Tiedonhakumenetelmät

Helena Ahonen-Myka

Kevät 2004, osa 6

Täsmäytysmenetelmät (2/2)

Tässä osassa

- dokumenttien välinen samanlaisuus
- relevanssipalautteen käyttäminen kyselyn muokkauksessa
- ryvästyksen käyttö haussa

Dokumenttien välinen samanlaisuus

- kahden dokumentin i ja j välistä samankaltaisuutta voidaan kuvata kosinimitalla $\cos(D_i,D_j)$
- lasketaan dokumentti-termi –matriisissa kaikkien dokumenttien keskinäiset samankaltaisuudet
  - → dokumentti-dokumentti –matriisi (koko n x n, jos n dokumenttia)
- dokumentti-dokumentti –matriisi on symmetrinen, joten riittää esittää lävistäjän toinen puoli

dokumentti-termi -matriisi

dok.			ter	mit				
vekt.	a	b	c	d	e	f	g	h
D1	1	0	1	1	0	1	0	1
D2	0	1	1	0	0	1	1	0
D3	0	0	0	1	1	0	1	0
D4	1	1	0	0	1	0	0	1

dokumentti-dokumentti -matriisi

doku		doku	mentit	
mentit	D1	D2	D3	D4
D1	1	0.45	0.26	0.45
D2		1	0.29	0.22
D3			1	0.26
D4				1

Dokumenttien välinen samanlaisuus

- samankaltaiset dokumentit voidaan ryhmitellä ryppäiksi eli klustereiksi (cluster)
  - on määriteltävä kynnysarvo samankaltaisuudelle: minkä muiden dokumenttien kanssa dokumentti sijoitetaan samaan ryppääseen
  - dokumentti voi yleensä kuulua myös useampaan kuin yhteen ryppääseen

Keskusalkiovektori

- ryvästä voidaan kuvata tiivistetysti joukon keskusalkion (centroid) avulla
- keskusalkio = joukon vektoreiden keskimääräinen vektori
- cen ($\{D_1,...,D_m\}$) = (w_1*, ..., w_t*), missä kukin w_i^* (i = 1,...,t) on vastinkomponenttien keskiarvo:

$$w_i^* = (1/m)\sum_{j=1}^{m} w_{ji}$$

7

Keskusalkiovektorin laskenta

D1	1	0	1	1	0	1	0	1
D2	0	1	1	0	0	1	1	0
D4	1	1	0	0	1	0	0	1
summa	2	2	2	1	1	2	1	2
keskusalkio	0.67	0.67	0.67	0.33	0.33	0.67	0.33	0.67

8

Relevanssipalaute

- käyttäjän on vaikea antaa hyviä hakutermejä
  - erityisesti, jos käyttäjä ei tunne dokumenttikokoelman dokumenttien sisältöä
- on tyypillistä muokata hakua hakuprosessin aikana
  - aluksi annetaan ensin mieleen tulevat hakutermit
  - kun on nähty haun tulos, tarkennetaan hakua
- järjestelmä voi muokata hakua myös automaattisesti **relevanssipalautteen** (relevance feedback) avulla

9

Relevanssipalaute

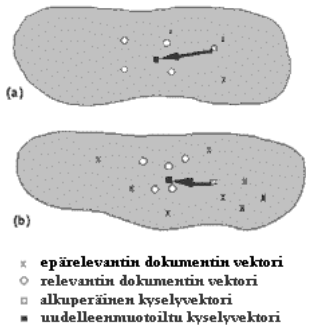
- perusoletus: haun kannalta relevantit dokumentit ovat samankaltaisia
- idea:
  - käyttäjä on tarkastellut k kpl hakutuloksena saatua (parasta) dokumenttia ja tunnistanut niistä osan relevanteiksi ja osan epärelevanteiksi
  - muokataan kyselyvektoria niin, että siitä tulee samankaltaisempi tunnettujen relevanttien dokumenttien kanssa ja erilaisempi epärelevanttien dokumenttien kanssa
  - oletetaan, että uusi kysely löytää lisää relevantteja dokumentteja ja/tai nostaa relevanttien dokumenttien paikkaa tuloslistassa

10

Relevanssipalaute

- seuraavan kalvon kuvassa (a) relevanttien ja epärelevanttien dokumenttivektorien välillä on erottavia piirteitä
  - relevanssipalautteella voidaan helposti siirtää kyselyvektori lähemmäs relevanttien dokumenttien ryppään keskusalkiota
- kuvassa (b) relevanttien dokumenttien joukko on tiivis ja epärelevanttien dokumenttien joukko on hajanainen
  - relevanssipalaute toimii myös tässä tapauksessa

