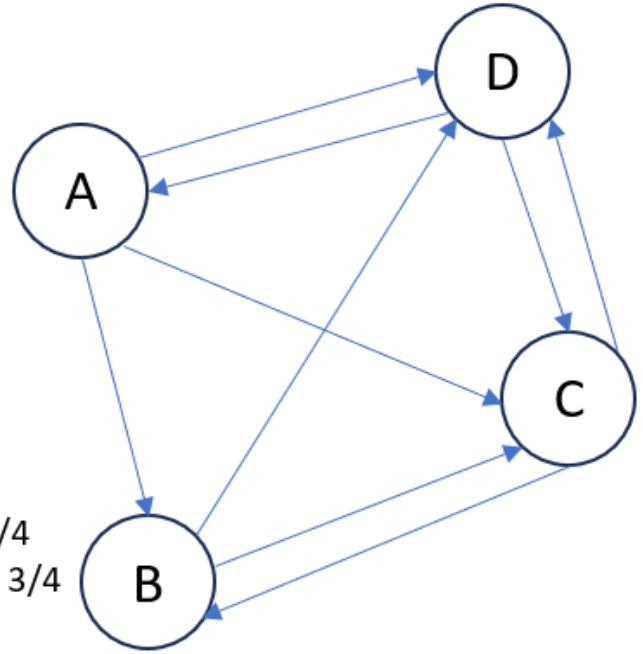
Άσκηση



α = 1/4

1-α = 3/4

- Αν μια γραμμή δεν έχει 1, τότε αντικαθιστούμε κάθε στοιχείο με 1/N – Για τις υπόλοιπες γραμμές κάνουμε το εξής:
- Διαιρούμε κάθε 1 με το πλήθος των 1 σε κάθε γραμμή – αν έχουμε τρία 1, τότε θα τα αντικαταστήσουμε με 1/3
- Πολλαπλασιάζουμε τον πίνακα που προκύπτει με 1-α
- Προσθέτουμε α/N σε κάθε αποτέλεσμα για να πάρουμε την τελική μορφή του P

	A	B	C	D
A	0	1	1	1
B	0	0	1	1
C	0	1	0	1
D	1	0	1	0

Διαιρώ με το πλήθος των 1

	A	B	C	D
A	0	1/3	1/3	1/3
B	0	0	1/2	½
C	0	1/2	0	½
D	½	0	1/2	0

Πολλαπλασιάζω με 3/4

	A	B	C	D
A	0	3/12	3/12	3/12
B	0	0	3/8	3/8
C	0	3/8	0	3/8
D	3/8	0	3/8	0

Προσθέτω α/N = 1/4/4 = 1/16

	A	B	C	D
A	1/16	1/16+3/12	1/16+3/12	1/16+3/12
B	1/16	1/16	1/16+3/8	1/16+3/8
C	1/16	1/16+3/8	1/16	1/16+3/8
D	1/16+3/8	1/16	1/16+3/8	1/16

P

	A	B	C	D
A	1/16	5/16	5/16	5/16
B	1/16	1/16	7/16	7/16
C	1/16	7/16	1/16	7/16
D	7/16	1/16	7/16	1/16

X

0	1	0	0

1ος πολ/μος

0	1	0	0	X	1/16	5/16	5/16	5/16
					1/16	1/16	7/16	7/16
					1/16	7/16	1/16	7/16
					7/16	1/16	7/16	1/16

Αποτέλεσμα

1/16	1/16	7/16	7/16

2ος πολ/μος

1/16	1/16	7/16	7/16	X	1/16	5/16	5/16	5/16
					1/16	1/16	7/16	7/16
					1/16	7/16	1/16	7/16
					7/16	1/16	7/16	1/16

.....

Topic Specific PageRank

- Μέχρι στιγμής έχουμε υποθέσει πως οι χρήστες κατά το teleporting επιλέγουν σελίδα μετάβασης με βάση την ομοιόμορφη κατανομή
- Αυτό σημαίνει πως όλες οι σελίδες έχουν την ίδια πιθανότητα να επιλεγούν
- Όμως αυτό δεν σημαίνει πως ισχύει πάντα
- Θα πρέπει να παράξουμε τις τιμές PageRank με βάση συγκεκριμένα ενδιαφέροντα
- Για παράδειγμα, ένας μάγειρας μπορεί να επιθυμεί, λόγω επαγγέλματος, να έχουν οι σελίδες μαγειρικής μεγαλύτερο βάρος
- Ας υποθέσουμε πως οι σελίδες μαγειρικής βρίσκονται κοντά στο γράφο
- Έτσι, ένας χρήστης-μάγειρας πιο συχνά θα πλοηγείται σε σελίδες μαγειρικής αυξάνοντας τις πιθανότητες αυτών των σελίδων

Topic Specific PageRank

- Υποθέτουμε πως ένας χρήστης εκτελεί τη λειτουργικότητα teleporting σε μια τυχαία σελίδα στο πεδίο όμως του ενδιαφέροντός του
- Συνεπώς, δεν ισχύει η ομοιόμορφη επιλογή όλων των σελίδων του γράφου
- Στόχος είναι η συλλογή όλων των σελίδων σε ένα πεδίο ενδιαφέροντος
- Χρειαζόμαστε ένα σύνολο M σελίδων π.χ. μαγειρικής
- Αυτό μπορεί να γίνει με χρήση κάποιου ευρετηρίου ή καταλόγου των σελίδων (π.χ. Yahoo)

Topic Specific PageRank

- Δοσμένου του συνόλου M συμπεραίνουμε πως υπάρχει ένα μη κενό σύνολο σελίδων Y υπερσύνολο του M στο οποίο ένας τυχαίος περίπατος θα έχει μια steady state κατανομή $\vec{\pi}_{\mathrm{M}}$
- Οι σελίδες εκτός του Y θα λάβουν τιμή 0
- Η κατανομή $\vec{\pi}_{\mathrm{M}}$ ονομάζεται topic specific PageRank

Topic Specific PageRank

- Μπορούμε να θεωρήσουμε πολλαπλές κατανομές για κάθε πεδίο ενδιαφέροντος
- Κάθε κατανομή αναθέτει μια τιμή PageRank σε κάθε σελίδα
- Για ένα χρήστη που έχει ενδιαφέροντα σε μόνο μια περιοχή, μπορούμε να βασιστούμε σε μια μόνο κατανομή
- Το πρόβλημα είναι το πως θα γνωρίζουμε το πεδίο ενδιαφέροντος του χρήστη
  - Μπορεί ο ίδιος ο χρήστης να έχει καταχωρήσει τις επιθυμίες του
  - Η μηχανή αναζήτησης μπορεί να υιοθετήσει μεθόδους μηχανικής μάθησης και να παρατηρήσει τη συμπεριφορά του

Topic Specific PageRank

- Η προηγούμενη περίπτωση του μοναδικού ενδιαφέροντος είναι εύκολη
- Τι συμβαίνει όταν ο χρήστης έχει πολλαπλά ενδιαφέροντα;
- Μπορούμε να υπολογίσουμε ένα ‘προσωπικό PageRanking’;
- Ένας τρόπος είναι προσεγγίσουμε τα ενδιαφέροντα ως ένα γραμμικό συνδυασμό ενός μικρού αριθμού ενδιαφερόντων

