Ανάκληση Πληροφορίας

Διδάσκων –
Δημήτριος Κατσαρός

Διάλεξη 17η: 21/05/2014

Η μέθοδος HITS

Hypertext Induced Topic Search

Hypertext Induced Topic Search (HITS)

- Επινοήθηκε από τον Jon Kleinberg το 1998
- Διαφορές των δυο κύριων αλγορίθμων ranking:
  - Ο PageRank είναι query-independent
  - Ο HITS είναι query-dependent
  - Ο PageRank παράγει ένα μέτρο "σημαντικότητας" που χαρακτηρίζει κάθε ιστοσελίδα
  - Ο HITS παράγει δυο τέτοιους αριθμούς
    - Το authority score & το hub score
- Ο HITS αναλύει τις σελίδες ως authorities και hubs
  - Μια authority είναι μια σελίδα με πολλούς εισερχόμενους υπερσυνδέσμους
  - Ένα hub είναι μια σελίδα με πολλούς εξερχόμενους υπερσυνδέσμους
- *Οι καλές authorities δείχνονται από καλά hubs, και τα καλά hubs δείχνουν σε καλές authorities*

Hypertext Induced Topic Search (HITS)

- Ο HITS ενσωματώθηκε στο έργο της IBM “CLEVER”
- Αποτέλεσε τη βάση για τη μηχανή αναζήτησης Teoma (αγοράστηκε από την Ask Jeeves, τώρα Ask.com
- Συνεπώς, κάθε σελίδα i έχει ένα **authority score $\mathbf{a_i}$** και ένα **hub score $\mathbf{h_i}$**
- Με e_{ij} συμβολίζουμε την ύπαρξη υπερσυνδέσμου από την ιστοσελίδα i στην j
- Υποθέτουμε ότι αρχικά έχουμε αναθέσει σε κάθε ιστοσελίδα ένα authority score $\mathbf{x_i}$ και ένα hub score $\mathbf{y_i}$
- Ο HITS υπολογίζει επαναληπτικά τις ποσότητες:

$$x_i^k = \sum_{j:e_{ji} \in E} y_j^{(k-1)} \qquad y_i^k = \sum_{j:e_{ij} \in E} x_j^{(k)} \qquad k = 0, 1, 2, \ldots$$

Hypertext Induced Topic Search (HITS)

- Έστω ο πίνακας γειτνίασης $\mathbf{L}_{ij}$ με στοιχεία ίσα με 1, εάν υπάρχει υπερσύνδεσμος από την ιστοσελίδα i στην j, και ίσα με 0, στην άλλη περίπτωση
- Οι προηγούμενες επαναληπτικές εξισώσεις μπορούν να γραφούν με τη βοήθεια πινάκων ως εξής:

$$\mathbf{x}^{(k)}=\mathbf{L}^T\mathbf{y}^{(k-1)} \quad \text{και} \quad \mathbf{y}^{(k)}=\mathbf{L}\mathbf{x}^{(k)}$$

Ο αρχικός αλγόριθμος HITS

- Αρχικοποίηση του $\mathbf{y}^{(0)}=\mathbf{e}$ (και άλλες επιλογές αρχικοποίησης είναι πιθανές)
- Μέχρι να επέλθει σύγκλιση:
  - $\mathbf{x}^{(k)}=\mathbf{L}^T\mathbf{y}^{(k-1)}$
  - $\mathbf{y}^{(k)}=\mathbf{L}\mathbf{x}^{(k)}$
  - ++k;
  - Κανονικοποίηση των $\mathbf{x}^{(k)}$ και $\mathbf{y}^{(k)}$

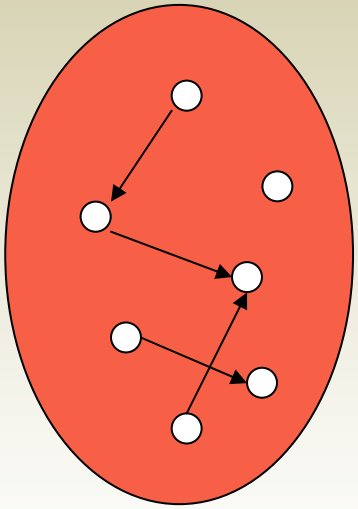
Hypertext Induced Topic Search (HITS)

- Οι προηγούμενες εξισώσεις μπορούν να απλοποιηθούν στις επόμενες:

$$\mathbf{x}^{(k)}=\mathbf{L}^{T}\mathbf{L}\mathbf{x}^{(k-1)} \quad \text{και} \quad \mathbf{y}^{(k)}=\mathbf{L}\mathbf{L}^{T}\mathbf{y}^{(k)}$$

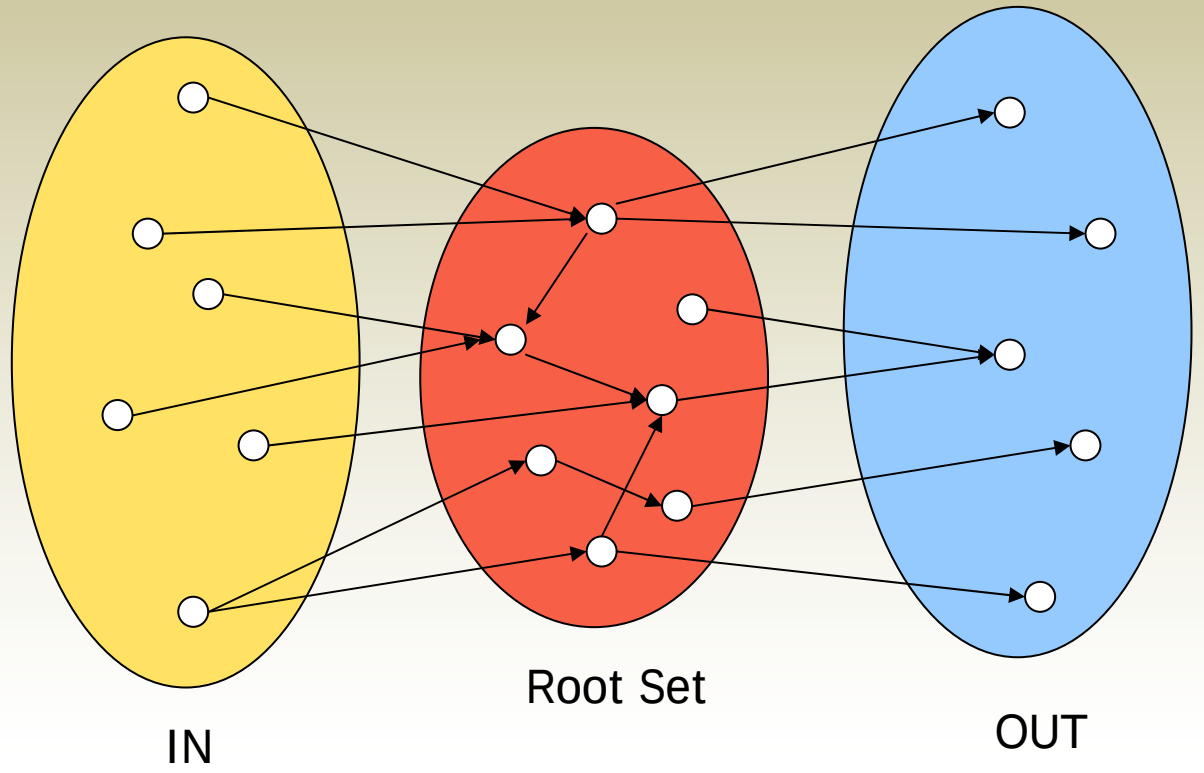
- Άρα, ορίζουν την επαναληπτική power μέθοδο για τον υπολογισμό των κυρίαρχων ιδιοδιανυσμάτων των πινάκων $\mathbf{L}^{T}\mathbf{L}$ και $\mathbf{L}\mathbf{L}^{T}$
- Είναι παρόμοια περίπτωση με τον υπολογισμό του PageRank, αλλά χρησιμοποιείται διαφορετικός πίνακας συντελεστών
- Ο πίνακας $\mathbf{L}^{T}\mathbf{L}$ λέγεται **πίνακας authority**, αφού καθορίζει τα authority scores
- Ο πίνακας $\mathbf{L}\mathbf{L}^{T}$ λέγεται **πίνακας hub**, αφού καθορίζει τα hub scores
- Και οι δυο πίνακας είναι συμμετρικοί, θετικοί και semidefinite

Υλοποίηση του HITS (1/5)