11



12

Relevanssipalaute

- jos relevanttien ja epärelevanttien dokumenttien vektorit ovat sekoittuneet tasaisesti kyselyvektorin ympäristöön, relevanssipalaute ei toimi
- relevantit dokumentit eivät välttämättä muodosta yhtenäistä ryvästä
  - tunnetut relevantit dokumentit voidaan ryvästää ja näin tunnistaa tilanne
  - jos ryppäitä on useita, voidaan kysely jakaa useampaan osaan

13

Relevanssipalaute

- kyselyä voidaan iteroida useita kertoja
  - käyttäjä arvioi aina uusia dokumentteja
  - → tunnettujen relevanttien ja epärelevanttien dokumenttien määrä kasvaa joka kierroksella
- prosessi päättyy esimerkiksi,
  - kun käyttäjä on tyytyväinen tulokseen (tai väsyy antamaan lisää palautetta…) tai
  - kun tietty määrä kierroksia tulee täyteen tai
  - kun kyselyn muokkaus ei enää suuresti vaikuta hakutuloksen kokoon ja/tai järjestykseen

14

Relevanssipalaute

- relevanssipalautteen käyttö on todettu tehokkaaksi menetelmäksi
  - jo yhdellä iteraatiolla keskimääräisen tarkkuuden parannus voi olla 40-60%
- tehokkuus perustunee siihen, että relevanssipalautteen avulla saadaan selville lisätietoa käyttäjän tiedontarpeesta
  - ja käytetään tätä tietoa lisäämällä tiedontarpeeseen liittyviä termejä ja muuttamalla termien suhteellista tärkeyttä
  - käyttäjien kyselyt ovat tyypillisesti hyvin lyhyitä

15

Relevanssipalaute

- Millä periaatteella kyselyvektoria pitäisi siirtää relevanttien dokumenttien vektoreita kohti (ja poispäin epärelevanteista dokumenteista)?
- ajatellaan, että tiedettäisiin, mitkä kokoelman dokumentit ovat haun kannalta relevantteja ja mitkä eivät
- voidaan muotoilla optimaalinen kysely, joka
  - maksimoi kyselyn samanlaisuuden relevanttien dokumenttivektorien kanssa ja
  - minimoi kyselyn samanlaisuuden epärelevanttien dokumenttivektorien kanssa

16

Relevanssipalaute

- optimaalinen kysely voitaisiin kuvata esim. kaavalla

$$Q_{opt} = \frac{1}{R}\sum_{D_i \in Rel} \frac{D_i}{|D_i|} - \frac{1}{N-R}\sum_{D_i \in Nonrel} \frac{D_i}{|D_i|}$$

- missä R on relevanttien dokumenttien määrä, N-R epärelevanttien dokumenttien määrä, Rel relevanttien dokumenttien joukko ja Nonrel epärelevanttien dokumenttien joukko

17

Relevanssipalaute

- optimaalista kyselyä ei tietysti oikeasti osata muodostaa
  - jos tiedettäisiin, mitkä dokumentit ovat haun kannalta relevantteja, ei koko hakua tarvitsisi tehdä
- alunperin annettua kyselyä voidaan kuitenkin muokata niin, että se kullakin iterointikierroksella lähestyy optimaalista kyselyä
  - kun käyttäjä antaa relevanssipalautetta, saadaan selville aina suurempi osajoukko R' relevanteista dokumenteista R ja osajoukko N' epärelevanteista dokumenteista N-R

18

Relevanssipalaute

- kyselyä voidaan muokata lisäämällä relevanttien dokumenttivektoreiden termit ja vähentämällä epärelevanttien dokumenttivektoreiden termit

$$Q^{i+1} = Q^i + \frac{1}{|R'|}\sum_{D_i \in R'} D_i - \frac{1}{|N'|}\sum_{D_i \in N'} D_i$$

19

Relevanssipalaute

- vaihtoehtoisesti voidaan määritellä kertoimet α ja β kuvaamaan haluttua relevanttien ja epärelevanttien dokumenttien merkityksen suhdetta:

$$Q^{i+1} = Q^i + \alpha\sum_{D_i \in R'} D_i - \beta\sum_{D_i \in N'} D_i$$

- $\alpha = \beta = 0.5$ → yhtä tärkeitä
- $\alpha = 1$ ja $\beta = 0$ → vain relevantit dokumentit otetaan huomioon

20

Relevanssipalaute

- voidaan myös ottaa huomioon kaikki käyttäjän tunnistamat relevantit dokumentit, mutta vain kaikkein korkeimmalle hakutuloksessa nousseet epärelevantit dokumentit