Topic Specific PageRank

- Ο χρήστης μπορεί να εκτελέσει το teleporting ως ακολούθως:
  - Αρχικά καθορίζουμε αν θα μετακινηθεί σε σελίδες π.χ. μαγειρικής (σύνολο M) ή σελίδες πολιτικής
  - Η επιλογή γίνεται τυχαία με κάποιο ποσοστό
  - Παράδειγμα: επιλογή σελίδων μαγειρικής στο 60% των μετακινήσεων και επιλογή σελίδων πολιτικής στο 40% των μετακινήσεων
  - Αφού επιλέξουμε το πεδίο μετακίνησης, στη συνέχεια επιλέγουμε με βάση την ομοιόμορφη κατανομή από τις διαθέσιμες σελίδες

Topic Specific PageRank

- Γενικά, οι υπολογισμοί θα μειωθούν αν παρατηρήσουμε πως η ‘εξέλιξη’ των πιθανοτήτων είναι ένα γραμμικό σύστημα πάνω από τις καταστάσεις της αλυσίδας Markov
- Για παράδειγμα, ο υπολογισμός που περιγράψαμε θα έχει ως εξής:

  $0.6\vec{\pi}_{\text{М}}+ 0.4\vec{\pi}_{\text{П}}$

Hubs and Authorities

- Στο επόμενο σχήμα, σε κάθε ερώτημα ανατίθενται δύο τιμές που ονομάζονται hub score και authority score
- Για κάθε ερώτημα εξάγουμε δύο λίστες με σελίδες αντί για μια
- Ορισμένες φορές οι χρήστες αναζητούν πληροφορίες για ένα ευρύτερο πεδίο (broader topic)
- Τα ερωτήματα αυτά ονομάζονται broad-topic searches
- Παράδειγμα: επιθυμώ πληροφορίες για τη Λαμία
- Για αυτές τις πληροφορίες υπάρχουν σελίδες που διατίθενται από τις αντίστοιχες αρχές - αυτές οι σελίδες ονομάζονται authorities
- Οι υπόλοιπες ονομάζονται hub σελίδες

Hubs and Authorities

- Η προσέγγιση είναι να χρησιμοποιήσουμε τις hub σελίδες για να εντοπίσουμε τις authorities
- Οι σημαντικότερες από αυτές θα έχουν υψηλό hub score
- Μια καλή hub σελίδα είναι αυτή που ‘δείχνει’ σε αρκετές authorities
- Μια καλή authority σελίδα είναι αυτή στην οποία ‘δείχνουν’ αρκετές hub σελίδες

Hubs and Authorities

- Για μια σελίδα υ, χρησιμοποιούμε το h(υ) για το hub score και το α(υ) για το authority score
- Αρχικά, θέτουμε h(υ)=α(υ)=1 για όλους τους κόμβους
- Το κομβικό σημείο του αλγορίθμου είναι η ενημέρωση των τιμών ανα ζεύγη (hub and authority scores) για όλες τις σελίδες
- Αν σε μια σελίδα υ υπάρχει σύνδεσμος προς την y τότε έχουμε:

$$h(v) \leftarrow \sum_{v \mapsto y} a(y)$$

$$a(v) \leftarrow \sum_{y \mapsto v} h(y)$$

- Στην πρώτη εξίσωση, το hub score είναι το άθροισμα των authorities των σελίδων στις οποίες δείχνει η υ

Hubs and Authorities

- Στη δεύτερη εξίσωση, το authority score είναι το άθροισμα όλων των hub scores των σελίδων οι οποίες δειχνουν στην υ

$$h(v) \leftarrow \sum_{v \mapsto y} a(y)$$
$$a(v) \leftarrow \sum_{y \mapsto v} h(y)$$

- Έστω ότι $\vec{h}$ και $\vec{\alpha}$ είναι τα διανύσματα όλων των τιμών
- Έστω A είναι ο τετραγωνικός πίνακας γειτνίασης
- Οι εξισώσεις μπορεί να ξαναγραφτούν ως εξής:

$$\vec{h} \leftarrow A\vec{a}$$
$$\vec{a} \leftarrow A^T\vec{h}$$

- A^T είναι ο αντίστροφος του A

Hubs and Authorities

- Στις εξισώσεις, αντικαθιστούμε τα διανύσματα και παίρνουμε:

$$\vec{h} \leftarrow A\vec{a}$$
$$\vec{a} \leftarrow A^T\vec{h}$$

$$\vec{h} \leftarrow AA^T\vec{h}$$
$$\vec{a} \leftarrow A^TA\vec{a}$$

- Τώρα μοιάζουν με εξισώσεις όπου εμπλέκονται ιδιοδιανύσματα και μπορούν να γραφούν ως εξής:

$$\vec{h} = (1/\lambda_h)AA^T\vec{h}$$
$$\vec{a} = (1/\lambda_a)A^TA\vec{a}$$

- Το λ_h είναι η ιδιοτιμή του AA^T και το λ_α είναι η ιδιοτιμή του A^TA

Hubs and Authorities

- Με την προηγούμενη ανάλυση καταλήγουμε στη μέθοδο Hyperlink-Induced Topic Search (HITS)
  - Επιλέγουμε τις σελίδες, σχηματίζουμε το γράφο με βάση τους συνδέσμους και υπολογίζουμε τα AA^T και A^TA
  - Υπολογίζουμε τα ιδιοδιανύσματα των AA^T και A^TA για να εξάγουμε τα $\vec{h}$ και $\vec{\alpha}$
  - Εξάγουμε τις υψηλότερες τιμές για τα hubs & authorities

Επιλογή Σελίδων

- Κατά την επιλογή των σελίδων ακολουθούμε τα επόμενα βήματα:
  - Χρησιμοποιούμε ένα ευρετήριο κειμένου για να πάρουμε τις σελίδες που περιλαμβάνουν τον όρο του ερωτήματος – το σύνολο αυτό ονομάζεται root set
  - Χτίζουμε ένα base set σελίδων με βάση τους συνδέσμους (περιλαμβάνουμε σελίδες που δείχνουν στο root set ήπου αναφέρονται στο root set)
- Το base set κατασκευάζεται ως εξής:
  - Μια καλή authority σελίδα μπορεί να μην περιέχει τον όρο του ερωτήματος
  - Αν με το ερώτημα ‘πιάσουμε’ μια καλή hub σελίδα στο root set τότε η εξέταση όλων των σελίδων που υπάρχει σύνδεσμος στο root set θα ‘πιάσει’ όλες τις καλές authorities στο base set
  - Αντίστοιχα, αν με το ερώτημα ‘πιάσουμε’ μια καλή authority σελίδα στο root set τότε η εξέταση όλων των σελίδων που δείχνουν στη σελίδα αυτή θα ‘πιάσει’ όλες τις καλές hub σελίδες στο base set

Επιλογή Σελίδων

- Παράδειγμα
  - Αποτέλεσμα εκτέλεσης του αλγορίθμου για το ερώτημα ‘japan elementary schools’