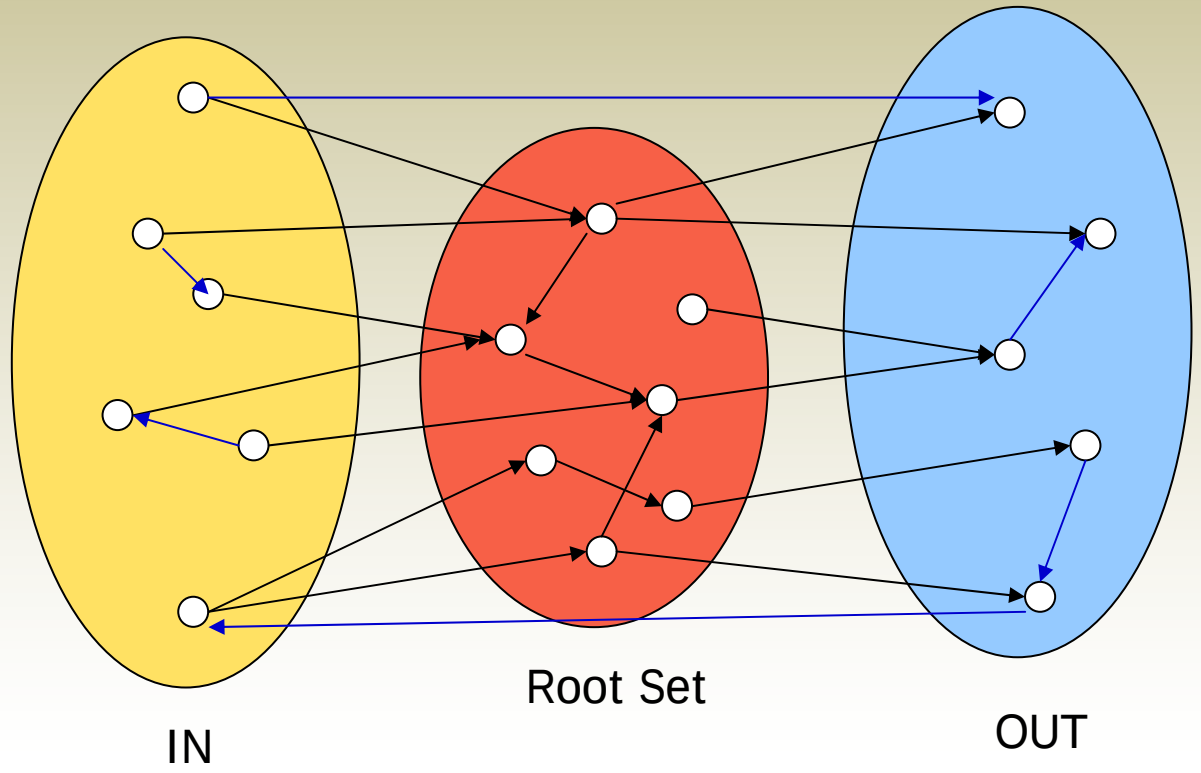


Root Set

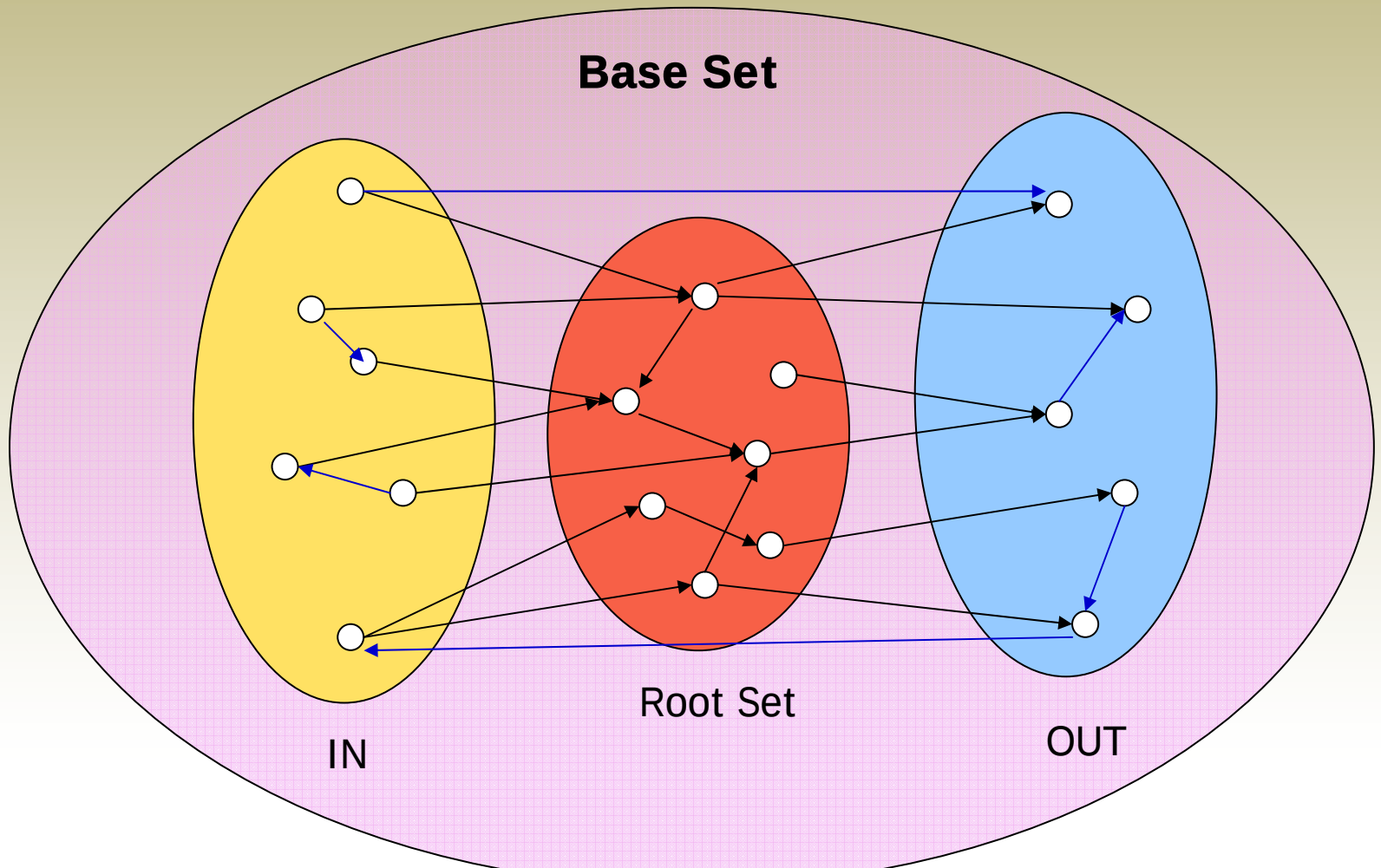
Υλοποίηση του HITS (2/5)



Υλοποίηση του HITS (3/5)



Υλοποίηση του HITS (4/5)



Υλοποίηση του HITS (5/5)

- Το Root Set βρίσκεται με τη βοήθεια μιας μηχανής αναζήτησης με keyword-search
- Το Base Set είναι ο Neighborhood Graph
- Ενσωματώνονται και τεχνικές σημασιολογικής συγγένειας στην επιλογή των σελίδων που θ' απαρτίζουν το Base Set
- Για να μην μεγαλώσει σε τεράστιο μέγεθος το Base Set, ορίζουμε έναν μέγιστο αριθμό κόμβων (με εισερχόμενους/εξερχόμενους) υπερσυνδέσμους για τον κάθε κόμβο του Root Set τους οποίους ενσωματώνουμε στο Base Set
- Δεν χρειάζεται να υπολογίσουμε το κυρίαρχο ιδιοδιάνυσμα και για τον πίνακα $\mathbf{L^T L}$ αλλά και για τον $\mathbf{L L^T}$, αλλά μόνο για τον ένα πίνακα, αφού ισχύει: $\mathbf{y = Lx}$

Σύγκλιση του HITS (1/6)

- Ο επαναληπτικός αλγόριθμος για τον υπολογισμό του HITS είναι συνήθως η power μέθοδος πάνω στους πίνακες $\mathbf{L}^T\mathbf{L}$ και $\mathbf{L}\mathbf{L}^T$
- Για έναν διαγωνιοποιήσιμο πίνακα $\mathbf{B}$ με διακριτές ιδιοτιμές $\{\lambda_1,\lambda_2,\ldots,\lambda_k\}$ όπου $|\lambda_1| > |\lambda_2| \geq |\lambda_3| \geq \ldots \geq |\lambda_k|$ η power μέθοδος υπολογίζει επαναληπτικά το εξής:

$$\mathbf{x}^{(k)} = \mathbf{B}\mathbf{x}^{(k-1)} \qquad \mathbf{x}^{(k)} \Longleftarrow \frac{\mathbf{x}^{(k)}}{m(\mathbf{x}^{(k)})}$$

  όπου $\mathrm{m}(\mathbf{x}^{(k)})$ είναι "σταθερά" κανονικοποίησης παραγόμενη από το $\mathbf{x}^{(k)}$
- Συνήθως, είναι η (προσημασμένη) συνιστώσα με το μέγιστο μέγεθος. Στην περίπτωση αυτή, το $\mathrm{m}(\mathbf{x}^{(k)})$ συγκλίνει στην κυρίαρχη ιδιοτιμή λ_1 και το $\mathbf{x}^{(k)}$ στο αντίστοιχο κανονικοποιημένο ιδιοδιάνυσμα

Σύγκλιση του HITS (2/6)

- Εάν απαιτείται μόνο το ιδιοδιάνυσμα και όχι και η αντίστοιχη ιδιοτιμή, τότε η "σταθερά" κανονικοποίησης μπορεί να είναι η $m(\mathbf{x}^{(k)}) = ||\mathbf{x}^{(k)}||$
- Εάν $\lambda_1 < 0$, τότε η $m(\mathbf{x}^{(k)}) = ||\mathbf{x}^{(k)}||$ δεν μπορεί να συγκλίνει στην λ_1, αλλά η $\mathbf{x}^{(k)}$ συγκλίνει στο ιδιοδιάνυσμα που αντιστοιχεί στην λ_1
- Επειδή οι πίνακες $\mathbf{L}^T\mathbf{L}$ και $\mathbf{L}\mathbf{L}^T$ είναι συμμετρικοί, θετικοί, semidefinite και μη-αρνητικοί, οι διακριτές τους ιδιοτιμές $\{\lambda_1,\lambda_2,\ldots,\lambda_k\}$ είναι πραγματικοί αριθμοί και μη-αρνητικοί, και ισχύει ότι με $\lambda_1 > \lambda_2 > \lambda_3 > \ldots > \lambda_k \geq 0$
- Έτσι, ο HITS με κανονικοποίηση πάντα συγκλίνει
- Ο ρυθμός σύγκλισης δίνεται από τον ρυθμό με τον οποίο ισχύει ότι $[\lambda_2(\mathbf{L}^T\mathbf{L})/\lambda_1(\mathbf{L}^T\mathbf{L})]^k \rightarrow 0$

Σύγκλιση του HITS (3/6)

- Δυστυχώς, δεν μπορούμε να δώσουμε ικανοποιητική προσέγγιση για τον ασυμπτωτικό ρυθμό σύγκλισης του HITS
- Πολλά πειράματα δείχνουν ότι η διαφορά των πρώτων ιδιοτιμών (λ_1-λ_2) είναι αρκετά μεγάλη, και συνεπώς απαιτούνται μόνο μερικές επαναλήψεις (10-15) για να συγκλίνει
- Παρά την ταχεία σύγκλιση όμως, υπάρχει πρόβλημα με τη μοναδικότητα των διανυσμάτων authority και hub που προκύπτουν ως λύση με την power μέθοδο, π.χ., για τον $\mathbf{L}$

$$\mathbf{L} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \end{pmatrix} \qquad \mathbf{L}^T\mathbf{L} = \begin{pmatrix} 2 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

Σύγκλιση του HITS (4/6)

- Έχει δυο διακριτές ιδιοτιμές, τις 2 και 0, οι οποίες έχουν πολλαπλότητα 2
- Εάν ξεκινήσουμε με το $\mathbf{x}^{(0)} = ¼\ \mathbf{e}^T$ καταλήγουμε στο διάνυσμα authority $\mathbf{x}^{(\infty)} = (1/3 \quad 1/3 \quad 1/3 \quad 0)^T$
- Εάν ξεκινήσουμε με το $\mathbf{x}^{(0)} = (1/4 \quad 1/8 \quad 1/8 \quad 1/2$ καταλήγουμε στο διάνυσμα authority $\mathbf{x}^{(\infty)} = (1/2 \quad 1/4 \quad 1/4 \quad 0)^T$
- Αιτία του προβλήματος μοναδικότητας είναι η reducibility
- Θα λέμε ότι ένας πίνακας **B** είναι reducible εάν υπάρχει πίνακας μετάθεσης **Q** τέτοιος ώστε (οι **X** και **Z** είναι τετραγωνικοί):

$$\mathbf{Q}^T\mathbf{B}\mathbf{Q} = \begin{pmatrix} \mathbf{X} & \mathbf{Y} \\ \mathbf{0} & \mathbf{Z} \end{pmatrix}$$

