ο επισκέπτης να βρεθεί σε μια σελίδα κατά την περιήγησή του στο web

Υπολογισμός της πιθανότητας

- Με χρήση αλυσίδων Markov
- Σε μια αλυσίδα Markov (ακολουθία κόμβων)
 - κάθε σελίδα θεωρείται μια κατάσταση και
 - κάθε σύνδεσμος θεωρείται μια μετάβαση από μια κατάσταση σε μια άλλη και έχει συγκεκριμένη πιθανότητα
- Ο PageRank θεωρεί ότι η περιήγηση στο Web είναι μια στοχαστική διαδικασία (μη προκαθορισμένη – μη ντετερμινιστική - έχει τυχαιότητα στην έκβασή της)
 - Ο επισκέπτης μπορεί να ακολουθήσει ένα σύνδεσμο (ντετερμινιστικό), αλλά
 - μπορεί και να μεταβεί σε μια εντελώς νέα διεύθυνση (τυχαίο)
- Ο επισκέπτης των σελίδων (**Web surfer**) τυχαία επιλέγει μεταβάσεις

Τυχαίες επισκέψεις

- Το πλήθος out-links μιας σελίδας i δηλώνεται O_i
- Κάθε πιθανότητα μετάβασης (transition probability) είναι $1/O_i$ αν θεωρήσουμε ότι όλοι οι υπερσύνδεσμοι επιλέγονται ομοιόμορφα από τους χρήστες και με την προϋπόθεση ότι
 - Ο χρήστης δεν πατά ποτέ το κουμπί “back” και
 - Δεν δίνει ποτέ απευθείας ένα νέο URL
- Μπορούμε για το γράφο του Web να δημιουργήσουμε τον πίνακα γειτνίασης A όπου

$$A_{ij} = \begin{cases} \frac{1}{O_i} & \text{if } (i, j) \in E \\ 0 & \text{otherwise} \end{cases}$$

E , το σύνολο όλων των συνδέσμων του γράφου

- Και να υπολογίσουμε τα PageRank σκορ με βάση την εξίσωση: $P = A^T P$

Όμως

- Στην πραγματικότητα κανένα από τα δύο δεν ισχύει
- Ο χρήστης που βρίσκεται σε μια σελίδα μπορεί να πάει σε οποιαδήποτε άλλη σελίδα του Web
- Η λύση είναι να προσθέσουμε στο γράφο του Web όλους τους συνδέσμους που λείπουν (ώστε να γίνει ισχυρά συνδεδεμένος γράφος) δίνοντας στις ακμές αυτές ένα πολύ μικρό βάρος
 - Θεωρούμε ότι υπάρχει μια πιθανότητα d για το χρήστη να επιλέξει ένα out-link
 - και μια πιθανότητα $1-d$ να επιλέξει ένα τυχαίο URL

$$P = \left((1-d) \frac{E}{n} + dA^T \right) P$$

E , $n \times n$ πίνακας
μόνο με 1

Ο τύπος του PageRank

- Υπολογίζεται επαναληπτικά
- Σε κάθε επανάληψη:

$$P(i) = (1 - d) + d \sum_{(j,i) \in E} \frac{P(j)}{O_j}$$

-
- O_j : σελίδες που δείχνει η j
- $(1-d)$: damping factor
- Θεωρεί ότι η πιθανότητα μεταφοράς (teleportation probability) είναι σταθερή: $tp = 1/N$
- N : μέγεθος του γράφου

Πλεονεκτήματα του PageRank

- **Αντιμετωπίζει το spam.** Μια σελίδα είναι σημαντική μόνο αν τη δείχνουν σημαντικές σελίδες
 - Δεν είναι εύκολο για τον ιδιοκτήτη μιας σελίδας να προσθέσει in-links στη σελίδα του, καθώς χρειάζεται πρόσβαση σε άλλες σημαντικές σελίδες
 - Βέβαια με την έλευση των blogs εμφανίστηκε και το google bombing
- **Ο PageRank είναι ένα καθολικό μέτρο ανεξάρτητο από τα ερωτήματα**
 - Οι τιμές του PageRank υπολογίζονται off-line και αποθηκεύονται και είναι άμεσα διαθέσιμες στη διάρκεια ενός ερωτήματος.
- **Κριτική:** Ανεξαρτησία από το ερώτημα: Δεν μπορεί να ξεχωρίσει μεταξύ γενικού και ειδικού ενδιαφέροντος authorities

