Hyväksymispäivä

Arvosana

Arvostelija

Esitelmä
Esitelmöitsijä: Sampsa Ahonen

Tiedonhaku ja hyperteksti

Helsingin yliopisto
Tietojenkäsittelytieteen laitos
Seminaari: Hypermediajärjestelmät
Esitelmä
Helsinki 5.11.2004
Ohjaaja: Hannu Erkiö

Tekijä: Sampsa Ahonen
Työn nimi: Tiedonhaku ja hyperteksti
Oppiaine: Tietojenkäsittelytiede
Työn laji: Esitelmä
Aika: 5.11.2004
Sivumäärä: 3 + 22

Tiivistelmä:

Hypertekstuaalisten dokumenttien tietorakennetta voidaan suoraan hyödyntää tiedonhaussa. Dokumenttien sisältöä voidaan hyödyntää esimerkiksi dokumentin termien kautta. Hypertekstin sisältämää linkkitopologiaa voidaan hyödyntää samalla periaatteella kuin tieteellisiä julkaisuja arvotetaan sen mukaan kuinka paljon muut tieteentekijät ovat siteeranneet julkaisua. Tiedonhaku tapahtuu muuntamalla kysely kyselynkuvaajaksi, esimerkiksi kyselyvektoriksi, ja tulkitsemalla dokumentti sopivaksi dokumentinkuvaajaksi, esimerkiksi dokumenttivektoriksi. Vastaus kysellyn saadaan täsmäyttämällä kyselynkuvaaja ja dokumentinkuvaaja. Jos halutaan välttää aiheeton tyhjä tulosjoukko, on useimmiten tyytyminen osittaistäsmäytykseen. Täystäsmäytysalgoritmissa nojataan Boolen algebraan ja esityksessä käsitellään lyhyesti vastaava Boolen haku. Osittaistäsmäytysalgoritmeja on useita. Lähes kaikkien niiden takana on kova matematiikka. Yksityiskohtaisemmin esityksessä käsitellään HITS-algoritmi, PageRank-algoritmi ja dokumenttivektorimalli. Dokumenttivektorimalli käsitellään osana termein erottelukykyä selvittävän TF-IDF – kaavion esittelyä. Lopuksi esitelmöijän omana näkemyksenä esitetään, että ontologiat ja semanttinen web ovat hypertekstuaalisen avaruuden ja tiedonhaun välitön jatkumo. Sen jälkeen on taas tieteelliseltä pohjalta esitelty hyvin lyhyesti ontologiaperustaista verkkoa ja semanttisen web´in todennäköistä kehittymistä myös kaupallisesti merkittäväksi WWW:n osaksi.

ACM Computing Classification System (1998):
H. Information Systems
H.5 Information interfaces and presentation
H.5.4 Hypertext/Hypermedia: Architectures, Navigation, Theory

Avainsanat: hyperteksti, tiedonhaku, täsmäytys, indeksointi, Boolen operaattorit, pageRank, dokumenttivektori, TF-IDF

Säilytyspaikka:

Muita tietoja:

Sisällys

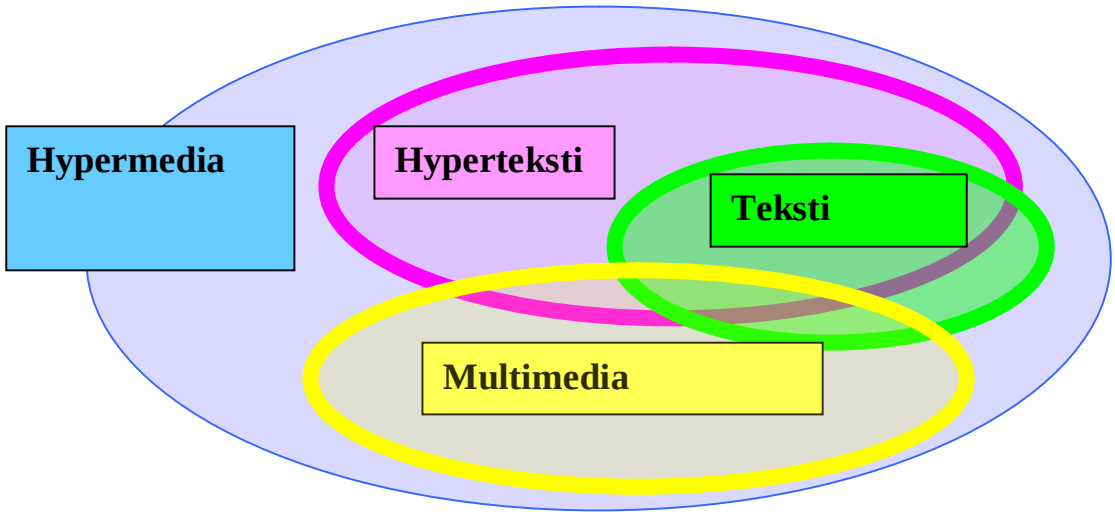
1 Johdanto

Hyperteksti on eräs tapa esittää informaatiota. Tiedonhaussa hypertekstistä tätä erityistä informaation esitystapaa hyödynnetään. Seuraava johdattelu perustuu notoriseen tietämykseen sekä seminaarin avausluentoon, tiedonhakumenetelmien laudaturkurssin luentorunkoon ja hypermedian peruskurssin luentorunkoon [Erk04], [Aho04], [Huh04]. Lisäksi viitataan muutamaan erityislähteeseen. Hypertekstissä informaatio on esitetty joukkona toisiinsa yhdistettyjä informaatioalkioita [BrP98]. Informaatio on tekstiä, mutta myös muu muoto, esimerkiksi kuva, tulee kyseeseen. Tiedonhaun kohteena on dokumentti, dokumenttijoukko tai dokumentin osa. Graafiteorian näkökulmasta dokumentti voidaan samaistaa solmuun. Dokumentit yhdistävää elementtiä kutsutaan linkiksi. Etuliite hyper tarkoittaa yli, hyperteksti tarkoittaa tekstiä ja siihen liitettyä lisäinformaatiota, metatietoa. Esimerkiksi html-merkkauskielessä hypertekstin alku ”<h1>alku” kertoo, että kyseessä ei ole vain merkkijono ”alku” vaan merkkijono muodostaa pääotsikon. Merkkaus <img SRC="omakuva.png” ALT="omakuva"> tarkoittaa, että esitetään omakuva-niminen kuva ja vain vaihtoehtoisesti omakuva-merkkijono. Hypertekstin alku ”<LINK” taas kertoo, että kyseessä on linkki ja kyseessä ei ole varsinainen teksti ollenkaan tai teksti on toissijainen. Tiedonhaku saatetaan kohdentaa lisäinformaatioon [Kle99]. Tiedonhaku yleensä poikkeaa tavallisesta tekstin, esimerkiksi kirjan lukemisesta: ei lueta lineaarisesti vaan selataan. Tiedonhaun hallinnan parantamiseksi hypertekstiin tai sen ohelle rakennetaan abstraktioita, esimerkiksi karttoja tiedoista, ja esitystä visualisoidaan. Satunnaista selaamista kehittyneempi tiedonhaku tapahtuu mentaalisen koordinaatiston avulla, esimerkiksi hakijan aiemman kokemuksen avulla tai nykyisen tiedon ja intuition avulla. Tietoalkiot ja linkit voivat itseisarvotietonsa lisäksi ohjata tiedonhakua [LiC99]. Kun dokumentti on hypertekstuaalinen, tiedonhakua voidaan automatisoida algoritmeilla ja matemaattisella laskennalla.

2 Hypertekstin asema tiedonhakuavaruuksissa ja tiedonhaun pseudokaavio

Hypertekstimäisen tiedonhaun lähikäsite on hypermedia ja lähimaailma on usein WWW-verkko. Kuvassa 1 on esitetty hypermedian peruskurssin näkemys hypertekstin käsitteen sijoittumisesta suhteessa trendikkäisiin lähikäsitteisiin [Huh04]. Kaikki teksti siis ei sisällä lisäinformaatiota tekstin rakenteesta eikä siten ole hypertekstiä. Toisaalta uusimmatkaan

median muodot eivät välttämättä tarjoa sellaista sisältöä, jota voitaisiin välittömästi hyödyntää tiedonhaussa. Hyödynnettävissä olevat epätekstiset tiedonhakumenetelmät, kuten kuvahahmontunnistus, jäävät tämän esitelmän ulkopuolelle. Lisäämällä kuviin, ääneen tai virtuaaliobjekteihin metatietoa eli tekstuaalista tietoa saadaan nämäkin kohteet hypertekstin piiriin. Korostettakoon vielä sitä, että hyperteksti ja WWW eivät ole sama asia. WWW on hypertekstin laajin käytössä oleva hakuavaruus.