Hubs	Authorities
schools	The American School in Japan
LINK Page-13	The Link Page
˜ú–{,ÌŠw⊏Z	‰⁴⊏è⊏s—§˜ä˜c⊏¬Šw⊏Zƒz⊏[ƒ⊏ƒy⊏[ƒW
⊏a‰„⊏¬Šw⊏Zƒz⊏[ƒ⊏ƒy⊏[ƒW	Kids' Space
100 Schools Home Pages (English)	ˆÀ⊏é⊏s—§ˆÀ⊏é⊏¼•˜⊏¬Šw⊏Z
K-12 from Japan 10/...rnet and Education)	«{⊏é‹ª˜ç‘åŠw•⊏‘®⊏¬Šw⊏Z
http://www...iglobe.ne.jp/~IKESAN	KEIMEI GAKUEN Home Page (Japanese)
,I,f,j⊏¬Šw⊏Z,U˜N,P‘g•¯Œê	Shiranuma Home Page
⊏ÒŠ—’¬—§⊏ÒŠ—˜Œ⊏¬Šw⊏Z	fuzoku-es.fukui-u.ac.jp
Koulutus ja oppilaitokset	welcome to Miasa E&J school
TOYODA HOMEPAGE	⊏_˜Þ⊏ìŒ§⊏E‰¡•I⊏s—§‘†⊏ì⊏¼⊏¬Šw⊏Z,Ìƒy
Education	http://www...p/~m_maru/index.html
Cay's Homepage(Japanese)	fukui haruyama-es HomePage
–y˜ì⊏¬Šw⊏Z,Ìƒz⊏[ƒ⊏ƒy⊏[ƒW	Torisu primary school
UNIVERSITY	goo
‰J—³⊏¬Šw⊏Z DRAGON97-TOP	Yakumo Elementary,Hokkaido,Japan
⊏Â‰⁴⊏¬Šw⊏Z,T˜N,P‘gƒz⊏[ƒ⊏ƒy⊏[ƒW	FUZOKU Home Page
¶µ°é¼ÂÁ© ¥á¥Ë¥å¡¼ ¥á¥Ë¥å¡¼	Kamishibun Elementary School...

Recommender Systems

Εισαγωγή

- Υπάρχουν δύο ειδών ανάγκες στους χρήστες όσον αφορά στην ανάκτηση πληροφορίας:
  - Short-term: είναι μια προσωρινή ανάγκη από μια στατική πηγή πληροφορίας
  - Long-term: πρόκειται για filtering ή recommendation για έγγραφα από μια δυναμική πηγή πληροφορίας
- Το filtering είναι παρόμοιο με το recommendation
- Πρόκειται να ληφθεί απόφαση από το σύστημα σχετικά με τη συσχέτιση ενός εγγράφου με ένα χρήστη
- Είναι πιο δύσκολη διαδικασία σε σχέση με την απλή αναζήτηση πληροφοριών

Εισαγωγή

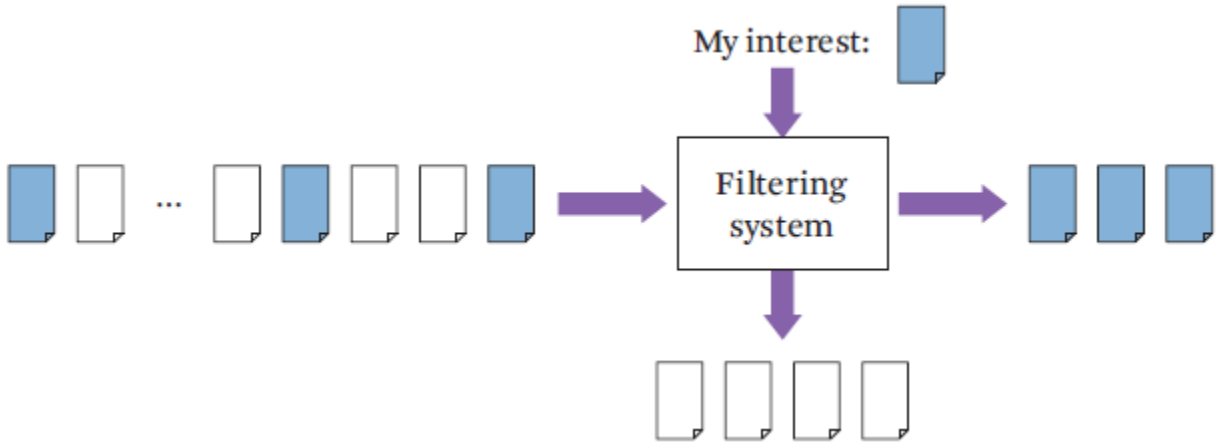
- Το βασικό ερώτημα έχει ως εξής: πρόκειται ο χρήστης u να ενδιαφέρεται για το αντικείμενο/έγγραφο x;
- Υιοθετούνται δύο στρατηγικές:
  - Content-based filtering: βλέπουμε τι αρέσει στο χρήστη και χαρακτηρίζουμε το αντικείμενο/έγγραφο
  - Collaborative filtering: βλέπουμε ποιος ενδιαφέρεται για το αντικείμενο/έγγραφο και χαρακτηρίζουμε το χρήστη

Content-Based Filtering

- Η βασική ιδέα είναι να μάθουμε το περιεχόμενο που αρέσει στο χρήστη
- Στη συνέχεια ταιριάζουμε το περιεχόμενο των εγράφων με αυτό που αρέσει στο χρήστη
- Μπορούμε να επεκτείνουμε ένα σύστημα ανάκτησης πληροφοριών και να προσθέσουμε ένα όριο
- Αρχικά, πρέπιε να ορίσουμε ένα όριο χωρίς να έχουμε πληροφόρηση για το χρήστη
- Στη συνέχεια, θα πρέπει να μάθουμε από την αλληλεπίδραση με το χρήστη και να ενημερώσουμε το όριο αυτό