21

Relevanssipalaute, muunnelmia

- kun käyttäjä arvioi kokonaisten dokumenttien relevanssia, voi syntyä ongelmia
  - relevantit dokumentit voivat sisältää myös paljon termejä, jotka eivät liity tiedontarpeeseen
    - silti nämäkin vaikuttavat uuteen kyselyyn
  - relevanssipalaute ei voi muuttaa alkuperäistä kyselyä kovin paljon: jos kysely ei palauttanut juuri lainkaan relevantteja dokumentteja, ei relevanssipalautteen käyttö muuta tilannetta
- voitaisiin myös kysyä käyttäjältä, mitkä termit tai fraasit tuntuvat tärkeimmiltä, ja käyttää vain näitä
  - vaatii kuitenkin enemmän vaivaa käyttäjältä

22

Relevanssipalaute, muunnelmia

- relevanssipalautteen ideaa voidaan soveltaa myös silloin, kun käyttäjää ei ole käytettävissä (tai ei tahdota vaivata)
- valitaan hakutuloksesta esim. 10 parasta dokumenttia ja kerätään niistä kaikki termit
- järjestetään termit yleisyysjärjestykseen ja poistetaan hukkasanat
- käytetään kyselyn laajennuksena esim. 10 parasta termiä

23

Dokumenttivektoreiden muokkaus

- relevanssipalautetta käytettäessä muokataan kyselyvektoria
- voitaisiin myös yrittää vaikuttaa dokumenttiavaruuteen → muokataan indeksointia dynaamisesti
- jos joukko dokumentteja on tunnistettu relevanteiksi jonkin kyselyn kannalta, muokataan näiden dokumenttien vektoreita lähemmäs kyselyvektoria
- samalla dokumenttivektorit lähestyvät myös toisiaan ja palautuvat helpommin yhdessä

24

Dokumenttivektoreiden muokkaus

- koska relevanssiarviot ovat aina subjektiivisia, kannattaa dokumentteja siirtää vain hiukan yhtä kyselyä kohti
- suurempia muutoksia syntyy vain, jos useat käyttäjät ovat olleet samaa mieltä
- dokumenttivektoreita muokkaamalla voidaan usein haettuja dokumentteja painottaa ja harvoin haettujen merkitystä heikentää
  - harvoin haetut dokumentit voidaan "siirtää eläkkeelle", jos dokumenttikokoelman kokoa halutaan säädellä ja uusia dokumentteja lisätään usein

25

Ryvästyksen käyttö haussa

- samankaltaiset dokumentit voidaan koota ryppääksi
- ryppäitä voidaan käyttää apuna kyselyä käsiteltäessä
- ryppäät muodostetaan harvoin → ryvästäminen (klusterointi) voi olla raskaampi operaatio
- kyselyn suorittamisen aikana ryppäitä pitää pystyä käsittelemään nopeasti

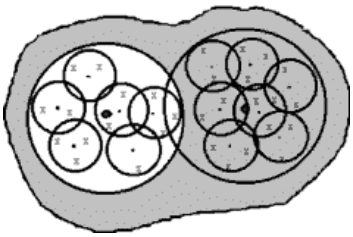
26

Ryvästyksen käyttö haussa

- kullekin dokumenttiryppäälle lasketaan keskusalkio
  - keskusalkio edustaa tiivistetysti rypästä
- keskusalkioiden samankaltaisuutta voidaan vuorostaan vertailla keskenään → korkeamman tason ryppäitä, joilla on taas omat keskusalkionsa
- → muodostuu ryväshierarkia
  - ylimmällä tasolla yksi tai muutama ryväs
  - dokumenttien termivektorit sijaitsevat alimman tason ryppäissä
  - ryppään dokumentit myös talletetaan ryppäänä → haku nopeaa

27

Hierarkkinen ryvästys



28

Haku ryvästetystä kokoelmasta

- kyselyvektoria verrataan ensin ylimmän tason ryppäiden keskusalkioihin
- kaikki ne ryppäät, joiden keskusalkioiden samanlaisuusarvo ylittää kynnysarvon, otetaan tutkittavaksi
- verrataan edelleen näiden ryppäiden aliryppäiden keskusalkioita kyselyvektoriin
- jatketaan, kunnes ollaan päästy alimman tason ryppäisiin
- näiden ryppäiden dokumentit palautetaan hakutuloksena

29

Tässä osassa

- samankaltaiset dokumentit voidaan ryhmitellään samaan ryppääseen
- ryvästystä voidaan hyödyntää kyselyn käsittelyssä
- relevanssipalautteen perusteella voidaan kyselyä muokata relevanttien dokumenttien muodostamia ryppäitä kohti ja poispäin epärelevanttien dokumenttien ryppäistä
- hierarkkista ryvästystä voidaan käyttää suoraan haussa

30