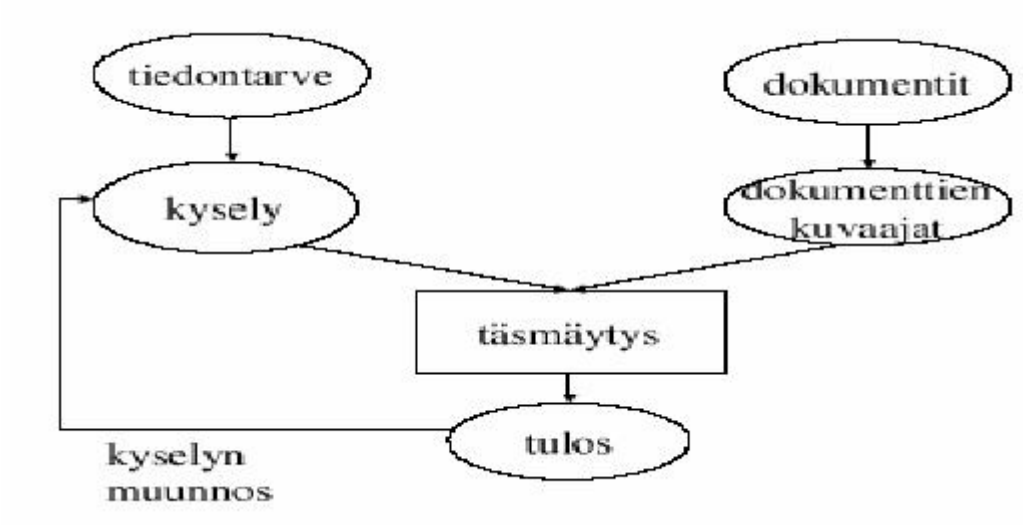


Kuva 1. Hypertekstin käsitteen suhde eräisiin lähikäsitteisiin.

Tiedonhaku taas sijoittuu informaatiotutkimuksen piiriin: Informaatiotutkimus tarkastelee tiedon välittymistä tiedon tuottajien ja tiedon käyttäjien välillä. Informaatiotutkimuksessa tutkitaan yhteisöjen ja yksilöiden tiedonkäytön ympäristöjä, tiedontarpeita ja hankintatapoja sekä tiedon organisointia. Keskeinen tavoite on tutkia ja kehittää käsitteitä, menetelmiä ja järjestelmiä, joiden avulla tarpeellinen tieto saadaan käyttöön mahdollisimman vaivattomasti. Tiedon tallennus tietokantoihin ja tiedonhaku niistä on yksi informaatiotutkimusta erityisesti kiinnostava tapa organisoida tietoa. Tiedon tallennus ja haku (information storage and retrieval), tai lyhyemmin tiedonhaku (information retrieval, IR) on informaatiotutkimuksen yksi keskeinen tutkimusalue [JäK02]. Tässä esityksessä rajoitutaan tietojenkäsittelytieteen näkökulmaan, siis lähinnä algoritmiseen ja matemaattiseen näkökulmaan.

Tiedonhaussa pyritään automatisointiin, koska käyttäjän muisti ja havainnointikyky ovat rajalliset. Automatisointi tehdään algoritmeilla. Käyttäjälle haku tarjotaan tyypillisesti visuaalisella käyttöliittymällä, jonka takana on hakukone. Hakukoneet hyödyntävät

dokumenttien sisältöä tai dokumenttien välisiä linkkirakenteita. Käyttäjä joutuu kuitenkin usein täydentämään hakuaan manuaalisella navigoinnillaan. Esimerkiksi tämän esitelmän hypertekstinen rakenne on, että esitelmä jakautuu tiivistelmään, sisällyssivuun ja kuuteen lukuun ja lähteet-sivuun. Hypertekstille suunniteltu ja toteutettu esitysmuoto kuitenkin vaikuttaa osumiin haussa: sisällön luvut voivat esimerkiksi kukin osoittaa erilliseen lukunsa dokumenttiin tai dokumentti voi esiintyä yhtenä tulosdokumenttina, jonka sisällä on linkkejä. Tiedonhaun pääpiirteet esittää kuvan 2 pseudokaavio.