Παραλλαγές PageRank

- Τροποποιούν:
 - είτε τον τρόπο υπολογισμού της πιθανότητας επιλογής ενός από τα out-links
 - είτε τον τρόπο υπολογισμού της πιθανότητας τυχαίας μετάβασης
- Εξατομικευμένος PageRank
 - Το teleportation probability δεν είναι ίδιο για όλες τις σελίδες, είναι διαφορετικό για κάποιες «αγαπημένες» σελίδες
- Biased PageRank
 - Αναλύεται το anchor text κάθε υπερσυνδέσμου (ή το κείμενο κάθε υποδεικνυόμενης σελίδας)
 - Η πιθανότητα επιλογής ενός out-link εξαρτάται από το κείμενο του anchor text ή της υποδεικνυόμενης σελίδας

Περιεχόμενα

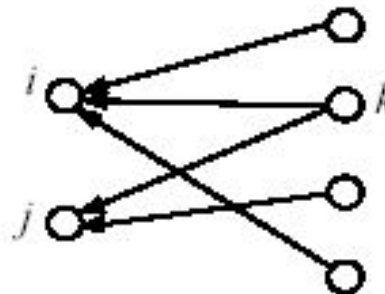
- Εισαγωγή
- Ανάλυση κοινωνικών δικτύων
- Βαθμονόμηση
 - HITS
 - PageRank
- Ομοιότητα
 - Συν-αναφορά (Co-citation) και βιβλιογραφική σύζευξη (coupling)
- Web Spamming

Συν-αναφορά και Σύζευξη

- Co-citation και Bibliographic Coupling
- Βασίζονται στη θεωρία ανάλυσης αναφορών (**citation analysis**) σε βιβλιογραφικές συλλογές
 - Μια δημοσίευση (ένα άρθρο, βιβλίο κλπ) δίνει αναφορές σε προηγούμενα δημοσιευμένα έργα ως αναγνώριση των ιδεών που παρουσιάζονται σε αυτά
- Μια αναφορά από ένα άρθρο σε ένα άλλο δημιουργεί μια (κατευθυνόμενη) σχέση μεταξύ τους

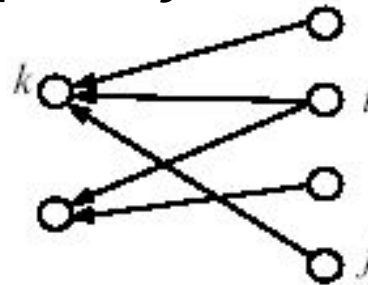
Co-citation

- **Η συν-αναφορά (Co-citation)** είναι ένα μέτρο ομοιότητας μεταξύ των i και j
- C_{ij} : Το πλήθος των άρθρων που αναφέρει από κοινού τα i και j
- C_{ii} : Το πλήθος των άρθρων που αναφέρουν το i
- Αν τα άρθρα i και j αναφέρονται από κοινού από το άρθρο k , τότε είναι πιθανό να σχετίζονται μεταξύ τους (π.χ. να μιλούν για το ίδιο θέμα)
- Όσο πιο πολλά άρθρα αναφέρουν από κοινού τα i και j τόσο πιο ισχυρή είναι η σχέση
- L είναι ο πίνακας συν-αναφορών (cocitation matrix):
 $L_{ij} = 1$ αν το άρθρο i αναφέρει το j , αλλιώς 0.



Bibliographic coupling

- Λειτουργεί παρόμοια
- **Βιβλιογραφική σύζευξη (Bibliographic coupling)** έχουμε όταν δύο άρθρα i και j αναφέρουν από κοινού το άρθρο k
- B_{ij} : Το πλήθος των άρθρων που αναφέρονται από κοινού από τα i και j
- B_{ii} : Το πλήθος των άρθρων που αναφέρει το άρθρο i
- Όσο πιο πολλά άρθρα αναφέρουν από κοινού τα i και j τόσο πιο ισχυρή είναι η ομοιότητά τους