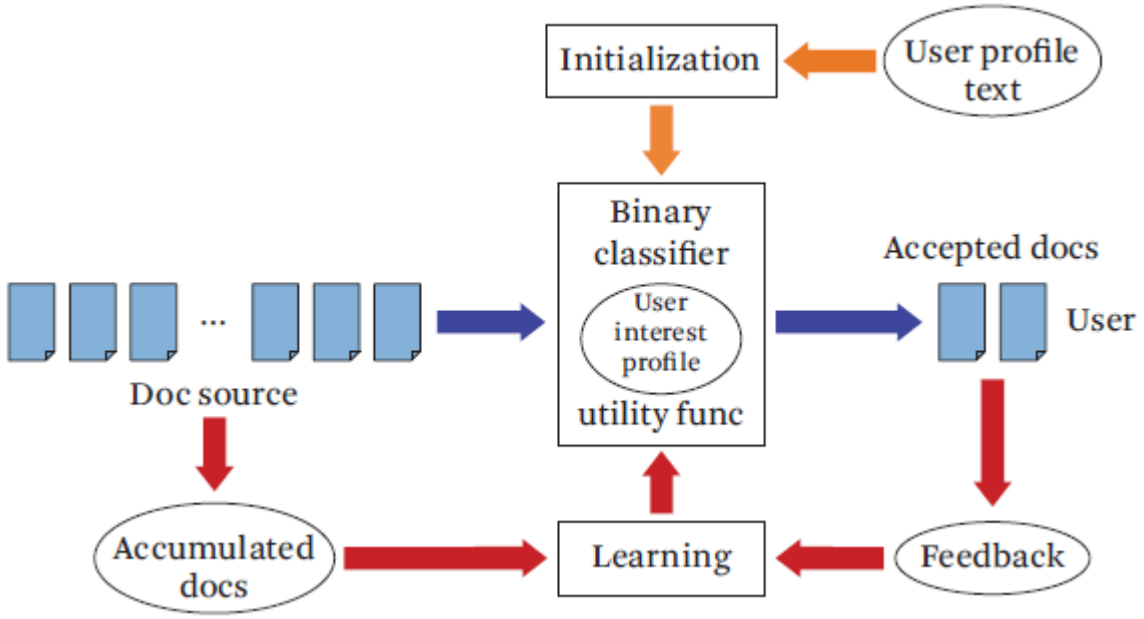
Content-Based Filtering

- Στην εικόνα βλέπουμε πως ένα σύστημα απορροφά τα έγγραφα που είναι σχετικά με το χρήστη
- Το σύστημα βλέπει το περιεχόμενο των αντικειμένων και το συγκρίνει με τις προτιμήσεις των χρηστών και τα αποτελέσματα της αλληλεπίδρασης με αυτούς



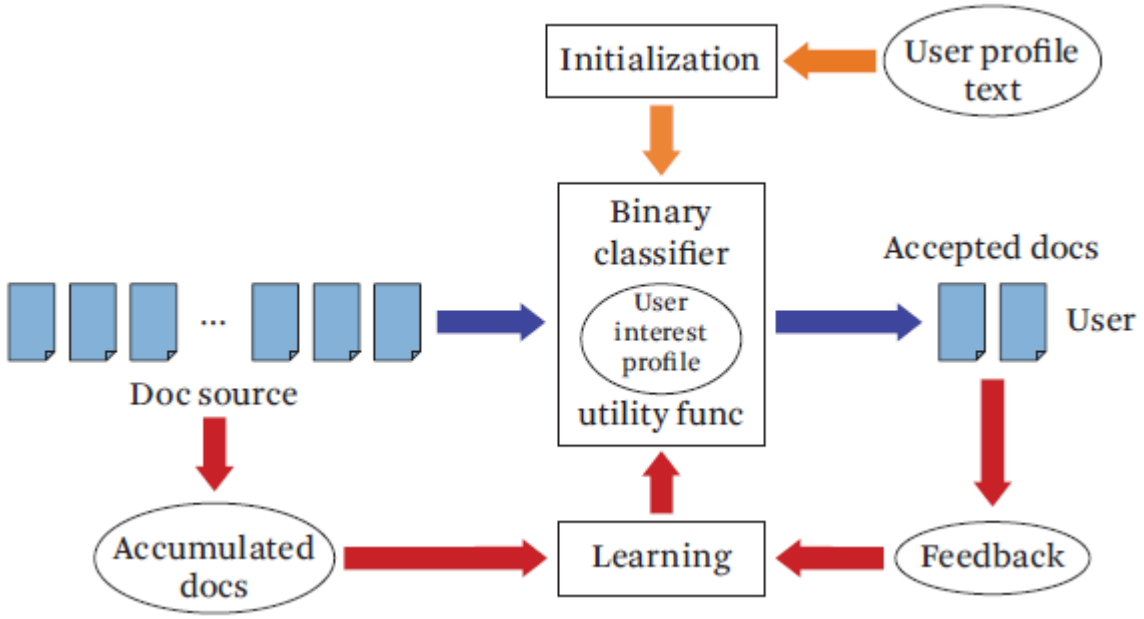
Content-Based Filtering

- Ένας binary classifier υιοθετείται από το σύστημα ο οποίος μπορεί να έχει γνώση για τις προτιμήσεις του χρήστη
- Το προφίλ του χρήστη μπορεί να είναι μια σύνοψη κειμένου
- Μπορεί επίσης να είναι λέξεις κλειδιά



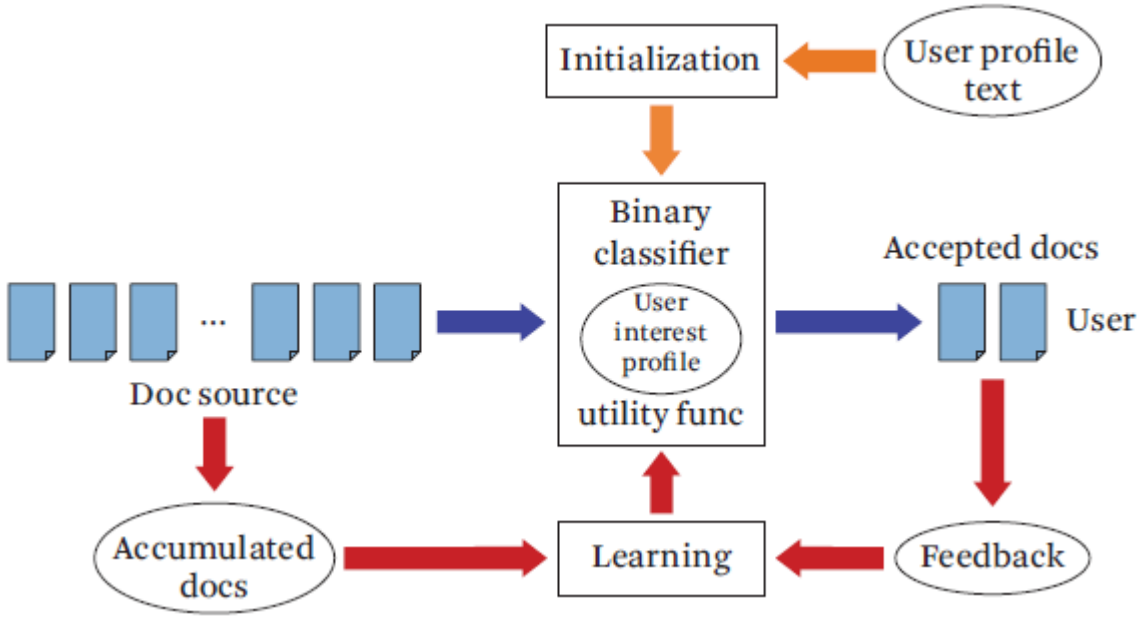
Content-Based Filtering

- Στη συνέχεια υιοθετείται μια συνάρτηση οφέλους (utility function)
- Η συνάρτηση βοηθά το σύστημα στο να θέσει ένα όριο θ που θα είναι το όριο αποδοχής
- Με βάση το όριο αποδοχής, ένα αντικείμενο θα παρουσιαστεί ή όχι στο χρήστη