Kuva 2. Pseudokaavio tiedonhakuprosessista [Aho04].

Tiedontarve puetaan kyselyksi ja dokumentit muunnetaan algoritmeilla käsittelykelpoiseen muotoon laatimalla dokumenttien kuvaajat. Kyselyjen ja dokumenttien esitysmuotojen vuoksi vapaamuotoisten, luonnollisella kielellä esitettyjen kyselyjen ja tekstidokumenttien vertailu on hankalaa. Tästä syystä sekä kyselyjen että dokumenttien esitysmuoto on muokattava sopivammaksi. Usein tarkasteltavana on joukko termejä, termillä tarkoitetaan tässä esityksessä semanttisen ilmaisun yksikköä, esimerkiksi sanaa, fraasia tai sanan vartaloa. Kyselyjen esitysmuodot ovat esimerkiksi joukko hakutermejä tai lauseke, jossa hakutermejä on yhdistetty operaattoreilla. Dokumentti voidaan kuvata automaattisesti siitä tilastollisin perustein valittujen termien avulla tai automaattisesti siitä lingvistisin perustein valittujen ja muokattujen termien avulla tai ihmisen valitsemien termien avulla. Dokumenttien kuvaajat tehdään yleensä indeksoimalla ja valitsemalla joukko termejä, jotka otetaan mukaan kuvaajaan. Myös hakemiston rakentaminen ja tallettaminen eli hakutietorakenteen implementointi on osa indeksointia. Indeksin suomenkielinen vastine tässä yhteydessä onkin yksinkertaisesti hakemisto [Aho04].

Indeksi esittää siis joukon dokumenttien kuvaajia ja myös hakutietorakenteen. Kyselyiden kuvaajat muodostetaan pääosin samoilla periaatteilla kuin dokumenttien kuvaajat. Nykyisin indeksointi tehdään useimmiten automaattisesti, tukena käytetään kontrolloitua sanastoa. Indeksin tehokkuutta säätelee kaksi parametria: indeksoinnin tyhjentävyys (indexing exhaustivity) ja termien spesifisyys (term specificity) [Aho04]. Tiedonhaun loppuvaihe käsittää vertaamisen: katsotaan, mitkä dokumentit termeiltään täsmäävät kyselyn hakutermien kanssa. Täsmäytyksellä – täydellisellä, osittaisella tai likimääräisellä – pyritään siihen, että tulosjoukko ei jää tyhjäksi. Tarvittaessa kyselyä muunnetaan. Tiedonhakua auttaa huomattavasti, jos tiedon tallentaja on nähnyt vaivaa. Hyvä tekstitiedoston rakennustapa edellyttää, että on muodostettu myös käänteisrakenne. Käänteisrakenteen muodostamisperiaate on seuraava: 1) Perustiedoston jokaisen tietueen tietyssä tai tietyissä kentissä esiintyvät hakuavaimet poimitaan yhdessä tietueen numeron kanssa listaksi, joka muodostuu (avain, tietuenumero)- pareista. 2) Seuraavaksi lista lajitellaan parien hakuavainten mukaan nousevaan järjestykseen. Saman avainarvon esiintymät tulevat siis peräkkäin kukin oman tietuenumeronsa kanssa. 3) Lopuksi yhdistetään saman avaimen sisältävät parit siten, että yhteiseen avainarvoon liitetään kaikki eri tietuenumerot lajiteltuna listana. Tuloksena on käänteistiedosto (hakemisto, basic index, inverted file), jossa kustakin hakuavaimesta kerrotaan sen kaikkien esiintymien osoitteet tiedoston eri tietueissa [JäK02].

Kolmannessa luvussa käsitellään hakua sisällön tai linkkien perusteella ja osittain siihen liittyvää matematiikkaa. Neljännessä luvussa esitetään lisää esimerkkejä, miten tiedonhaun automatisointi voidaan nojata puhtaaseen matematiikkaan; luvun aluksi esitetään kattava täsmäytyksen tietomalli. Viidennessä luvussa selvitetään lyhyesti luokituksen eli taksonomian ja ontologioiden eli käsitemallien lisäämistä hypertekstiin - semanttisen eli merkitykseen perustuvan tiedonhaun mahdollistamiseksi; lisäksi esitetään esimerkki hypertekstimäisen tiedonhaun kaupallisesta jatkojalostamisesta WWW-sovelluspalveluksi rajoitettujen, keskitettyjen hakemistorekistereiden (UDDI) avulla. Kuudennessa luvussa esitetään yhteenveto.