Περιεχόμενα

- Εισαγωγή
- Ανάλυση κοινωνικών δικτύων
- Βαθμονόμηση
 - HITS
 - PageRank
- Ομοιότητα
 - Συν-αναφορά (Co-citation) και βιβλιογραφική σύζευξη (coupling)
- **Web Spamming**

Web Spamming

- **Spamming:** Η προσπάθεια παραπλάνησης των μηχανών αναζήτησης ώστε να βαθμονομούν συγκεκριμένες σελίδες πιο ψηλά απ' ότι θα έπρεπε
 - SEO (Search Engine Optimization)
- Πολλαπλοί τρόποι spamming:
 - Content spamming (term spamming): παρέμβαση στα περιεχόμενα μιας σελίδας ώστε να φαίνεται πιο σχετική με ορισμένα ερωτήματα (Title, Meta-tags, Body, Anchor text, URL)
 - Link spamming: παρέμβαση στη δομή του γράφου του web με σελίδες και συνδέσμους που δημιουργούνται αυτοματοποιημένα

Content Spamming

- **Επανάληψη σημαντικών όρων**
 - Αυξάνει το TF των όρων αυτών
 - Η απλή επανάληψη εντοπίζεται, γι' αυτό και συχνά οι όροι ανακατεύονται σε προτάσεις (συνήθως χωρίς νόημα)
- **Προσθήκη πολλών άσχετων όρων**
 - Έτσι η σελίδα σχετίζεται με πολλά ερωτήματα
 - Γίνεται με αντιγραφή περιεχομένου από πολλές σχετικές σελίδες και επικόλλησή του στη σελίδα (το αποτέλεσμα δεν βγάζει νόημα)

Link Spamming

- **Out-link spamming**

- Προσθέτουμε πολλά out-links προς σημαντικές σελίδες (π.χ. σελίδες μηχανών αναζήτησης, μεγάλων οργανισμών και εταιριών) ώστε να αυξήσουμε το hub σκορ
- **Directory cloning:** αναπαράγουμε το περιεχόμενο μεγάλων καταλόγων του δικτύου (π.χ. dmoz.org)

- **In-link spamming**

- Πιο δύσκολη
- Καταχωρούμε συνδέσμους προς τη σελίδα μας σε καταλόγους του Web
- Δημιουργούμε συνδέσμους προς τη σελίδα μας σε user-generated content (forums, blogs, κλπ.)
- Φτιάχνουμε **link farms** (πολλά δικά μας sites με συνδέσμους προς τη σελίδα μας) ή συμμετέχουμε σε δίκτυα ανταλλαγής συνδέσμων

Πως κρύβουμε το spam;

- Απόκρυψη περιεχομένου

- Πώς κάνουμε το spam περιεχόμενο αόρατο;
- Από το χρήστη, από μια anti-spam μηχανή

- Cloaking

- Επιστρέφω διαφορετικό περιεχόμενο σε έναν χρήστη και σε μια μηχανή (π.χ. crawler)