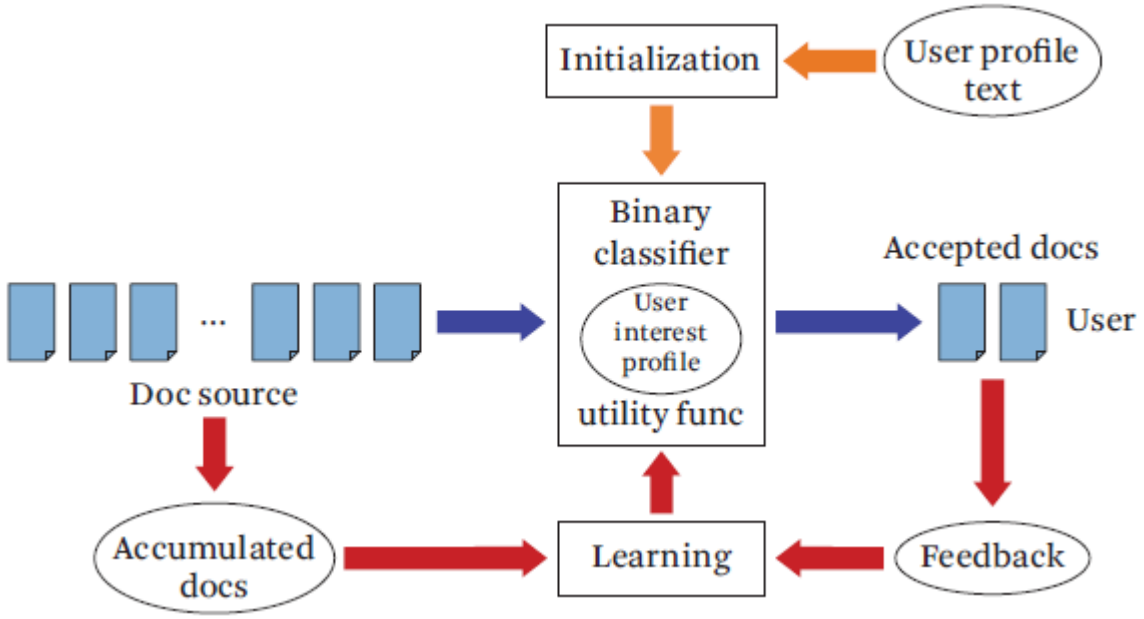
Content-Based Filtering

- Η μονάδα εκμάθησης θα προσαρμόσει τις παραμέτρους του συστήματος
- Οι χρήστες μπορεί να υποδείξουν αν ένα αντικείμενο είναι σχετικό ή όχι με τις προτιμήσεις τους



Content-Based Filtering

- Για την αποτίμηση του αποτελέσματος μπορούμε να χρησιμοποιήσουμε την ακόλουθη συνάρτηση: U=3|R| - 2|R'|
- R: τα σχετικά έγγραφα, R': τα άσχετα έγγραφα



Content-Based Filtering

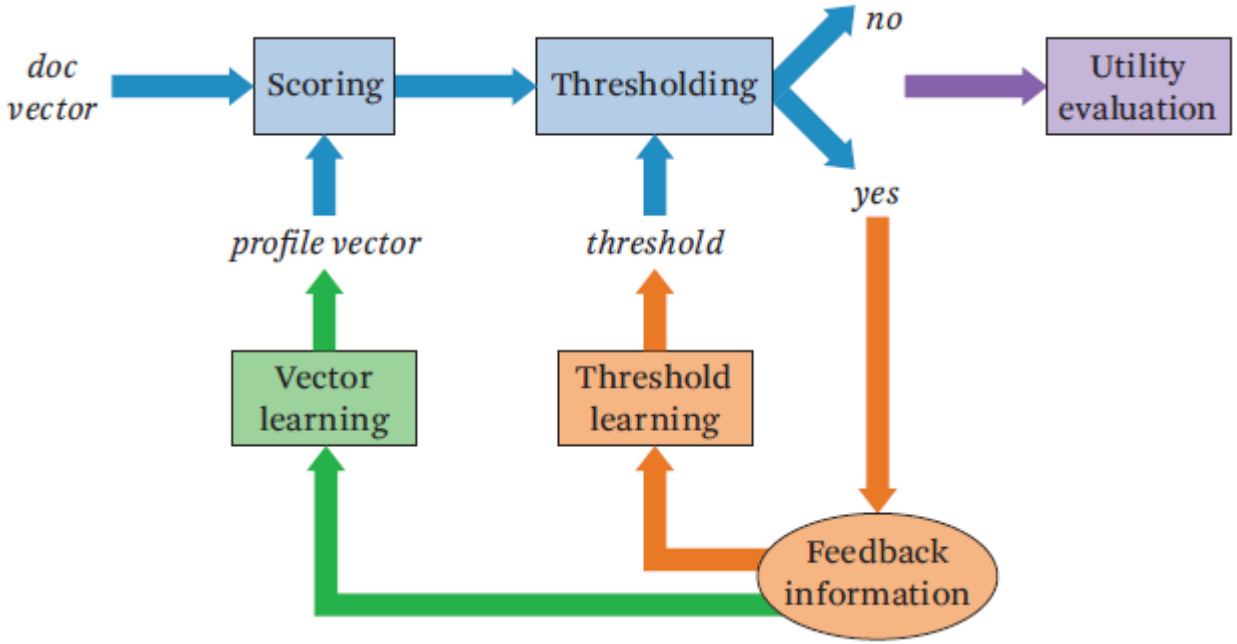
- Τα τρία βασικά στοιχεία του συστήματος είναι:
  - Initialization module: εκκινεί το σύστημα με βάση περιορισμένη περιγραφή από το χρήστη
  - Decision module: δοσμένου του κειμένου ενός εγγράφου και του προφιλ του χρήστη αποφασίζει αν το έγγραφο είναι σχετικό ή όχι
  - Learning module: μαθαίνει από την αλληλεπίδραση με το χρήστη
- Όλα τα στοιχεία του συστήματος πρέπει να λειτουργούν με το βέλτιστο τρόπο ώστε να μεγιστοποιηθεί το όφελος αποτίμησης

Content-Based Filtering

- Αρχικά, υιοθετούμε τεχνικές που ήδη έχουμε δει για να βγάλουμε τα scores
- Έχουμε δει τεχνικές ομοιότητας ερωτημάτων με έγγραφα
- Υιοθετούμε ένα όριο θ για την απόφαση
- Αν score(d)>θ τότε θεωρούμε πως το d είναι σχετικό
- Το θ ενημερώνεται καθώς το σύστημα λειτουργεί