Σύγκλιση του HITS (5/6)

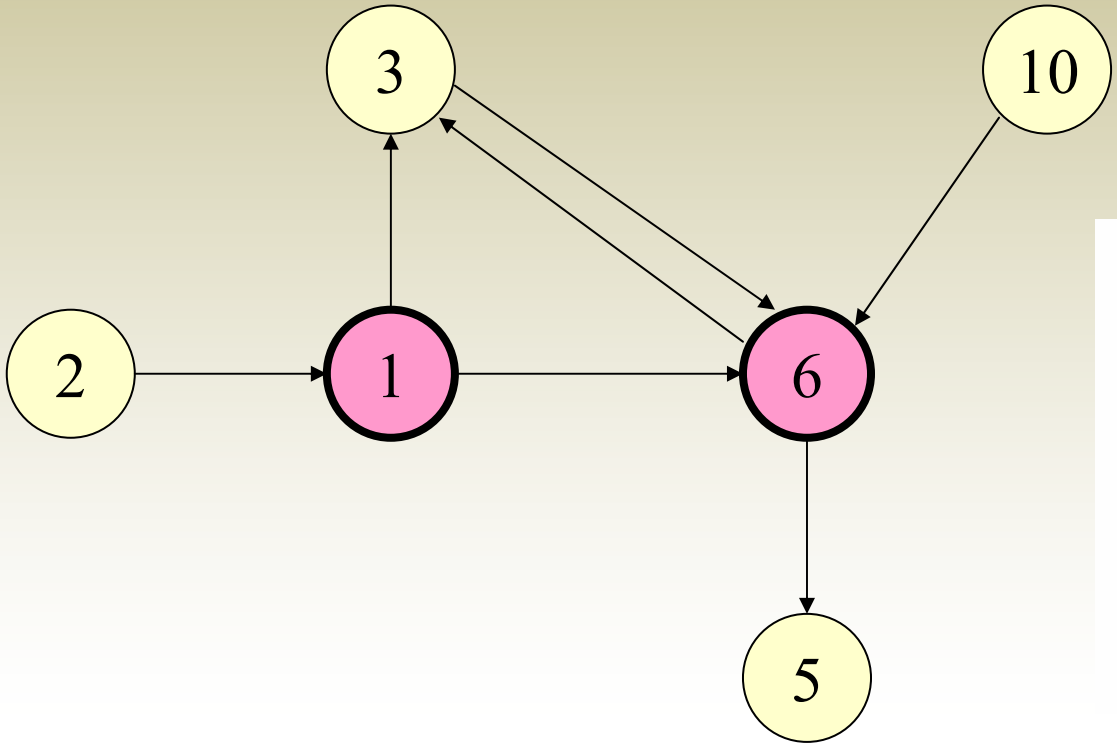
- Η reducibility ενός πίνακα σημαίνει ότι υπάρχουν καταστάσεις "καταβόθρες", ενώ η irreducibility σημαίνει ότι από οποιαδήποτε κατάσταση μπορώ να μεταβώ σε οποιαδήποτε άλλη κατάσταση
- Το θεώρημα Perron-Frobenious εγγυάται ότι ένας irreducible, μη-αρνητικός πίνακας έχει ένα μοναδικό, κανονικοποιημένο θετικό κυρίαρχο ιδιοδιάνυσμα, το λεγόμενο *διάνυσμα Perron*
- Συνεπώς, η reducibility του $\mathbf{L}^{\mathbf{T}}\mathbf{L}$ είναι υπεύθυνη για η σύγκλιση σε περισσότερα του ενός διανύσματα-λύσεις
- Το ίδιο πρόβλημα αντιμετώπισε και ο πίνακας S στο PageRank, αλλά με μια μετατροπή του πίνακα σε irreducible επιλύθηκε
- Το ίδιο μπορεί να γίνει και με τον HITS

Σύγκλιση του HITS (6/6)

- Αντί για τον αρχικό πίνακα authority, χρησιμοποιούμε τον πίνακα $\xi\mathbf{L}^{\mathbf{T}}\mathbf{L} + (1-\xi)/n\ \mathbf{e}\mathbf{e}^{\mathbf{T}}$
- Όμοια για τον πίνακα hub
- Αυτός ο τροποποιημένος HITS, λέγεται Exponential HITS
- Τέλος, ανεξάρτητα από το εάν η κυρίαρχη ιδιοτιμή του πίνακα επανάληψης **B** είναι απλή ή πολλαπλή, η σύγκλιση σε ένα μη-μηδενικό διάνυσμα εξαρτάται από εάν το αρχικό διάνυσμα $\mathbf{x}^{(0)}$ δεν βρίσκεται στην εμβέλεια του $(\mathbf{B}-\lambda_1\mathbf{I})$
- Εάν το $\mathbf{x}^{(0)}$ παράγεται τυχαία, τότε με (σχεδόν) βεβαιότητα δεν θα υφίσταται το πρόβλημα

Παράδειγμα εφαρμογής του HITS (1/4)

- Έστω ότι, σε απάντηση ενός ερωτήματος σε μια παραδοσιακή keyword-based μηχανή αναζήτησης, επιστρέφονται οι ιστοσελίδες που αντιστοιχούν στους κόμβους 1 και 6



$$\mathbf{L} = \begin{pmatrix} 0 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$

Παράδειγμα εφαρμογής του HITS (2/4)

$$\mathbf{L}^{\mathrm{T}}\mathbf{L} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 3 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \qquad \mathbf{L}\mathbf{L}^{\mathrm{T}} = \begin{pmatrix} 2 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 2 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 \end{pmatrix}$$

- Τα κανονικοποιημένα κύρια ιδιοδιανύσματα με τα authority και hub scores είναι:

$$\mathbf{x}^{\mathrm{T}} = (0 \quad 0 \quad 0.3660 \quad 0.1340 \quad 0.5 \quad 0)$$

$$\mathbf{y}^{\mathrm{T}} = (0.3660 \quad 0 \quad 0.2113 \quad 0 \quad 0.2113 \quad 0.2113)$$

Παράδειγμα εφαρμογής του HITS (3/4)

- Μπορεί να συμβούν δυο τύπων ισοπαλίες
  - Στις τιμές 0
    - Μπορεί να αποφευχθούν με την primitivity τροποποίηση
  - Στις θετικές τιμές
    - αυτές είναι σπάνιες σε μεγάλους θετικά-ορισμένους πίνακες
    - μπορούν να επιλυθούν με FCFS
- Authority ranking = (6 3 5 1 2 10)
- Hub ranking = (1 3 6 10 2 5)
- Για λόγους σύγκρισης, υπολογίζουμε ξανά τα διανύσματα authority και hub, αλλά χρησιμοποιώντας τον irreducible πίνακα $\xi\mathbf{L}^T\mathbf{L} + (1-\xi)/n\ \mathbf{e}\mathbf{e}^T$ ως πίνακα authority και τον irreducible πίνακα $\xi\mathbf{L}\mathbf{L}^T + (1-\xi)/n\ \mathbf{e}\mathbf{e}^T$ ως πίνακα hub

Παράδειγμα εφαρμογής του HITS (4/4)

- Για ξ=0.95, έχουμε

$$\mathbf{x}^T = (0.0032 \quad 0.0023 \quad 0.3634 \quad 0.1351 \quad 0.4936 \quad 0.0023)$$
$$\mathbf{y}^T = (0.3628 \quad 0.0032 \quad 0.2106 \quad 0.0023 \quad 0.2106 \quad 0.2106)$$

- Στο παράδειγμα αυτό, η μετατροπή τους σε irreducible πίνακες δεν άλλαξε το ranking, ούτε το authority ούτε το hub ranking
- Όμως, απέφυγε τις ισοπαλίες στο 0

Πλεονεκτήματα/Μειονεκτήματα του HITS

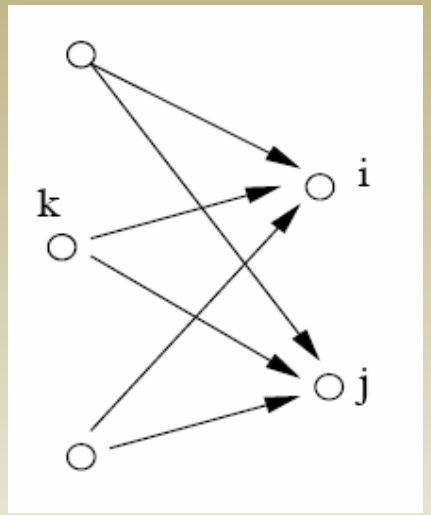
- Παρουσιάζει δυο λίστες στο χρήστη
  - Authoritative σελίδες, για εις βάθος αναζήτηση σε κάποιο αντικείμενο
  - Hub σελίδες, δηλ., portal σελίδες, για αναζήτηση σε εύρος
- Λύνει ένα μικρό πρόβλημα, σε μέγεθος πινάκων
- Είναι query-dependent
  - Σε run-time, δηλ., για κάθε ερώτημα του χρήστη, χτίσιμο του Base Set, και εύρεση ενός ιδιοδιανύσματος
- Μπορεί να γίνει query-independent, δηλ., να εκτελεστεί πάνω σε όλο το γράφημα του Web
- Είναι πολύ επιρρεπής σε spamming
  - Με προσθήκη συνδέσμων από/προς την ιστοσελίδα μας
  - Φυσικά, είναι ευκολότερο να αυξήσουμε το hub score της ιστοσελίδας μας, αλλά εξ' αιτίας της αλληλεξάρτησής τους μπορεί να αυξηθεί και το authority score ως αποτέλεσμα της αύξησης του hub score