- Redirection

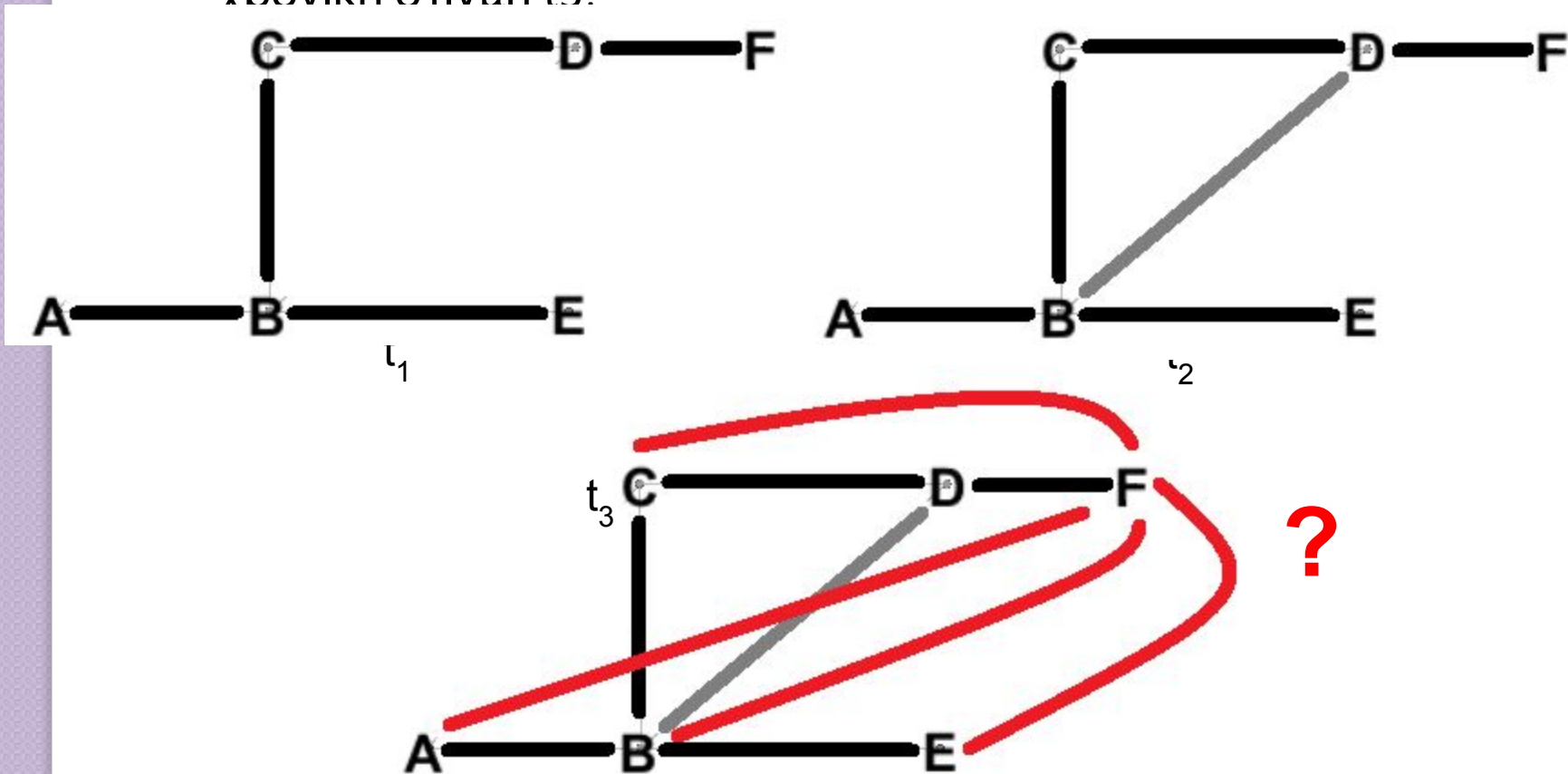
- Ανακατευθύνω το browser από μια «φαινομενικά ενδιαφέρουσα» σε μια spam σελίδα

Αντιμετώπιση Spam

- Πολλές τεχνικές
 - Ο PageRank έδωσε κάποιες λύσεις
 - Τεχνικές κατηγοριοποίησης εκτιμούν αν μια σελίδα είναι spam ή όχι
 - Μέγεθος σελίδας
 - Μέσο μέγεθος λέξης
 - Πλήθος λέξεων στον τίτλο
 - Ποσοστό ορατού περιεχομένου
 - κλπ.
- Ανάλυση του ρυθμού δημιουργίας περιεχομένου και συνδέσμων

Link prediction

- Χρήσιμο σε κοινωνικά δίκτυα
- Αν γνωρίζω τη δομή του δικτύου μου τις χρονικές στιγμές t_1 και t_2 μπορώ να προβλέψω ποιες ακμές θα εμφανιστούν τη χρονική στιγμή t_3 :