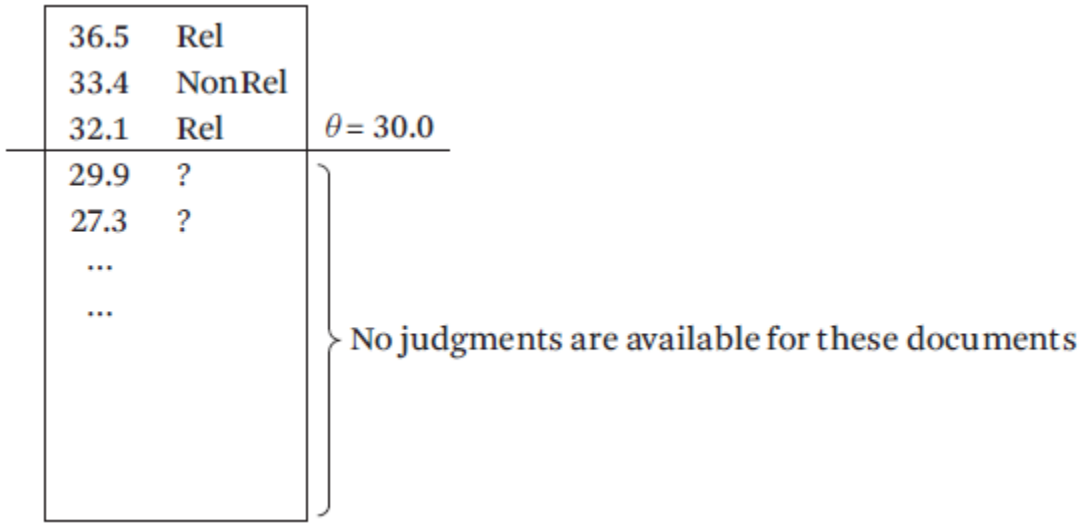
Content-Based Filtering

- Το σχήμα απεικονίζει το σύστημα με την υιοθέτηση του vector-space μοντέλου
- Τα έγγραφα και το προφιλ των χρηστών αναπαριστώνται ως διανύσματα



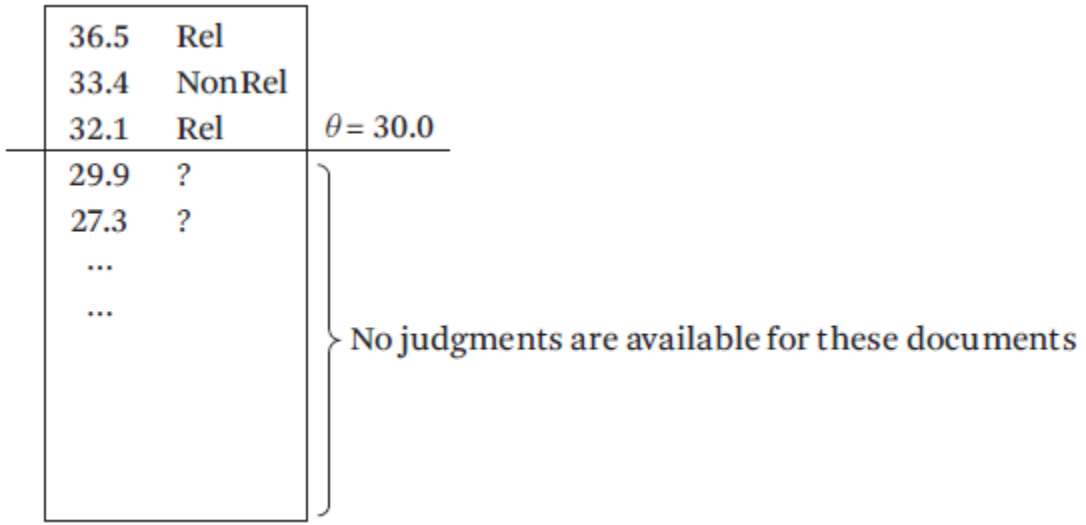
Content-Based Filtering

- Τι συμβαίνει όμως με την εκμάθηση του ορίου θ;
- Ουσιαστικά έχουμε τις τελικές τιμές και την απεικόνιση της συσχέτισης
- Έχουμε πολλά έγγραφα που δεν γνωρίζουμε τη συσχέτισή τους αφού η τελική τιμή είναι κάτω από το όριο
- Έτσι όμως τα προτεινόμενα έγγραφα δεν αποτελούν τυχαία επιλογή



Content-Based Filtering

- Γενικά, υπάρχουν λίγα έγγραφα που μπορούν να χρησιμοποιηθούν για την απεικόνιση της συσχέτισης και λίγα έγγραφα σχετικά
- Όμως, οι μέθοδοι μηχανικής μάθησης απαιτούν μεγάλα σύνολα δεδομένων
- Στην εξαιρετική περίπτωση της πρώτης απόφασης δεν υπάρχουν καθόλου δεδομένα



Content-Based Filtering

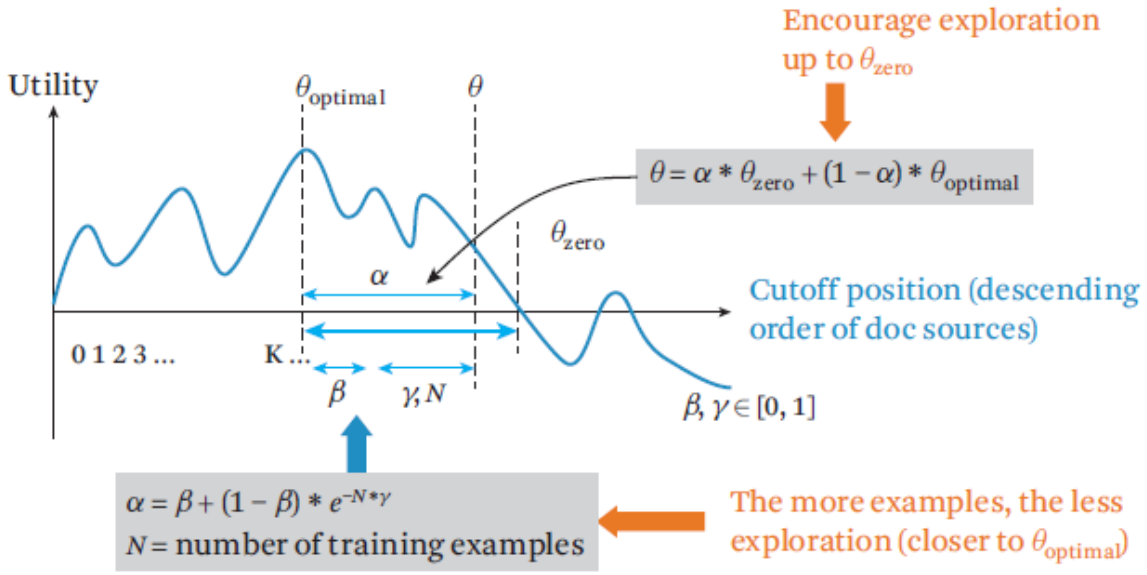
- Το προηγούμενο πρόβλημα ονομάζεται exploration-exploitation tradeoff
- Πρέπει να ‘εξερευνήσουμε’ τα έγγραφα για να δούμε αν ο χρήστης ενδιαφέρεται για αυτά
- Όμως δεν έχουμε ετικέτες για τα έγγραφα αυτά
- Επίσης, δεν θέλουμε να δείξουμε στο χρήστη πολλά άσχετα έγγραφα

Content-Based Filtering

- Θα μπορούσαμε να μειώσουμε το όριο θ και να δούμε την αλληλεπίδραση με το χρήστη για τα επιπλέον έγγραφα
- Πρόκειται για ένα tradeoff επειδή από τη μια θέλουμε να εξερευνήσουμε (explore) και από την άλλη δεν θέλουμε να δούμε πάρα πολλά έγγραφα μήπως και δώσουμε πολλά άσχετα έγγραφα
- Αν ξέρουμε τα ενδιαφέροντα του χρήστη δεν θέλουμε να αποκλίνουμε από αυτά
- Όμως, αν δεν αποκλίνουμε καθόλου δεν θα έχουμε ικανοποιητικό exploration και πιθανώς να χάσουμε κάποια ενδιαφέροντα του χρήστη