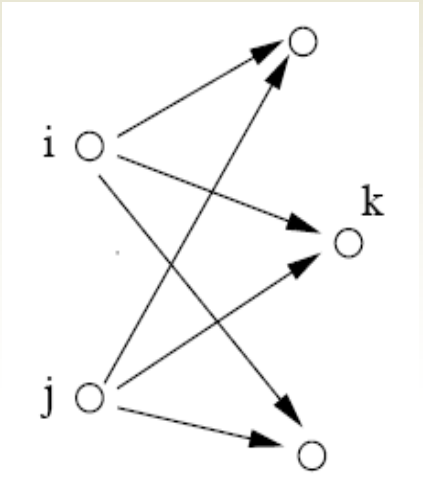
Πλεονεκτήματα/Μειονεκτήματα του HITS

- Παρουσιάζει το φαινόμενο **topic drift**
  - Κατά το χτίσιμο του Base Set, είναι πιθανό ότι μια πολύ authoritative σελίδα, αλλά εκτός αντικειμένου της αναζήτησης, να εμφανιστεί στο Base Set επειδή έχει σύνδεσμο από/προς κάποια σελίδα του Root Set
  - Αυτή η σελίδα μπορεί να έχει τόσο βάρος, ώστε αυτή και όλες οι "γειτονικές" της να εμφανίζονται στην κορυφή της λίστας των αποτελεσμάτων, με συνέπεια η λίστα των αποτελεσμάτων να κυριαρχηθεί από σελίδες εκτός του ζητούμενου αντικειμένου
  - Το πρόβλημα μπορεί να αντιμετωπιστεί εάν "ζυγίσουμε" τα hub και authority scores των σελίδων με βάση τη σχετικότητά της κάθε σελίδας ως προς το αντικείμενο της αναζήτησης, δηλ., αντί για τον δυαδικό πίνακα **L**, να έχουμε κάτι ανάλογο με τον πίνακα του intelligent surfer

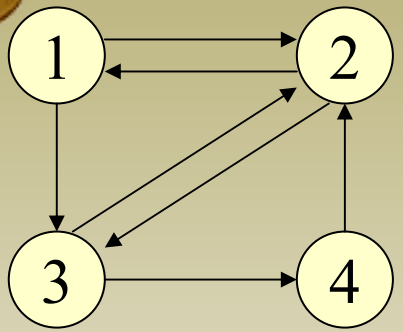
Σχέση του HITS με τη Βιβλιομετρία (1/2)

- **Co-citation**: Δυο σελίδες δέχονται υπερσύνδεσμο από την ίδια σελίδα
- Ο πίνακας authority $\mathbf{L}^T\mathbf{L}$ σχετίζεται με την έννοια του co-citation
- Ranking με βάση το inlink παρέχει αξιοπρεπή προσέγγιση του HITS authority
- **Co-reference**: Δυο σελίδες έχουν υπερσύνδεσμο προς την ίδια σελίδα
- Ο πίνακας hub $\mathbf{L}\mathbf{L}^T$ σχετίζεται με την έννοια του co-reference
- Ranking με βάση το outlink παρέχει αξιοπρεπή προσέγγιση του HITS hub





Σχέση του HITS με τη Βιβλιομετρία (2/2)



$$\mathbf{L} = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 \end{pmatrix}$$

$$\mathbf{L}^{\mathrm{T}}\mathbf{L} = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 3 & 1 & 1 \\ 1 & 1 & 2 & 0 \\ 0 & 1 & 0 & 1 \end{pmatrix} = \mathbf{D}_{in} + \mathbf{C}_{cit}$$

$$\mathbf{L}\mathbf{L}^{\mathrm{T}} = \begin{pmatrix} 2 & 1 & 1 & 1 \\ 1 & 2 & 0 & 0 \\ 1 & 0 & 2 & 1 \\ 1 & 0 & 1 & 1 \end{pmatrix} = \mathbf{D}_{out} + \mathbf{C}_{ref}$$

- $\mathbf{D}_{\mathbf{in}}$: διαγώνιος πίνακας με το indegree των κόμβων και $\mathbf{C}_{\mathbf{cit}}$ ο πίνακας co-citation
  - Το στοιχείο (3,3) δείχνει ότι ο κόμβος 3 έχει indegree 2
  - Το στοιχείο (4,3) δείχνει ότι οι κόμβοι 4 και 3 δεν έχουν κοινό inlink
- $\mathbf{D}_{\mathbf{out}}$: διαγώνιος πίνακας με το outdegree των κόμβων και $\mathbf{C}_{\mathbf{ref}}$ ο πίνακας co-reference
  - Το στοιχείο (1,2) δείχνει ότι οι κόμβοι 1 και 2 έχουν 1 κοινό σύνδεσμο (προς τον κόμβο 3)
  - Το στοιχείο (4,2) δείχνει ότι οι κόμβοι 4 και 2 δεν έχουν κοινό σύνδεσμο προς κάποιο κόμβο

Query-independent HITS (1/4)

Ο query-independent αλγόριθμος HITS

- Αρχικοποίηση του $\mathbf{x}^{(0)}=\mathbf{e}/n$ (και άλλες επιλογές αρχικοποίησης είναι πιθανές)
- Μέχρι να επέλθει σύγκλιση:
  - $\mathbf{x}^{(k)} = \xi \mathbf{L}^T \mathbf{L} \mathbf{x}^{(k-1)} + (1-\xi)/n\ \mathbf{e}$
  - $\mathbf{x}^{(k)} = \mathbf{x}^{(k)} / \|\mathbf{x}^{(k)}\|_1$
  - $\mathbf{y}^{(k)} = \xi \mathbf{L}\mathbf{L}^T \mathbf{y}^{(k-1)} + (1-\xi)/n\ \mathbf{e}$
  - $\mathbf{y}^{(k)} = \mathbf{y}^{(k)} / \|\mathbf{y}^{(k)}\|_1$
  - ++k;
  - Θέτουμε $\mathbf{x} = \mathbf{x}^{(k)}$ και $\mathbf{y} = \mathbf{y}^{(k)}$

- Συγκλίνει στα μοναδικά θετικά διανύσματα hub και authority, ανεξάρτητα από τη reducibility του πίνακα
- Ο **L** είναι ο πίνακας γειτνίασης όλου του Web γραφήματος

Query-independent HITS (2/4)

Μέθοδος	Πολλαπλασιασμοί	Προσθέσεις
HITS	0	2nnz(**L**)
Modified HITS	0	4nnz(**L**) + 2n
Random surfer PageRank	n	nnz(**L**) + n
Intelligent surfer PageRank	nnz(**H**)	nnz(**H**) + n

- **ΘΕΩΡΗΜΑ.** Έστω ότι M = είναι ο τροποποιημένος πίνακας authority. Έστω ότι $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_n$ είναι οι ιδιοτιμές του $\mathbf{L}^T\mathbf{L}$ και $\gamma_1 \geq \gamma_2 \geq \ldots. \geq \gamma_n$ είναι οι ιδιοτιμές του M. Τότε ισχύει η σχέση:

$$\gamma_1 \geq \alpha\lambda_1 \geq \gamma_2 \geq \alpha\lambda_2 \geq \ldots. \geq \geq \gamma_n \geq \alpha\lambda_n$$

Υπάρχουν scalars $\beta_i \geq 0$, $\sum\beta_i = 1$, ώστε $\gamma_i = \xi\lambda_i + (1-\xi)\beta_i$

Query-independent HITS (3/4)

- Τα όρια γ_2/γ_1 παράγονται εξετάζοντας ακραία συμπεριφορά
- Στην καλύτερη περίπτωση, η τροποποίηση του $\mathbf{L}^T\mathbf{L}$ αυξάνει μόνο το λ_2 σε μέγιστη τιμή $\lambda_2+1-\xi$ (δηλ., $\beta_2=1$, $\beta_i=0$, $i\neq 2$)
- Στη χειρότερη περίπτωση, το λ_1 αυξάνει σε μέγιστη τιμή $\lambda_1+1-\xi$ (δηλ., $\beta_1=1$, $\beta_i=0$, $i\neq 1$)
- Στην πράξη, πολλά β_i αυξάνουν ταυτόχρονα, αλλά η σχέση $\Sigma\beta_i=1$ εγγυάται ότι το αποτέλεσμα δεν έχει δραματικές επιπτώσεις, όπως στις δυο ακραίες περιπτώσεις

Μέθοδος	**Σύγκλιση**
HITS	λ_2/λ_1
Modified HITS	$(\xi\lambda_2)/(\xi\lambda_1+1-\xi) \leq \gamma_2/\gamma_1 \leq \lambda_2/\lambda_1 \leq (1-\xi)/(\xi\lambda_1)$
Random surfer PageRank	α
Intelligent surfer PageRank	α

Query-independent HITS (4/4)

- Ανεξάρτητα από τις ακριβείς τιμές των β_i, το ξ επιλέγεται συνήθως κοντά στο 1, οπότε $\gamma_2/\gamma_1 \approx \lambda_2/\lambda_1$
- Έτσι ο ασυμπτωτικός ρυθμός σύγκλισης του HITS και του τροποποιημένου HITS είναι ο ίδιος
- Πειράματα έχουν δείξει ότι $\lambda_2/\lambda_1 < 0.5$, το οποίο είναι πολύ μικρότερο του α=0.85 του PageRank
- Με περίπου διπλάσιο κόστος ανά επανάληψη, ο query-independent-HITS απαιτεί λιγότερο από ¼ των επαναλήψεων του PageRank και παράγει 2 διανύσματα στο χρήστη

Επιτάχυνση του HITS

- Ο Kleinberg χρησιμοποίησε την power μέθοδο για τον υπολογισμό των κυρίαρχων δεξιών ιδιοδιανυσμάτων των πινάκων $\mathbf{L^T L}$ και $\mathbf{L L^T}$
- Οι πίνακες του HITS είναι πολύ μικροί σε σχέση με τους αντίστοιχους του PageRank, και κατά πάσα πιθανότητα, χρησιμοποιούνται τεχνικές που είναι memory-intensive για τον υπολογισμό των δυο πινάκων του HITS, π.χ., Lanczos
- Για τον query-dependent HITS δεν υπάρχει έρευνα σχετική με μεθόδους επιτάχυνσης
- Για τον query-independent HITS μπορούν να χρησιμοποιηθούν οι ίδιες τεχνικές που έχουμε συζητήσει για τον PageRank

Ευαισθησία του HITS

- **ΘΕΩΡΗΜΑ**. Έστω **E** ο πίνακας διαταραχής, ώστε $\acute{\mathbf{L}}^{\mathbf{T}}\acute{\mathbf{L}} = \mathbf{L}^{\mathbf{T}}\mathbf{L} + \mathbf{E}$. Όταν η λ_1 είναι απλή, τότε

$$\sin \angle(\mathbf{x},\mathbf{x'}) \leq ||\mathbf{E}||_2/(\lambda_1-\lambda_2)$$

- Συνεπώς, εάν το χάσμα ιδιοδιανυσμάτων είναι μεγάλο, τότε ο πίνακας authority δεν είναι ευαίσθητος σε μικρές αλλαγές στο Web γράφημα