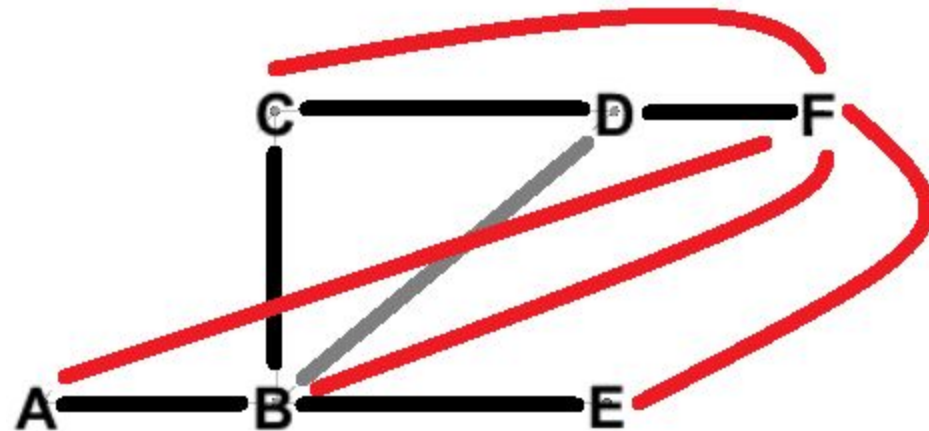


Συμβολισμοί

- Graph: $G\langle V, E \rangle$
- Edge: $e = \langle u, v \rangle \in E$
- Graph snapshots: $G[t_0, t_1]$
- Graph current and future status: $G[t_2, t_3]$
- Εκτιμώμενο σκορ ακμής: $\text{Score}(u, v)$

Ομοιότητα κόμβων

- Όσο περισσότερο μοιάζουν δύο κόμβοι τόσο πιο πιθανό είναι να συνδεθούν
- **Ομοιότητα βάση απόστασης:**
 - Όσο πιο κοντά είναι δύο κόμβοι τόσο πιο όμοιοι είναι. Shortest Path (Dijkstra's algorithm)
 - $\text{Sim}(A,F) = -d(A,F) = -3$
 - $\text{Sim}(B,F) = -d(B,F) = -2$
 - $\text{Sim}(C,F) = -d(C,F) = -2$
 - $\text{Sim}(E,F) = -d(E,F) = -3$

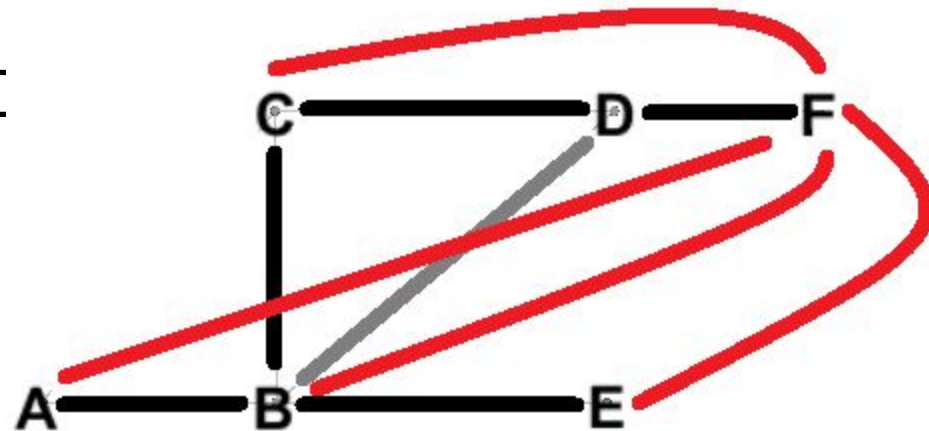


$\text{Score}(x,y) = -\text{Shortest Path Length}(x,y)$

Ομοιότητα κόμβων

- **Ομοιότητα βάση κοινών γειτόνων:**

- Όσο πιο πολλούς κοινούς γνωστούς έχουν δύο κόμβοι τόσο πιο όμοιοι είναι
- $\text{Sim}(A,F) = |\{B\} \cap \{D\}| = 0$
- $\text{Sim}(B,F) = |\{A,C,D,E\} \cap \{D\}| = 1$
- $\text{Sim}(C,F) = |\{B,D\} \cap \{D\}| = 1$
- $\text{Sim}(E,F) = |\{B\} \cap \{D\}| = 0$