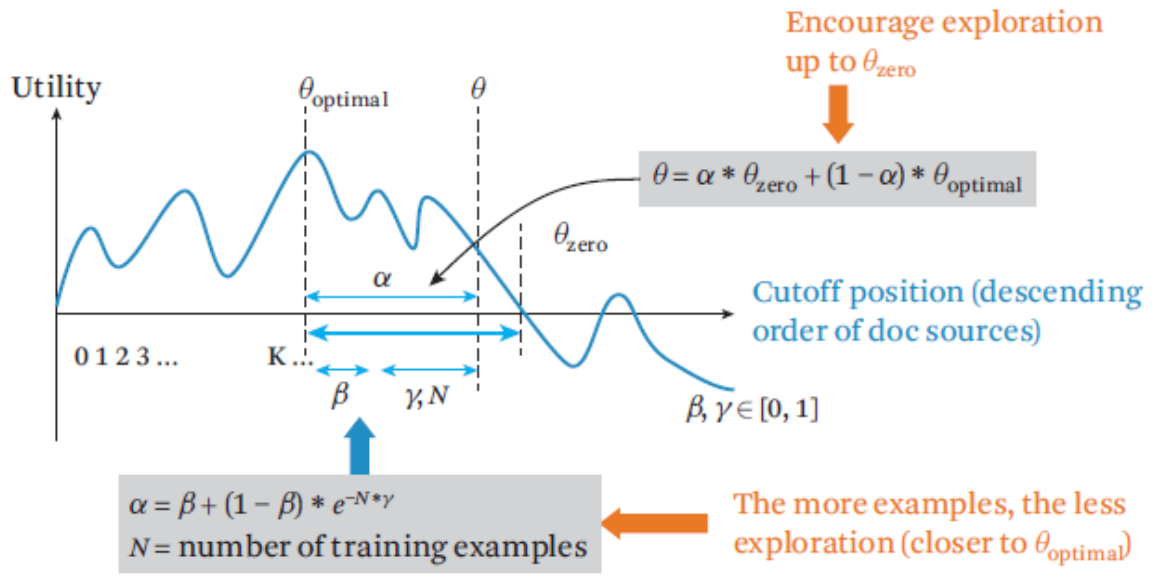
Content-Based Filtering

- Μια τεχνική για να ‘μάθουμε’ το όριο θ είναι ο αλγόριθμος beta-gamma
- Δοσμένης μια ταξινομημένης λίστας από ένα training dataset, της συσχέτισής τους και ένα όφελος U, απεικονίζουμε σχηματικά το όφελος για διάφορες τιμές του θ
- Κάθε θέση αντιστοιχεί σε ένα όριο



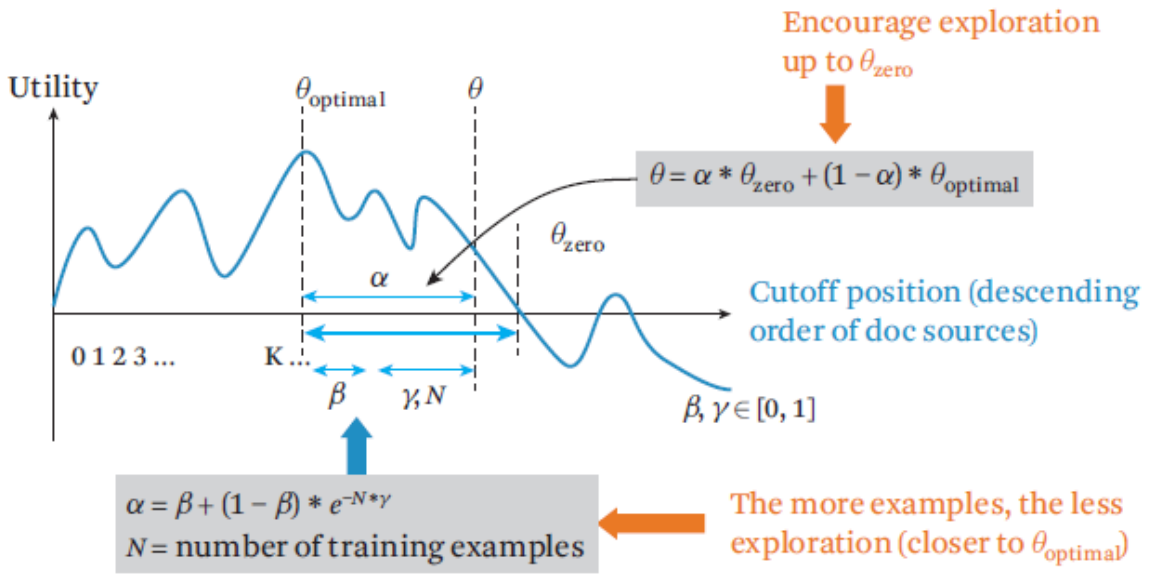
Content-Based Filtering

- Η επιλογή του α επηρεάζει το cutoff σημείο
- Επίσης, οι παράμετροι β και γ βοηθούν στην ενημέρωση του α
- Το βέλτιστο σημείο θ_{opt} είναι το μέγιστο όφελος που μπορούμε να αποκομίσουμε ενώ το θ_{zero} είναι το μηδενικό όφελος



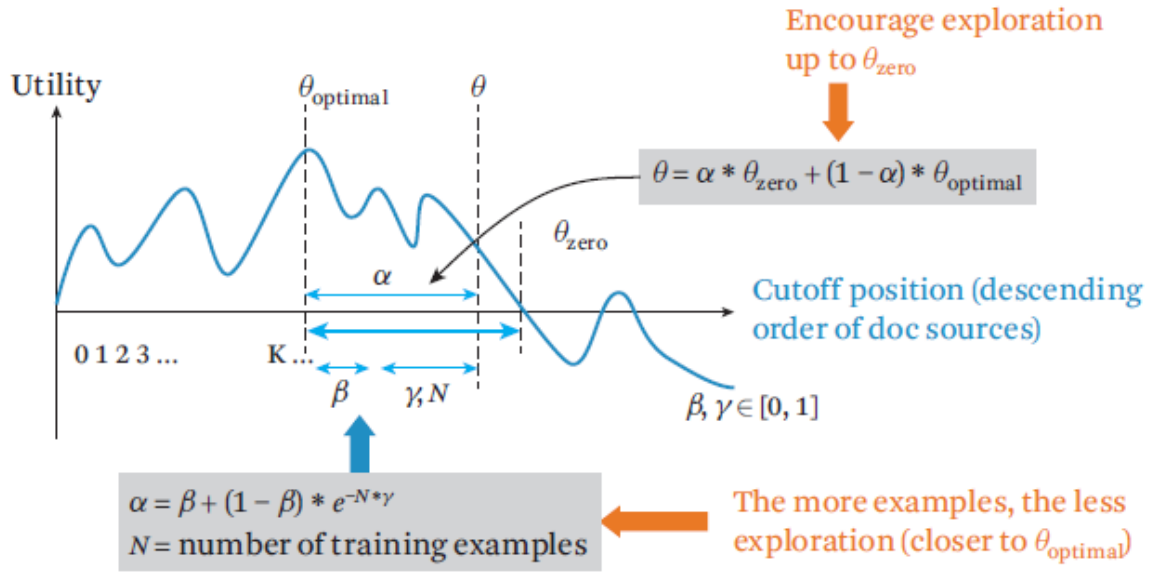
Content-Based Filtering

- Το β ελέγχει την απόκλιση από το θ_{opt} που μπορεί να οριστεί από προηγούμενα έγγραφα
- Το γ ελέγχει την επιρροή του μεγέθους του dataset N
- Όταν το N είναι μεγάλο έχουμε μικρότερο exploration