Η μέθοδος SALSA

Stochastic Approach for Link Structure Analysis

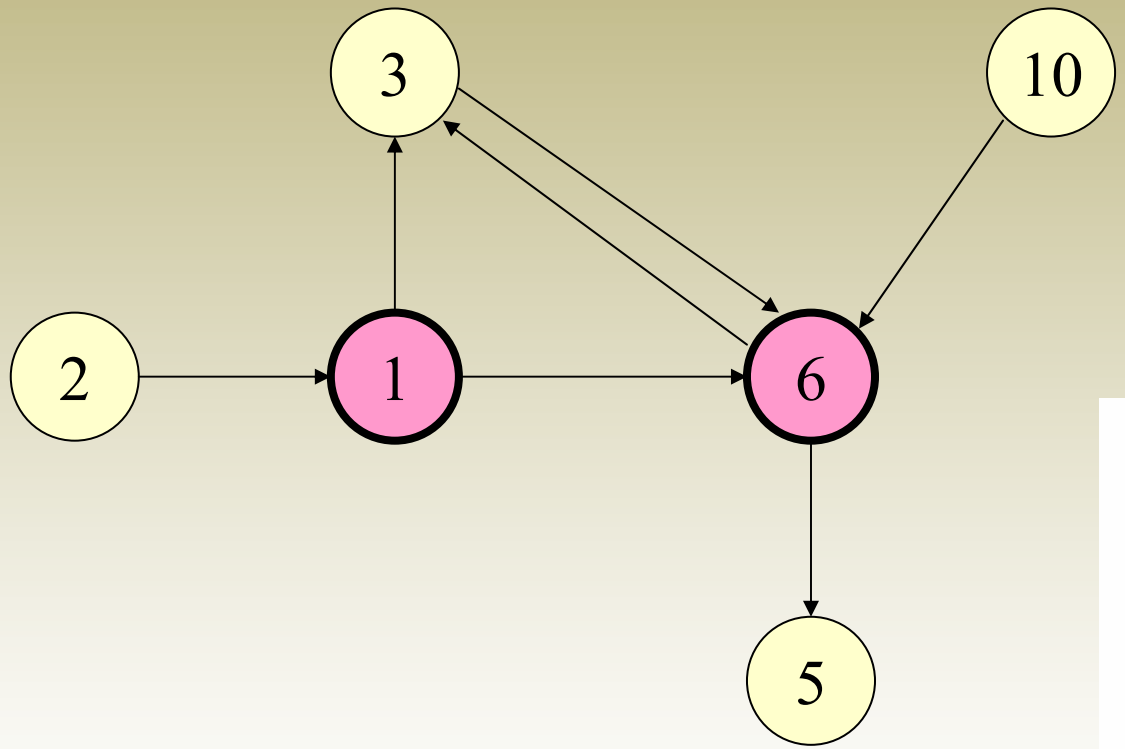
Ομοιότητες SALSA με HITS και PageRank

- Επινοήθηκε από τους Ronny Lempel και Shlomo Moran το 2000
- Συνδυασμός HITS και PageRank
- Ο SALSA χρησιμοποιεί authority και hub score
- Ο SALSA δημιουργεί ένα neighborhood graph χρησιμοποιώντας authority και hub ιστοσελίδες και υπερσυνδέσμους

Διαφορές SALSA με HITS και PageRank

- Η μέθοδος SALSA δημιουργεί ένα διμερές γράφημα (bipartite graph) των σελίδων authority και hub στο neighborhood γράφημα
- Το ένα σύνολο περιέχει τις hub σελίδες
- Το άλλο σύνολο περιέχει τις authority σελίδες
- Μια σελίδα μπορεί να περιέχεται και στα δυο σύνολα

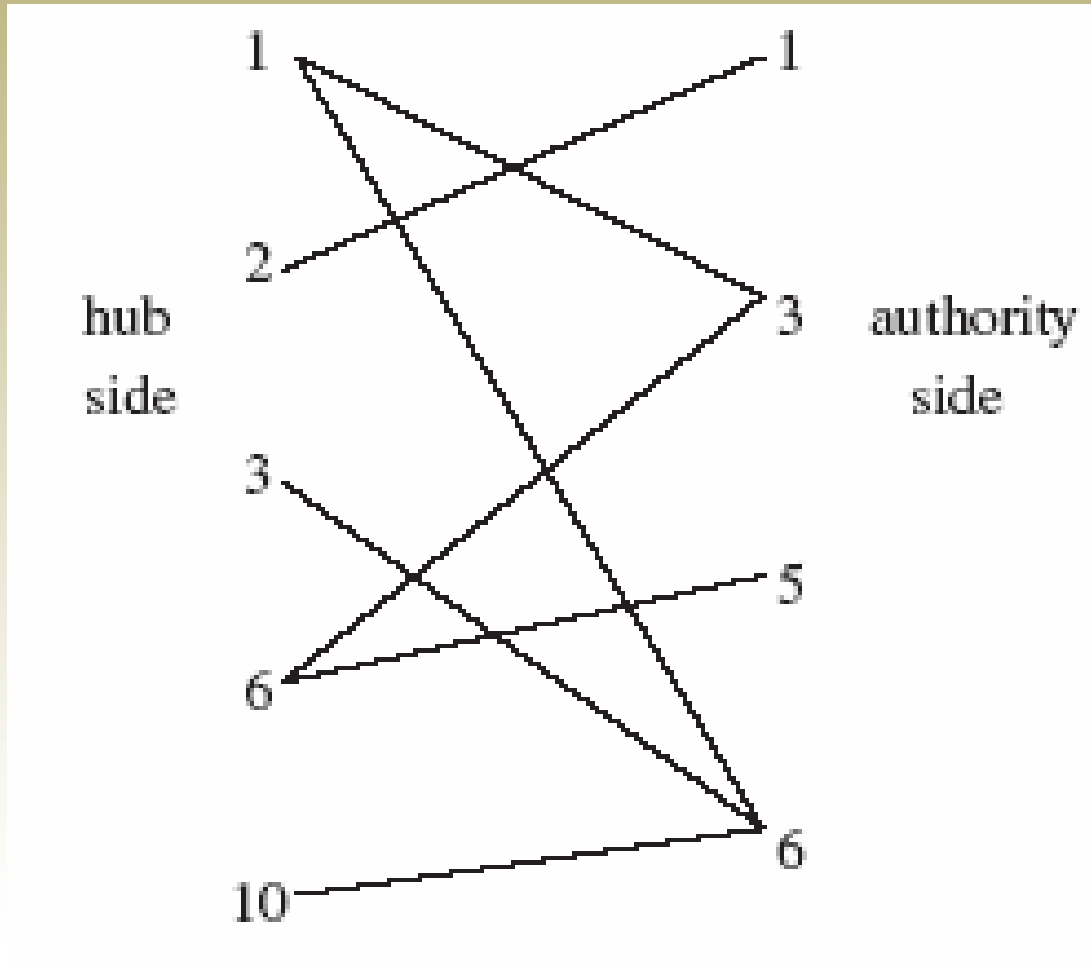
Neighborhood Graph N



$$\mathbf{L} = \begin{pmatrix} 0 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$

Διμερές γράφημα G του Neighborhood Graph N

$$V_h = \{1, 2, 3, 6, 10\},$$
$$V_a = \{1, 3, 5, 6\}.$$



Markov αλυσίδες

- Από το διμερές γράφημα G σχηματίζονται δυο πίνακες
  - Μια hub Markov chain με πίνακα **H**
  - Μια authority Markov chain με πίνακα **A**
- Οι πίνακες **H** και **A** μπορεί να παραχθούν από τον πίνακα γειτνίασης (adjacency matrix) **L** που έχουμε δει στον υπολογισμό του HITS και του PageRank
- Ο HITS χρησιμοποιεί τον unweighted matrix **L**
- Ο PageRank χρησιμοποιεί τη row-weighted έκδοση του πίνακα **L**
- SALSA χρησιμοποιεί row και column weighting

Πώς υπολογίζονται οι **H** και **A**;

- Έστω ότι $\mathbf{L}_r$ είναι ο **L** με κάθε μη-μηδενική γραμμή του να διαιρείται με το άθροισμά της
- Έστω ότι $\mathbf{L}_c$ είναι ο **L** με κάθε μη-μηδενική στήλη του να διαιρείται με το άθροισμά της

Παράδειγμα των $\mathbf{L}_r$ και $\mathbf{L}_c$

$$\mathbf{L} = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 5 \\ 6 \\ 10 \end{array} \begin{array}{c} \begin{array}{cccccc} 1 & 2 & 3 & 5 & 6 & 10 \end{array} \\ \begin{pmatrix} 0 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix} \end{array}, \qquad \mathbf{L}_r = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 5 \\ 6 \\ 10 \end{array} \begin{array}{c} \begin{array}{cccccc} 1 & 2 & 3 & 5 & 6 & 10 \end{array} \\ \begin{pmatrix} 0 & 0 & \frac{1}{2} & 0 & \frac{1}{2} & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{2} & \frac{1}{2} & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix} \end{array},$$

and

$$\mathbf{L}_c = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 5 \\ 6 \\ 10 \end{array} \begin{array}{c} \begin{array}{cccccc} 1 & 2 & 3 & 5 & 6 & 10 \end{array} \\ \begin{pmatrix} 0 & 0 & \frac{1}{2} & 0 & \frac{1}{3} & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{3} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{2} & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{3} & 0 \end{pmatrix} \end{array}.$$

Οι πίνακες **H** και **A**

- Ο πίνακας **H**, SALSA hub matrix, αποτελείται από τις μη-μηδενικές γραμμές και στήλες του $\mathbf{L}_r\mathbf{L}_c^T$

- Ο πίνακας **A**, SALSA authority matrix, αποτελείται από τις μη-μηδενικές γραμμές και στήλες του πίνακα $\mathbf{L}_c^T\mathbf{L}_r$

Οι πίνακες $\mathbf{L}_r\mathbf{L}_c{}^T$ και $\mathbf{L}_c{}^T\mathbf{L}_r$

$$\mathbf{L}_r\mathbf{L}_c^T = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 5 \\ 6 \\ 10 \end{array} \begin{array}{c} \begin{array}{cccccc} 1 & 2 & 3 & 5 & 6 & 10 \end{array} \\ \begin{pmatrix} \frac{5}{12} & 0 & \frac{2}{12} & 0 & \frac{3}{12} & \frac{2}{12} \\ 0 & 1 & 0 & 0 & 0 & 0 \\ \frac{1}{3} & 0 & \frac{1}{3} & 0 & 0 & \frac{1}{3} \\ 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{1}{4} & 0 & 1 & 0 & \frac{3}{4} & 0 \\ \frac{1}{3} & 0 & \frac{1}{3} & 0 & 0 & \frac{1}{3} \end{pmatrix} \end{array},$$

$$\mathbf{L}_c^T\mathbf{L}_r = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 5 \\ 6 \\ 10 \end{array} \begin{array}{c} \begin{array}{cccccc} 1 & 2 & 3 & 5 & 6 & 10 \end{array} \\ \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{2} & \frac{1}{4} & \frac{1}{4} & 0 \\ 0 & 0 & \frac{1}{2} & \frac{1}{2} & 0 & 0 \\ 0 & 0 & \frac{1}{6} & 0 & \frac{5}{6} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \end{array}.$$