3 Haun kaksi päälinjaa

Mielivaltaisen dokumentin haku voidaan perustaa dokumentin sisältöön tai dokumenttien välisiin linkkeihin [LiC99]. Sisältöhaku nojataan etsittyihin, dokumentissa esiintyviin termeihin. Myös lähisivut termipainoineen voidaan ottaa huomioon. Linkkihaussa nojataan

linkkitopologiaan. Taustalla on oletus, että linkkiyhteys osoittaa piilevää sivun tekijän harkintaan perustuvaa yhteyttä eli viitatun sivun arvoa. Kaikki linkit eivät tosin välttämättä ole tällaisia [Erk01], [LSZ02]. Mainittua ajattelua tarvitaan, sillä pelkkään verkon rakenteen analysointiin nojaavat tiedonhakumenetelmät (IR Systems) eivät riitä WWW:n tapauksessa, vaan törmätään raekoon (granularity) ongelmiin; yleensä saadaan liian laaja looginen, atominen dokumentti, jonka fyysinen dokumenttikokoelma on hakijan vaikeasti käsitettävissä [Sug03].

3.1 Sisältöhaku

Kyselyllä haetaan ne solmut eli dokumentit, joissa halutut termit esiintyvät. Dokumenttien näyttöjärjestys tulee tärkeäksi, kun osumien määrä on suuri. Silloin dokumenteille kannattaa antaa painoja termiesiintymien määrän mukaan. Jos haun tulosjoukko uhkaa jäädä tyhjäksi, niin kyselyä kannattaa väljentää hakemalla likimääräisesti oikeita vastauksia. Myös jos hakutulosta näytetään osumien ympäriltä laajemmin, hakija osaa paremmin muokata hakuaan.

Haku helpottuu ja sen laatu paranee, kun solmun ympäristö huomioonotetaan haussa. Luonnollinen oletus on, että linkillä yhdistetyillä sivuilla on muutakin kuin teknistä yhteyttä. Ne ovat ehkä tiedonhaun mielessä lähekkäisiä, tai ne muodostavat yhdessä hakuun paremmin sopivan kokonaisuuden. Toisin sanoen pelkästään termien esiintymät solmussa eivät ratkaise solmun samanlaisuutta kyselyn nähden. Solmun sisäinen paino riippuu termiesiintymistä. Termiesiintymät saadaan summaamalla suoraan lukumäärä tai käyttämällä kaavaa 1:

(kaava 1) $\mathbf{w}_{ij} = \mathbf{tf}_{ij} \times \log(\mathbf{N}/\mathbf{df}_j)$,

missä $\mathbf{w}_{ij}$ esittää termin j painon eli relevanssin dokumentissa i ja $\mathbf{tf}_{ij}$ esittää termin $\mathbf{t}_j$ frekvenssin dokumentissa $\mathbf{d}_i$ ja $\mathbf{df}_j$ esittää niiden dokumenttien lukumäärän, joissa termi $\mathbf{t}_j$ esiintyy, ja **N** on tarkasteltavassa kokoelmassa olevien dokumenttien määrä. Solmun ulkoinen paino ilmaisee jälkeläissolmujen vaikutuksen, joka saadaan esimerkiksi laskemalla näiden solmujen sisäisten painojen summa. Solmun n kokonaispaino saadaan silloin kaavalla 2:

(kaava 2) $\mathbf{W}_n = \mathbf{W}_{n,sis} + \alpha \Sigma \mathbf{W}_{i,kok}$,

missä **W**n on solmun n kokonaispaino, **W**n,sis on solmun n sisäinen paino ja **α** on kerroin, esimerkiksi ½, jolla kertoimella ympäristösolmujen vaikutusta painotetaan; kaikkien solmun n jälkeläissolmujen i kokonaispainot **W**i,kok on ensin laskettu yhteen. Laskutapoja voidaan varioida. Voidaan ottaa jälkeläisten painojen keskiarvo. Voidaan antaa erilaisille linkkityypeille erilaiset kertoimet. Tarvittaessa voidaan antaa painoa myös edeltäjäsolmuille. Myös voidaan rajoittaa etäisyyttä, mihin saakka ympäristö otetaan huomioon, ja joka tapauksessa voidaan rajata huomioonottaminen vain kokoelman rajaan asti, sikäli kuin kokoelma tai solmujoukko on kyetty rajaamaan. Sisällölliselle samanlaisuudelle voidaan antaa merkitystä. Voidaan edetä hierarkiatasoittain. Sykliset rakenteet kuitenkin voivat tuottaa virhepäätelmiä. Tiedonhakua auttaa, jos kokoelman tai sivuston laatija on oheistanut tiedot, miten kokoelma on rakennettu [Erk01], [Rod99].