Content-Based Filtering

- Όταν το N είναι μικρό ο αλγόριθμος προσπαθεί να εξερευνήσει περισσότερο
- Όσο περισσότερα δείγματα θα έχουμε, τόσο λιγότερο θα εξερευνεί ο αλγόριθμος και τόσο πιο κοντά στο βέλτιστο όριο θα βρισκόμαστε

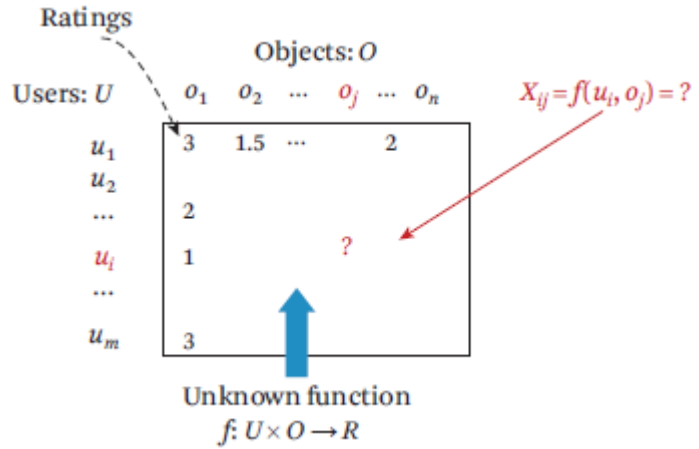


Collaborative Filtering

- Στην τεχνική του **collaborative filtering**, το σύστημα εξάγει συστάσεις με βάση τις κρίσεις άλλων χρηστών
- Η βασική ιδέα είναι να συμπεράνουμε τα ενδιαφέροντα και τις προτιμήσεις των χρηστών με βάση την ομοιότητα που έχουν με άλλους χρήστες
- Οι υποθέσεις που κάνουμε έχουν ως εξής:
  - Χρήστες με κοινά ενδιαφέροντα έχουν κοινές προτιμήσεις
  - Χρήστες με κοινές προτιμήσεις έχουν κοινά ενδιαφέροντα

Collaborative Filtering

- Δοσμένου ενός χρήστη u, κατηγοριοποιούμε τους άλλους χρήστες βασιζόμενοι στην ομοιότητα με τους u_1, u_2, …, u_m
- Στη συνέχεια εκτιμούμε τις προτιμήσεις του χρήστη βασιζόμενοι στις προτιμήσεις των m χρηστών
- Οι προτιμήσεις θεωρούμε πως αφορούν ένα σύνολο αντικειμένων o_1, o_2, …, o_n
- Μπορούμε να απεικονίσουμε τις προτιμήσεις σε ένα πίνακα



Collaborative Filtering

- Στην τεχνική δεν λαμβάνεται υπόψιν το περιεχόμενο κάθε αντικειμένου
- Λαμβάνει υπόψιν της μόνο τις συσχετίσεις των χρηστών
- Αυτό το χαρακτηριστικό κάνει πιο γενική την τεχνική αφού μπορεί να χρησιμοποιηθεί για οποιοδήποτε αντικείμενο και όχι μόνο για έγγραφα
- Χρειαζόμαστε όμως τα ratings των χρηστών
- Φυσικά, στον πίνακα μπορεί να υπάρχουν κενά που σημαίνει πως για το συγκεκριμένο συνδυασμό χρήστη – αντικείμενο δεν έχουμε κάποιο rating
- Η ‘συμπλήρωση’ των κενών είναι ο στόχος της τεχνικής

Collaborative Filtering

- Στην περίπτωση που δεν έχουμε αρκετά ratings / προτιμήσεις των χρηστών τότε έχουμε το λεγόμενο cold start problem
- Υιοθετούμε μια συνάρτηση f() που αποτυπώνει για ένα χρήστη και ένα αντικείμενο σε ένα rating
- Θέλουμε να συμπεράνουμε το αποτέλεσμα της συνάρτησης στις περιπτώσεις που δεν έχουμε τα ratings
- Όταν εστιάζουμε σε ένα χρήστη, προσπαθούμε να βρούμε τους χρήστες που είναι όμοιοι με αυτόν
- Από τους όμοιους χρήστες και τα ratings τους εξάγουμε τις προτιμήσεις του εξεταζόμενου χρήστη

Collaborative Filtering

- Έστω ότι ο εξεταζόμενος χρήστης είναι ο u_a
- Έστω ότι θέλουμε να προτείνουμε το αντικείμενο o_j
- Η εκτίμηση του rating είναι ο συνδυασμός των κανικοποιημένων ratings των όμοιων χρηστών
- Εξετάζουμε το άθροισμα των ratings αλλά δεν συνεισφέρουν με τον ίδιο τρόπο οι χρήστες
- Σε κάθε χρήστη ανατίθεται ένα βάρος
- Όσο πιο όμοιος είναι ένας χρήστης με τον εξεταζόμενο χρήστη τόσο περισσότερο συνεισφέρει

Collaborative Filtering

- Αρχικά, υπολογίζουμε το κανονικοποιημένο rating

  $V_{ij} = X_{ij} - n_i$

- n_i είναι το μέσο rating για όλα τα αντικείμενα του χρήστη u_i
- Η εκτίμηση του rating είναι:

$$\hat{V}_{aj} = k \cdot \sum_{i=1}^{m} w(u_a, u_i) \cdot V_{ij}$$

$$w_p(u_a, u_i) = \frac{\sum_j (X_{aj} - n_a)(X_{ij} - n_i)}{\sqrt{\sum_j (X_{aj} - n_a)^2 \sum_j (X_{ij} - n_i)^2}}$$

- k είναι μια παράμετρος κανονικοποίησης

$$k = \frac{1}{\sum_{i=1}^{m} w(u_a, u_i)}$$

- Μετασχηματίζουμε στο διάστημα των ratings του συγκεκριμένου χρήστη

$$\hat{X}_{aj} = \hat{V}_{aj} + n_a$$

Collaborative Filtering

- Ως συνάρτηση ομοιότητας μπορούμε να χρησιμοποιήσουμε την Pearson Correlation Coefficient

$$w_p(u_a, u_i) = \frac{\sum_j (X_{aj} - n_a)(X_{ij} - n_i)}{\sqrt{\sum_j (X_{aj} - n_a)^2 \sum_j (X_{ij} - n_i)^2}}$$

- Άλλη μετρική είναι η cosine measure

$$w_c(u_a, u_i) = \frac{\sum_j x_{aj} x_{ij}}{\sqrt{\sum_j x_{aj}^2 \sum_j x_{ij}^2}}$$