Οι πίνακες **H** και **A**

$$\mathbf{H} = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 6 \\ 10 \end{array} \begin{array}{c} \begin{array}{ccccc} 1 & 2 & 3 & 6 & 10 \end{array} \\ \begin{pmatrix} \frac{5}{12} & 0 & \frac{2}{12} & \frac{3}{12} & \frac{2}{12} \\ 0 & 1 & 0 & 0 & 0 \\ \frac{1}{3} & 0 & \frac{1}{3} & 0 & \frac{1}{3} \\ \frac{1}{4} & 0 & 0 & \frac{3}{4} & 0 \\ \frac{1}{3} & \frac{1}{3} & 0 & 0 & \frac{1}{3} \end{pmatrix} \end{array}.$$

$$\mathbf{A} = \begin{array}{c} \\ 1 \\ 3 \\ 5 \\ 6 \end{array} \begin{array}{c} \begin{array}{cccc} 1 & 3 & 5 & 6 \end{array} \\ \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \frac{1}{2} & \frac{1}{4} & \frac{1}{4} \\ 0 & \frac{1}{2} & \frac{1}{2} & 0 \\ 0 & \frac{1}{6} & 0 & \frac{5}{6} \end{pmatrix} \end{array}.$$

Ιδιοδιανύσματα

- $Av = \lambda v$
- $v^T A = \lambda\, v^T$
- Αριθμητικός υπολογισμός: Power μέθοδος

Η power μέθοδος

- $X_{k+1} = AX_k$
- $X_{k+1}^T = X_k^T A$
- Συγκλίνει στο κυρίαρχο ιδιοδιάνυσμα (dominant eigenvector), δηλ., σ' αυτό που αντιστοιχεί στην κυρίαρχη ιδιοτιμή ($\lambda = 1$).

Η power μέθοδος

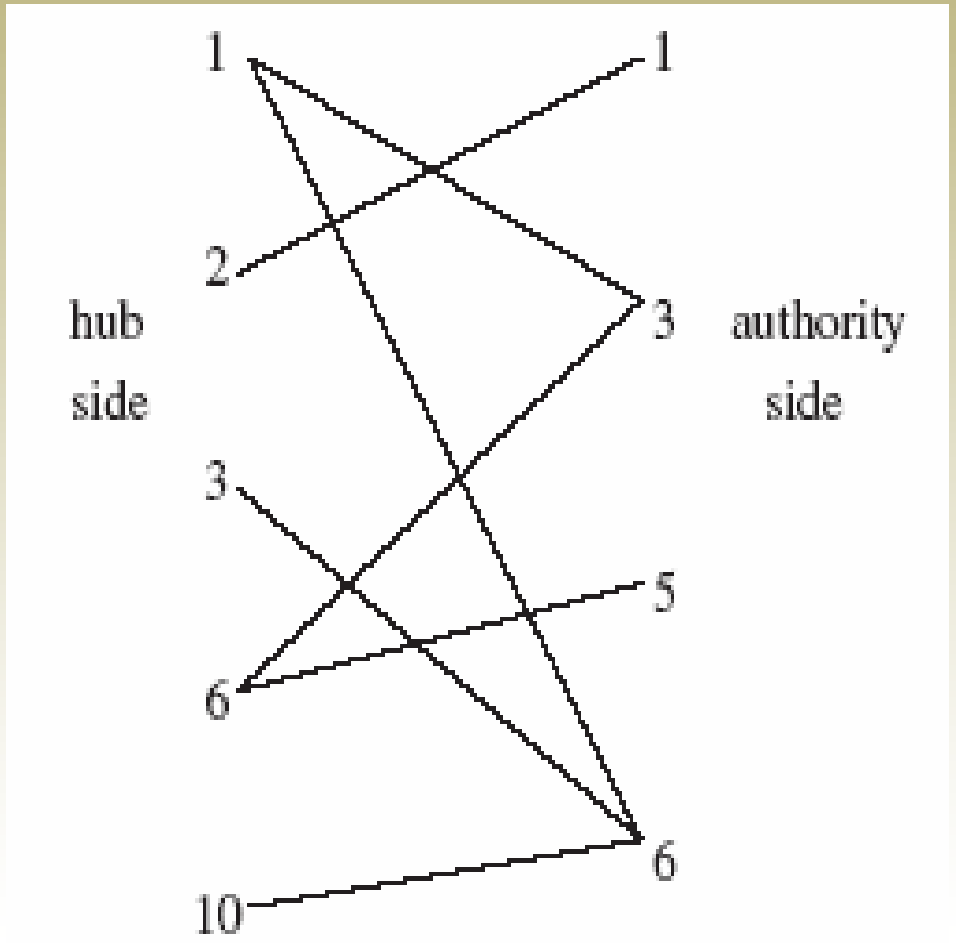
- Οι πίνακες **H** και **A** πρέπει να είναι irreducible, ώστε να συγκλίνει η power μέθοδος σε ένα unique eigenvector, ξεκινώντας από κάποια αρχική τιμή

- Εάν το neighborhood γράφημα **G** είναι συνδεδεμένο (connected), τότε ο **H** και ο **A** είναι irreducible

- Εάν το **G** δεν είναι συνδεδεμένο, τότε η εκτέλεση της power μεθόδου στον **H** και **A** δεν θα έχει ως αποτέλεσμα τη σύγκλιση σε ένα μοναδικό κυρίαρχο ιδιοδιάνυσμα

Στο παράδειγμα το G δεν είναι συνδεδεμένο

- Είναι προφανές ότι το γράφημα δεν είναι συνδεδεμένο, αφού η σελίδα 2 στο σύνολο hub συνδέεται μόνο με τη σελίδα 1 στο σύνολο authority, και αντίστροφα
- Οι **H** και **A** είναι reducible και επομένως περιέχουν multiple irreducible connected components



Connected Components

- Ο πίνακας **H** περιέχει δυο connected components, C = {2} και D = {1, 3, 6, 10}

- Ο πίνακας **A** περιέχει δυο connected components, E = {1} και F = {3, 5, 6}

Cutting και Pasting. Μέρος I

- Εκτελούμε την power method σε κάθε συνιστώσα των **H** και **A**

$$\pi_h^T(C) = \begin{matrix} 2 \\ (1), \end{matrix} \quad \pi_h^T(D) = \begin{matrix} 1 & 3 & 6 & 10 \\ (\frac{1}{3} & \frac{1}{6} & \frac{1}{3} & \frac{1}{6}), \end{matrix}$$

$$\pi_a^T(E) = \begin{matrix} 1 \\ (1), \end{matrix} \quad \pi_a^T(F) = \begin{matrix} 3 & 5 & 6 \\ (\frac{1}{3} & \frac{1}{6} & \frac{1}{2}). \end{matrix}$$

Cutting και Pasting. Μέρος II

- Ενώνουμε τις δυο συνιστώσες για κάθε πίνακα
- Πρέπει να πολλαπλασιάσουμε κάθε στοιχείο του διανύσματος με το κατάλληλο weight

H:

$$\pi_h^T = \begin{matrix} 1 & 2 & 3 & 6 & 10 \\ (\frac{4}{5}\cdot\frac{1}{3} & \frac{1}{5}\cdot 1 & \frac{4}{5}\cdot\frac{1}{6} & \frac{4}{5}\cdot\frac{1}{3} & \frac{4}{5}\cdot\frac{1}{6}) \end{matrix}$$
$$= \begin{matrix} 1 & 2 & 3 & 6 & 10 \\ (.2667 & .2 & .1333 & .2667 & .1333). \end{matrix}$$

A:

$$\pi_h^T = \begin{matrix} 1 & 3 & 5 & 6 \\ (\frac{1}{4}\cdot 1 & \frac{3}{4}\cdot\frac{1}{3} & \frac{3}{4}\cdot\frac{1}{6} & \frac{3}{4}\cdot\frac{1}{2}) \end{matrix}$$
$$= \begin{matrix} 1 & 3 & 5 & 6 \\ (.25 & .25 & .125 & .375). \end{matrix}$$

SALSA hub ranking: 1/6 2 3/10

HITS hub ranking: 1 3/6/10 2 5

SALSA authority ranking: 6 1/3 5

HITS hub ranking: 6 3 5 1 2/10

Πλεονεκτήματα/Μειονεκτήματα του

- Δεν εμφανίζει το φαινόμενο topic drift, όπως ο HITS
- Παρέχει authority και hub scores
- Χειρίζεται το spamming καλύτερα από ότι ο HITS, αλλά όχι τόσο καλά όσο ο PageRank
- Είναι query-dependent

Η μέθοδος BrowseRank

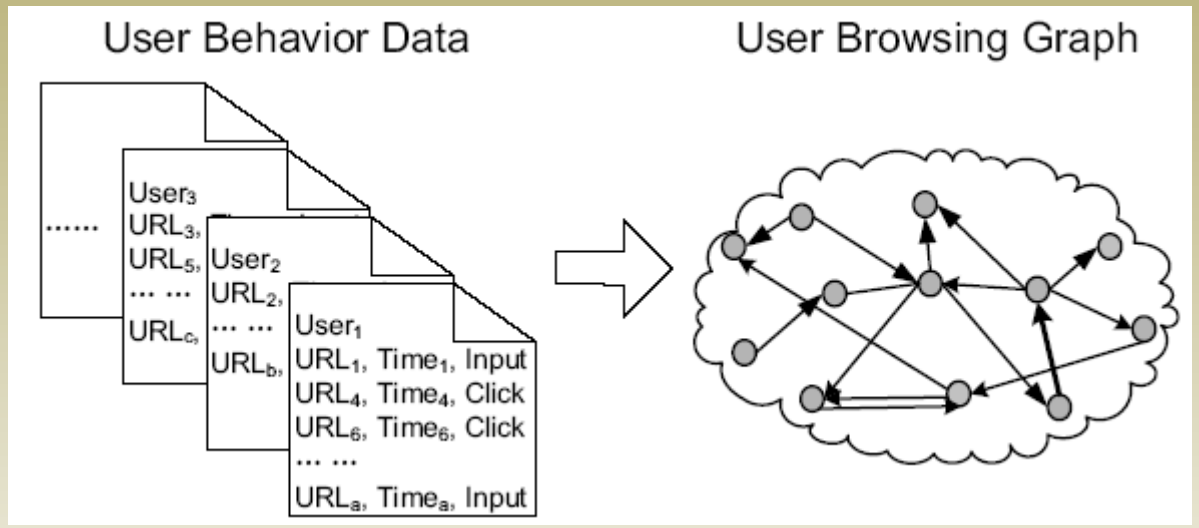
Introduction

- *Page importance*, which represents the 'value' of an individual page on the web, is a key factor for web search, because for contemporary search engines, the crawling, indexing, and ranking are usually guided by this measure
- Currently, page importance is calculated by using the *link graph* of the web and such a process is called link analysis
- Well known link analysis algorithms include *HITS* and *PageRank*