3.2 Linkkirakenteiden hyväksikäyttö

Linkkirakenteitakin läpikäytäessä lopullinen tavoite on löytää oikeat dokumentit. Dokumentin linkkiympäristö voi kuitenkin kertoa dokumentista ja erityisesti sen tärkeydestä enemmän kuin dokumentin sisältö. Ajatus on sama kuin tieteellisten julkaisujen arvostuksessa: julkaisun sanasto voi kertoa laadusta vähemmän kuin se, kuinka paljon muut tieteentekijät ovat siteeranneet julkaisua. Kun on vastattava väljästi määritettyyn kyselyyn, linkkitopologiaan perustuva solmujen valinta on osoittautunut hyödylliseksi. Ongelmia kuitenkin jää paljon, koska suuresta tulosjoukosta eivät korostu välttämättä relevantit dokumentit. Linkkitopologian hyödyntämisen strategia on tiivistetysti seuraava. Määritellään arvosivu (authority page) eli sivu, johon viitataan paljon; tarkemmin sanottuna siis viittaukset tulevat sivuilta, joilta viitataan monille arvosivuille. Sen jälkeen määritellään napasivu (hub page) eli paljon arvosivuille kohdistuvia linkkejä sisältävä sivu. Arvosivut ja napasivut voidaan määrittää pelkästään linkkiyhteyksiä tutkimalla. Oletuksena taustalla on, että linkkiyhteys osoittaa piilevää sivun tekijän harkintaan perustuvaa yhteyttä eli viitatun sivun arvoa [Klei99].

Etsintätaktiikka nojaa hypertekstin tarkasteluun suunnattuna verkkona G, missä G = (V, E), ja särmä (p,q) ∈ E on linkki sivulta p sivulle q. Tarkastellaan laajaa sivujoukkoa. Solmujen termisisältöä käytetään vain alkukyselyssä. Paikalliset solmut, joilla on sama domain-nimi, tulkitaan teknisesti yhteen liitetyiksi ja niissä linkki ei ilmaise toisen solmun arvosta mitään.