$$\text{Score}(x,y) = |N(x) \cap N(y)|$$

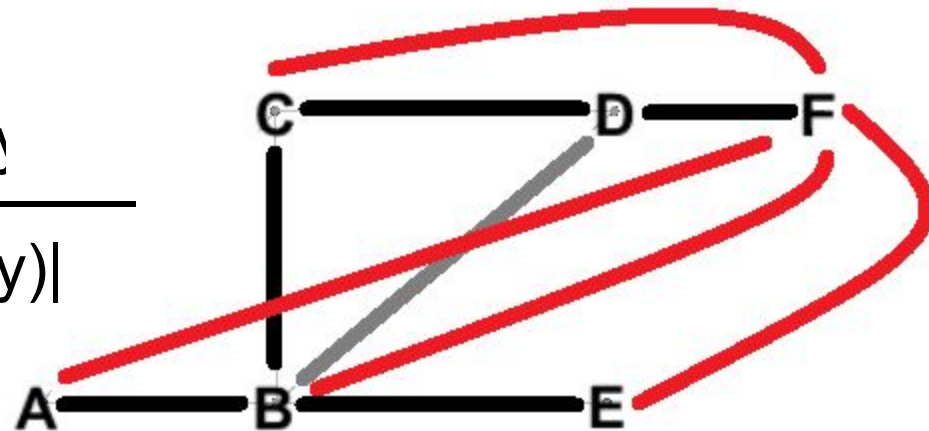
Ομοιότητα κόμβων

- **Jaccard Similarity.** Διαιρώ με το union για κανονικοποίηση

$$\text{Sim}(B,F) = |\{A,C,D,E\} \cap \{D\}| / |\{A,C,D,E\} \cup \{D\}| = 1/4$$

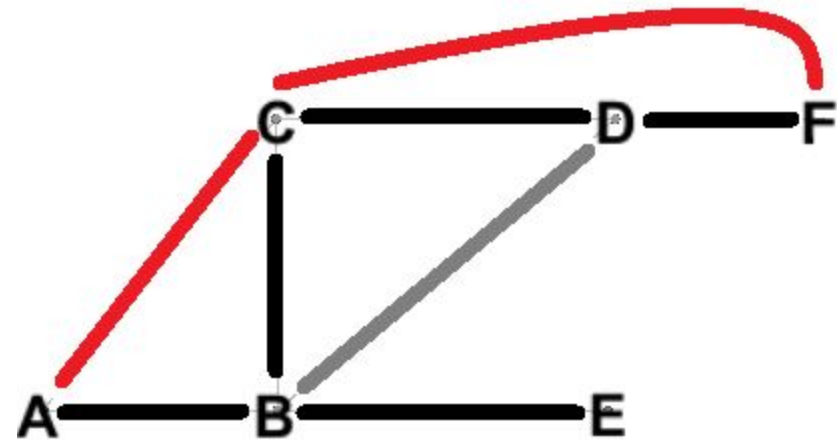
$$\text{Sim}(C,F) = |\{B,D\} \cap \{D\}| / |\{B,D\} \cup \{D\}| = 1/2$$

$$\text{Score}(x,y) = \frac{|N(x) \cap N(y)|}{|N(x) \cup N(y)|}$$



Ομοιότητα κόμβων

- Adamic/Adar: **Κοινοί γείτονες με βάρη.**
- Οι κοινοί γείτονες που έχουν λίγους άλλους γείτονες είναι πιο σημαντικές συστάσεις
 - $N(A) \cap N(C) = \{B\}$
 - $N(C) \cap N(F) = \{D\}$
 - $N(B) = \{A, C, D, E\}$
 - $N(D) = \{B, C, F\}$
 - $\text{Sim}(A, C) = 1/\log(4) = 1.7$
 - $\text{Sim}(C, F) = 1/\log(3) = 2.1$



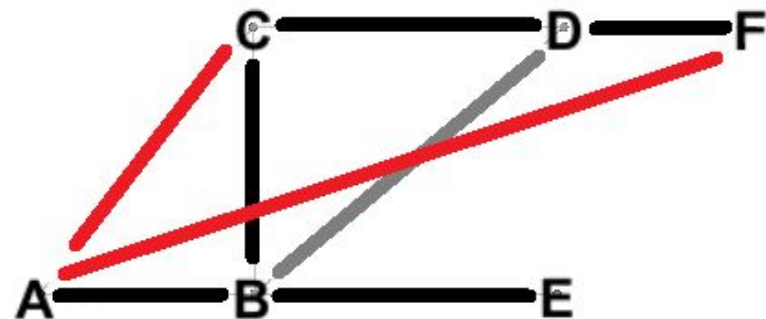
$$\text{Score}(x, y) = \sum_{z \in N(x) \cap N(y)} \frac{1}{\log |N(z)|}$$