Google's PageRank

- PageRank employs a discrete-time Markov process on the web link graph to compute page importance, which in fact simulates a random walk along the hyperlinks on the web of a web surfer
- PageRank limitations
  - The link graph, which PageRank relies on, is not a very reliable data source, because hyperlinks on the web can be easily added or deleted by web content creators
  - PageRank only models a random walk on the link graph, but does not take into consideration the lengths of time which the web surfer spends on the web pages during the random walk

User Browsing Graph



- **<u>Can find a better data source than the link graph?</u>**
- Utilize the user browsing graph, generated from user behavior data
- User behavior data can be recorded by Internet browsers at web clients and collected at a web server

Continuous-time Markov chain

- **What kind of algorithm we should use to leverage the new data source?**
- The use of a discrete-time Markov process would not be sufficient
- Define a continuous-time Markov process as the model on the
- user browsing graph
- Assume the process to be time-homogenous
- The stationary probability distribution of the process can be used
- to define the importance of web pages
- Employ *BrowseRank*, to efficiently compute the stationary probability distribution of the continuous-time Markov process
- Make use of an additive noise model to represent the observations with regard to the Markov process and to conduct an unbiased and consistent estimation of the parameters in the process
- Adopt an embedded Markov chain based technology to speed up the calculation of the stationary distribution.

User Behavior Data

URL	TIME	TYPE
http://aaa.bbb.com/	2007-04-12, 21:33:05	INPUT
http://aaa.bbb.com/1.htm	2007-04-12, 21:34:11	CLICK
http://ccc.ddd.org/index.htm	2007-04-12, 21:34:52	CLICK
http://eee.fff.edu/	2007-04-12, 21:39:03	INPUT
...	...	...

- The user behavior data can be recorded and represented in triples consisting of <URL, TIME, TYPE>
- From the data extract transitions of users from page to page and the time spent by users on the pages as follows:
  - Session segmentation (break by: time rule & type rule)
  - URL pair construction
  - Reset probability estimation
  - Staying time extraction

Staying time extraction

- For each URL pair, we use the difference between the time of the second page and that of the first page as the observed staying time on the first page
- For the last page in a session, we use the following heuristics to decide its observed staying time
  - If the session is segmented by the *time rule*, we ***randomly*** (!?) sample a time from the distribution of observed staying time of pages in all the records and take it as the observed staying time
  - If the session is segmented by the *type rule*, we use the difference between the time of the last page in the session and that of the first page of the next session (INPUT page) as the staying time

Building a user browsing graph

- Each vertex in the graph represents a URL in the user behavior data, associated with
  - reset probability, and
  - staying time as metadata
- Each directed edge represents the transition between two vertices, associated with the number of transitions as its weight
- In other words, the user browsing graph is a weighted graph with vertices containing metadata and edges containing weights

Assumptions

- Independence of users and sessions
  - The browsing processes of different users in different sessions are independent. In other words, we treat web browsing as a stochastic process, with the data observed in each session by a user as an I.I.D. sample of this process
- Markov property
  - The page that a user will visit next only depends on the current page, and is independent of the pages she visited previously
  - This assumption is also a basic assumption in PageRank
- Time-homogeneity
  - The browsing behaviors of users (e.g. transitions and staying time) do not depend on time points. Although this assumption is not necessarily true in practice, it is mainly for technical convenience
  - This assumption is also a basic assumption in PageRank

Continuous-time Markov Model

- Suppose there is a web surfer walking through all the webpages
- We use X_s to denote the page which the surfer is visiting at time s, s>0
- Then, with the aforementioned three assumptions, the process $X = \{X_s, s \geq 0\}$ forms a continuous-time time-homogenous Markov process
- Let $p_{ij}(t)$ denotes the transition probability from page i to page j for time interval t in this process (also referred to as time increment in statistics)
- One can prove that there is a stationary probability distribution π, which is unique and independent of t, associated with $P(t) = [p_{ij}(t)]_{NXN}$, such that for any $t > 0$

$$\pi = \pi P(t)$$

- The i^{th} entry of the distribution π stands for the ratio of the time the surfer spends on the i^{th} page over the time she spends on all the pages when time interval t goes to infinity
- In this regard, this distribution π can be a measure of page importance

Mechanics

- In order to compute this stationary probability distribution, we need to estimate the probability in every entry of the matrix P(t)
- In practice, this matrix is usually difficult to obtain, because it is hard to get the information for all possible time intervals
- To tackle this problem, a novel algorithm is proposed which is based on the *transition rate matrix*
- The transition rate matrix is defined as the derivative of P(t) when t goes to 0, if it exists

$$\mathbf{Q = P'(0)}$$

- We call the matrix $Q = (q_{ij})_{NXN}$ the *Q-matrix*

The Q-matrix

- When the state space is finite, then there is a one-to-one correspondence between the Q-matrix and P(t), and $-INF < q_{ii} < +INF$ and $SUM_j\ q_{ij} = 0$
- Due to this correspondence, one also uses Q-Process to represent the original continuous-time Markov process, that is, the browsing process $X = \{X_s, s \geq 0\}$ defined before is a Q-Process because of the finite state space
- Advantages of using the Q-matrix
  - The parameters in the Q-matrix can be effectively estimated from the data
  - Based on the Q-matrix, there is an efficient way of computing the stationary probability distribution of P(t)
- The so-called EMC is a discrete-time Markov process featured by a transition probability matrix with zero values in all its diagonal positions and $-q_{ij}/q_{ii}$ in the off-diagonal positions

The Theorem

THEOREM 1. *Suppose X is a Q-process, and Y is the Embedded Markov Chain derived from its Q-matrix. Let* $\pi = (\pi_1, \ldots, \pi_N)$ *and* $\tilde{\pi} = (\tilde{\pi}_1, \ldots, \tilde{\pi}_N)$ *denote the stationary probability distributions of the process X and Y, then we have*

$$\pi_i = \frac{\frac{\tilde{\pi}_i}{q_{ii}}}{\sum_{j=1}^{N} \frac{\tilde{\pi}_j}{q_{jj}}} \qquad (2)$$

- Note that the process Y is a discrete-time Markov chain, so its stationary probability distribution $\tilde{\pi}$ can be calculated by many simple and efficient methods such as the power method
- Next we will explain how to estimate the parameters in the Q-matrix, or equivalently parameter q_{ii} and the transition probabilities $-q_{ij}/q_{ii}$ ($-q_{ij}/q_{ii} > 0$, since $q_{ii} < 0$)

Estimation of q_{ii}

- For a Q-Process, the staying time T_i on the i^{th} vertex is governed by an exponential distribution parameterized by q_{ii}:

$$P(Ti > t) = \exp(q_{ii}\ t)$$

- This implies that we can estimate q_{ii} from large numbers of observations on the staying time in the user behavior data
- This task is non-trivial because the observations in the user behavior data usually contain noise due to Internet connection speed, page size, page structure, and other factors, i.e., the observed values do not completely satisfy the exponential distribution
- We suppose that Z is the combination of real staying time T_i and noise U, i.e.,

$$Z = U + T_i$$

Estimation of Transition Probability in EMC

- Transition probabilities in the EMC describe the 'pure' transitions of the surfer on the user browsing graph
- Estimation of them can be based on the observed transitions between pages in the user behavior data
- It can also be related to the green traffic in the data
- We use the following method to integrate these two kinds of information for the estimation

Estimation of Transition Probability in EMC

We start with the user browsing graph $G =< V, W, T, \sigma >$. We then add a pseudo-vertex (the $(N+1)^{th}$ vertex) to G, and add two types of edges: the edges from the last page in each session to the pseudo-vertex, associated with the click number of the last page as its weight; and the edges from the pseudo-vertex to the first page in each session, associated with the reset probability. We denote the new graph as $\tilde{G} =< \tilde{V}, \tilde{W}, T, \tilde{\sigma} >$, where $|\tilde{V}| = N+1$, $\tilde{\sigma} =< \tilde{\sigma}_1, \ldots, \tilde{\sigma}_N, 0 >$. Then we explain the EMC model as the random walk on this new graph $\tilde{G}$. Based on *the law of large number* [19], the transition probabilities in the EMC are estimated as below,

$$-\frac{q_{ij}}{q_{ii}} = \begin{cases} \alpha \frac{\tilde{w}_{ij}}{\sum_{k=1}^{N+1} \tilde{w}_{ik}} + (1-\alpha)\sigma_j, & i \in V, j \in \tilde{V} \\ \sigma_j, & i = N+1, j \in V \end{cases} \quad (8)$$

Estimation of Transition Probability in EMC

- The intuitive explanation of the above transition is as follows:
  - When the surfer walks on the user browsing graph, she may go ahead along the edges with the probability α, or choose to restart from a new page with the probability (1- α)
  - The selection of the new page is determined by the reset probability
  - One advantage of using (8) for estimation is that the estimation will not be biased by the limited number of observed transitions.
  - The other advantage is that the corresponding EMC is primitive, and thus has a unique stationary distribution
  - Therefore, we can use the power method to calculate this stationary distribution in an efficient manner.

The BrowseRank algorithm

Input: the user behavior data.
Output: the page importance score π
Algorithm:

1. Construct the user browsing graph (see Section 3.1).
2. Estimate q_{ii} for all pages(see Section 3.3.2).
3. Estimate the transition probability matrix of the EMC and then get its stationary probability distribution by means of power method (see Section 3.3.3).
4. Compute the stationary probability distribution of the Q-process by using of equation (2).