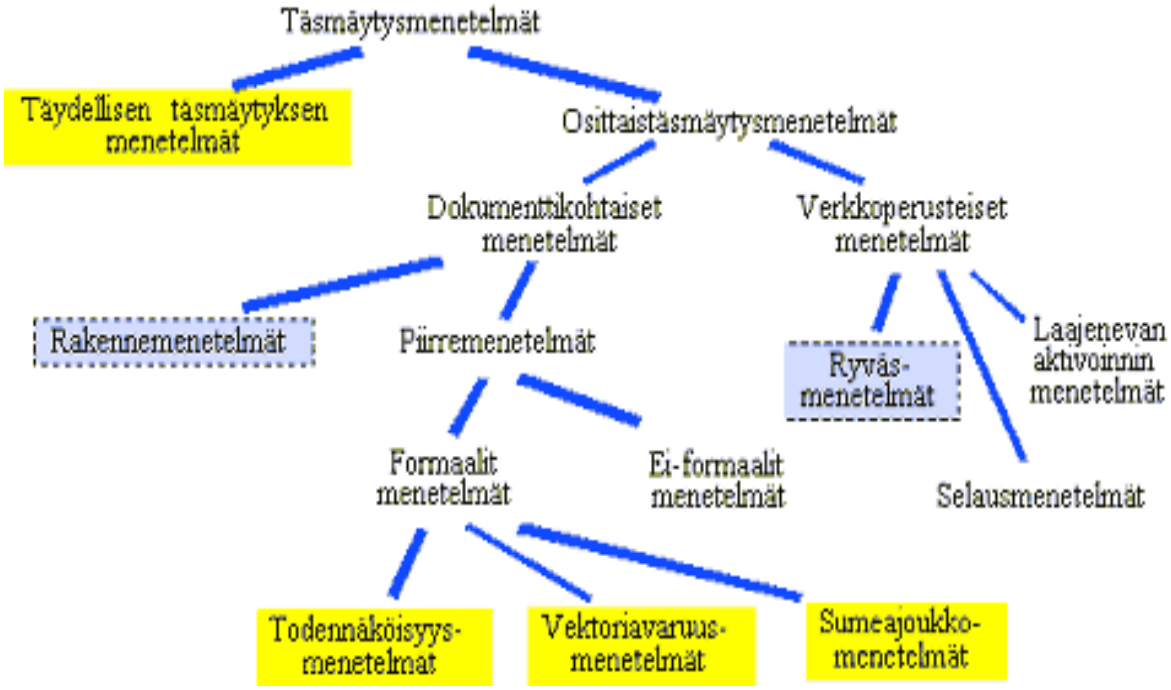
Indeksiä tai sivujen sisältöä, termejä käytetään vain aluksi, tavallaan epäsuorasti. Merkitään muuttujalla out-degree solmusta lähtevien linkkien lukumäärää ja muuttujalla in-degree tulevien linkkien lukumäärää ja merkitään operaatioita, joilla ne saadaan, vastaavasti I:llä ja O:lla. Verkon aliverkkoina otetaan huomioon solmut ja niiden väliset linkit. Määritetään arvo- ja napasivut seuraavasti: 1) Suoritetaan kysely tavanomaisella hakukoneella ja valitaan sen tuloksesta t ensimmäistä sivua juurijoukoksi. 2) Laajennetaan juurijoukko perusjoukoksi liittämällä siihen kaikki ei-paikalliset solmut, joihin viitataan juurijoukon solmusta, ja osa juurijoukon solmuihin viittaavista ei-paikallisista solmuista. 3) Arvo- ja napasivut määritetään soveltamalla linkkejä summaavia operaatioita I ja O iteratiivisesti. Aivan haitaton menetelmä ei ole: Lopulliseen tulokseen voi linkkiyhteyksien kautta tulla solmuja, jotka eivät kuulu alkukyselyn tulokseen ja joissa ei ehkä ole kyselytermin esiintymiä lainkaan. Mukaan voi tulla sivuja, joilla on esimerkiksi kuvia, mutta hyvin vähän tekstisisältöä. Saatetaan myös törmätä erillisiin aliverkkoihin, joissa on omat arvo- ja napasivuparinsa mutta vähän yhteyksiä aliverkosta toiseen. Luvun 4 alaluvussa 4 esiteltävä PageRank-algoritmi hyödyntää nimenomaisesti linkkitopologiaa [Erk01], [BrP98].

4 Hypertekstin matematisoinnista tiedonhaun suoraviivaistamiseksi

Kun haetaan tekstistä tai hypertekstistä informaatiota automaattisesti, haetaan todellisuudessa merkkijonoa. Tulosjoukon ollessa suuri hakutermeille ja solmuille tai linkeille on asetettava matemaattisia arvoja, kuten edellisessä luvussa osin jo esitettiin. Matematiikka ja kieli saati ihmisen ajattelu eivät vastaa toisiaan ja siksi matematiikka ei löydä välttämättä oikeita tuloksia. Silloin kyselyn ja dokumenttien täsmäytyksen täydellisyydestä pitää tinkiä ja hyväksyä epätäydellisiäkin, osin hyödyttömiä tulossivuja sisältäviä tulosjoukkoja.

4.1 Täsmäytyksen tietomalli

Täsmäytyksen kattava tietomalli on esitetty kuvassa 3. Tietomalli kattaa menetelmät dokumenttien ja kyselyjen esittämiseen sekä näiden esitysten vertailuun.



Kuva 3. Täsmäytyksen hallinnan tietomalli [JäK02].