Ομοιότητα κόμβων

- **Katz measure:**

- Όσο πιο πολλές σύντομες διαδρομές υπάρχουν μεταξύ δύο κόμβων τόσο πιο όμοιοι είναι.
- $\text{path}^2(A,C) = \{(A,B,C)\}$ $\text{path}^3(A,C) = \{(A,B,D,C)\}$
- $\text{path}^2(A,F) = \{\}$ $\text{path}^3(A,F) = \{(A,B,D,F)\}$ $\text{path}^4(A,F) = \{(A,B,C,D,F)\}$

$$\text{Score}(x, y) = \sum_{l=1}^{\infty} \beta^l * |\text{path}_{x,y}^l|$$



- $\text{Sim}(A,C) = 1/2 * 1 + 1/4 * 1 = 0.75$
- $\text{Sim}(C,F) = 1/2 * 0 + 1/4 * 1 + 1/8 * 1 = 0.375$

- Γενικά Katz index = $\beta A + \beta^2 A^2 + \beta^3 A^3$

Friends-measure

- Όσο πιο πολλούς κοινούς γείτονες ή όσο πιο πολλούς συνδεδεμένους γείτονες έχουν τόσο πιο όμοιοι είναι.

$$Score(x, y) = \sum_{u \in N(x)} \sum_{v \in N(y)} \delta(u, v)$$

- $\delta(u, v) = 1$ αν $u = v$ ή $\langle u, v \rangle \in E$, αλλιώς $\delta = 0$
- Local Path: Katz index για path length 2 ή 3 μόνο $LP = A^2 + \alpha A^3$

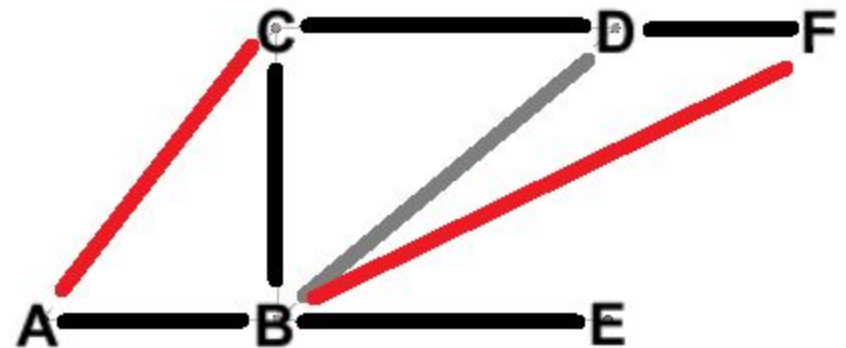
Τυχαία περιήγηση (random walk)

- **Τάση σύνδεσης** (Preferential attachment):
Όσο πιο πολλούς γείτονες έχει ένας κόμβος τόσο πιο μεγάλη η τάση του να συνδεθεί (rich get richer).

- $\text{Sim}(A,C) = |N(A)| * |N(C)| = 1 * 2 = 2$

- $\text{Sim}(B,F) = |N(B)| * |N(F)| = 4 * 1 = 4$

$$\text{Score}(x,y) = |N(x)| * |N(y)|$$



Τυχαία περιήγηση (random walk)

- Ένας τυχαίος περιηγητής σε κάθε κόμβο έχει ίση πιθανότητα να ακολουθήσει μια από τις εξερχόμενες ακμές
- Hitting time: Πόσα βήματα θα χρειαστεί ο τυχαίος περιηγητής για να πάει από τον ένα κόμβο στον άλλο

$$\text{Score}(x,y) = -H(x,y)$$

- Ένας τυχαίος περιηγητής σε κάθε κόμβο έχει μια πιθανότητα (α) να ακολουθήσει μια από τις εξερχόμενες ακμές και τη συμπληρωματική της πιθανότητα ($1-\alpha$) να ξαναγυρίσει στην αρχή
- Rooted PageRank: Μειώνει τα score σε πολύ μακρινά μονοπάτια

$$\text{Score}(x,y) = -H(x,y) * \pi_y$$

όπου π_y = πόσες φορές περνά από το y (κεντρικότητα του y)