Top-20 Websites by 3 algorithms

No.	PageRank	TrustRank	BrowseRank
1	adobe.com	adobe.com	***myspace.com***
2	passport.com	yahoo.com	msn.com
3	msn.com	google.com	yahoo.com
4	microsoft.com	msn.com	***youtube.com***
5	yahoo.com	microsoft.com	live.com
6	google.com	passport.net	***facebook.com***
7	mapquest.com	ufindus.com	google.com
8	miibeian.gov.cn	***sourceforge.net***	ebay.com
9	w3.org	***myspace.com***	***hi5.com***
10	godaddy.com	***wikipedia.org***	***bebo.com***
11	statcounter.com	phpbb.com	***orkut.com***
12	apple.com	yahoo.co.jp	aol.com
13	live.com	ebay.com	***friendster.com***
14	xbox.com	nifty.com	***craigslist.org***
15	passport.com	mapquest.com	google.co.th
16	***sourceforge.net***	cafepress.com	microsoft.com
17	amazon.com	apple.com	***comcast.net***
18	paypal.com	infoseek.co.jp	***wikipedia.org***
19	aol.com	miibeian.gov.cn	***pogo.com***
20	***blogger.com***	***youtube.com***	***photobucket.com***

Results 1

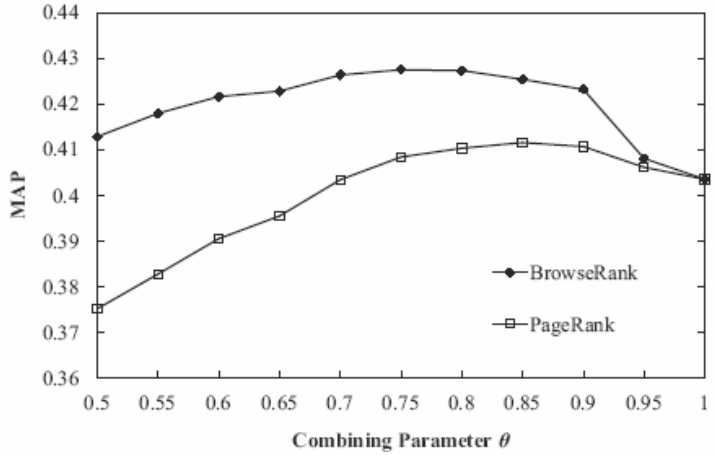


Figure 3: Search performance in terms of MAP for BrowseRank and PageRank

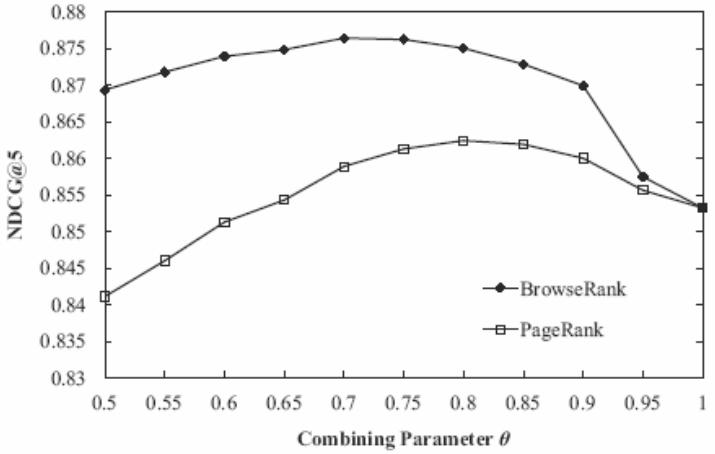


Figure 7: Search performance in terms of NDCG@5 for BrowseRank and PageRank

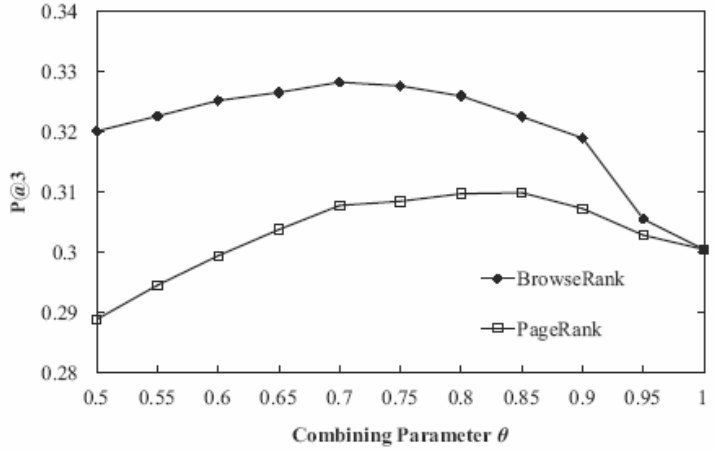


Figure 4: Search performance in terms of P@3 for BrowseRank and PageRank

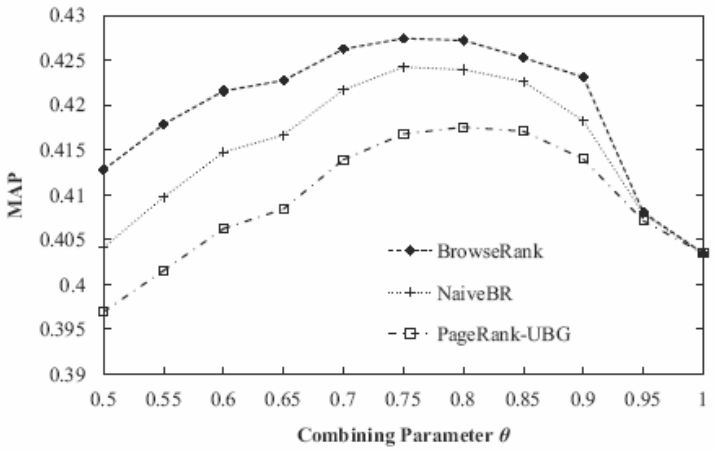


Figure 8: Search performance in terms of MAP for BrowseRank and two simple algorithms

Results 2

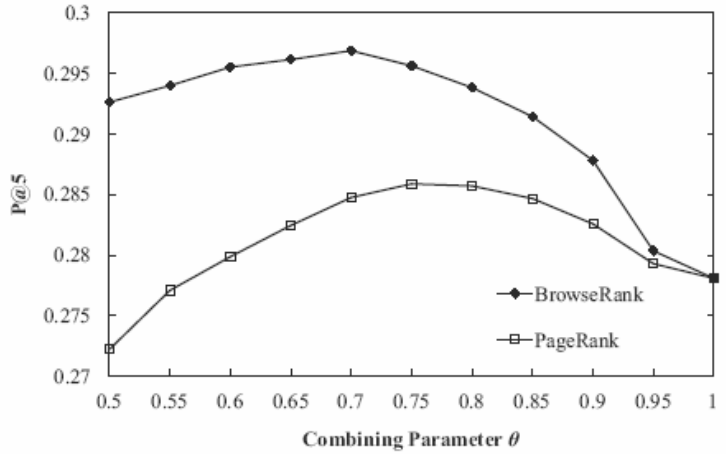


Figure 5: Search performance in terms of P@5 for BrowseRank and PageRank

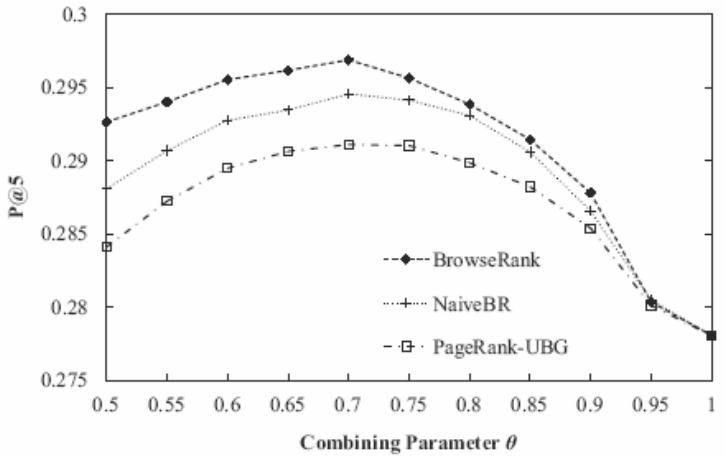


Figure 9: Search performance in terms of P@5 for BrowseRank and two simple algorithms

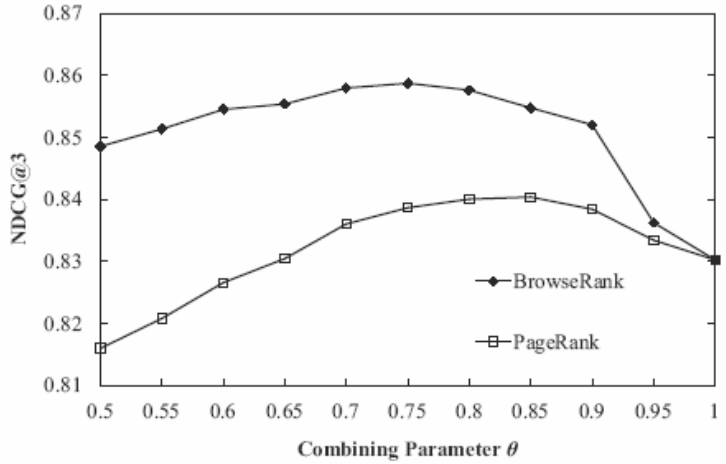


Figure 6: Search performance in terms of NDCG@3 for BrowseRank and PageRank

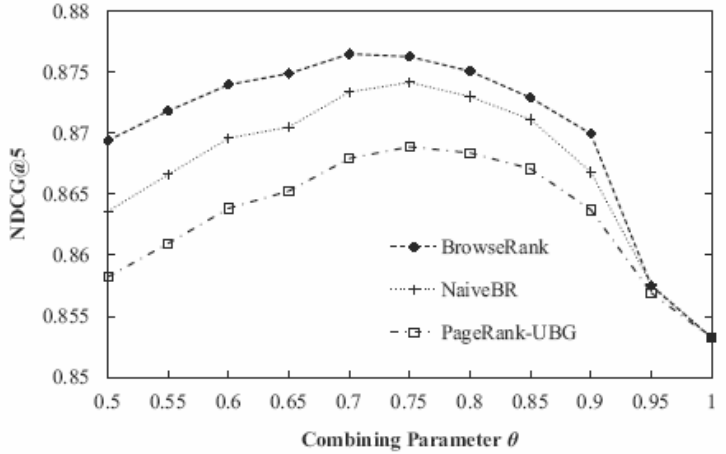


Figure 10: Search performance in terms of NDCG@5 for BrowseRank and two simple algorithms

Spam: Δεν είναι μόνο για τα inboxes

Link Spam Farms

- **Spamming**: Παραπλάνηση των μηχανών αναζήτησης για να αποκτηθεί υψηλότερη διάταξη (ranking) για κάποιες σελίδες (ή ιστοτόπους) απ' αυτή που πραγματικά αξίζουν.