Täydellisen täsmäytyksen onnistuminen on harvinaista. Useimmat tiedonhaun menetelmät ovatkin osittaistäsmäytysmenetelmiä. Kuten luvussa 3 havaittiin, voidaan hyödyntää vaihtoehtoisesti tai rinnakkain dokumenttien sisältöjä tai dokumentteja yhdistäviä linkkejä, jotka verkossa ovat. Täydellisen täsmäytyksen menetelmistä esitellään seuraavassa Boolen algebraan perustuvaa menetelmää. Dokumenttikohtaisista menetelmistä käy esimerkiksi edellisessä luvussa 3 esitelty dokumenttien termisisältöön perustuva esimerkki. Sille jatkona esitetään nyt avaruusvektori- eli dokumenttivektorimenetelmä ja sen erikoistapaus TF-IDF-laskentakaavio. Rakennemenetelmiä tai piirremenetelmien ei-formaaleja menetelmiä ei edempänä käsitellä eikä myöskään formaalien menetelmien todennäköisyyslaskentaosiota erikseen. Verkkoperustaisista menetelmistä luvussa 3 tuli esitetyksi linkkitopologiaan perustuva menetelmä. Sille jatkona esitellään nyt PageRank-algoritmia ja sen matematiikkaa, menetelmä sisältää todellisuudessa todennäköisyysjakauman hyödyntämisen. Ryvästäminen eli samantapaisten informaatioalkioiden samaistaminen tarkasteltavalla hierarkiatasolla voi sisältyä osana useaankin menetelmään. Selausmenetelmiä tai laajenevan aktivoinnin menetelmiä ei edempänä tarkastella.

4.2 Boolen haku

Boolen algebraan perustuva kysely käsittää listan termejä, jotka on yhdistetty loogisilla konnektiiveilla AND, OR ja NOT. Vastauksena ovat ne dokumentit, jotka täyttävät kyselyn määrittelemät ehdot. NOT-kyselyt eivät voi esiintyä yksinään, vaan ne toteutetaan AND NOT – kyselyinä. Ongelmista Boolen kyselyissä, kuten muutoinkin täydellisessä täsmäytyksessä mainittakoon seuraavaa: Kyselyyn lähes täsmääviä dokumentteja ei löydetä. Hakujärjestyksen ja tuloksen järjestys on satunnainen. Kyselyitä ei ole helppo muodostaa ja tuloksen koon säätely on vaikeaa. Kyselyyn lähes tai osittain täsmääviä dokumentteja ei löydetä. Tiedontarvetta voidaan harvoin esittää yksiselitteisesti hakuavainten avulla, olisi parempi saada hakutulos dokumenttien todennäköisen relevanssin mukaan laskevassa järjestyksessä. Tästä syystä tiedonhaun tutkimus on suuntautunut paljolti osittaistäsmäytyksen (partial match) menetelmiin, joiden avulla kaikki yllä mainitut ongelmat voidaan ratkaista. Keskeisiä osittaistäsmäytyksen menetelmiä ovat vektorimalliin (vector space model) ja todennäköisyyslaskelmiin perustuvat menetelmät [Aho04], [JäK02].

4.3 HITS-algoritmi

HITS-algoritmin nimi on lyhenne ilmauksesta hypertext-induced topic selection. Algoritmi olettaa, että hyvä napasivu sisältää dokumentin, joka osoittaa linkein moneen muuhun dokumenttiin ja että hyvä arvosivu sisältää dokumentin, johon monet dokumentit linkein osoittavat. Siten napasivut ja arvosivut muodostavat vastavuoroisesti toinen toisilleen lisää painoarvoa antavan suhteen: parempi napasivu osoittaa moniin hyviin arvosivuihin ja parempi arvosivu osoittaa moniin hyviin napasivuihin. Näin ollen algoritmin tekemiseen tarvitsee koota perusjoukko, joka sisältää juurijoukon ja sen naapuruston tiedot, juurijoukon dokumenttiin sisääntulevat ja siitä ulosmenevät linkit. Algoritmiin liitetään sitten rekursiivinen laskenta mainitun vastavuoroisuuden pohjalta. Erikoistilanteissa HITS-algoritmissa laskenta saattaa konvergoida epärealistiseen suuntaan ja saatavan saaliin laatu ei vastaa todellisuutta [LSZ02].

HITS-algoritmiin onkin kombinoitu neljää muuta hakumenetelmää. Yksi niistä on edempänä alaluvun 4.5 alussa esiteltävä dokumenttivektorimenetelmä (vector space model VSM). Toinen menetelmä on Okapi-menetelmä. Se ottaa huomioon dokumenttikokoelman keskikoon ja arvioitavana olevan dokumentin koon. Kolmas liitännäismenetelmä on tiheän peittävyyden arvostamisen menetelmä (cover density ranking CDR). Sekin, useiden muiden menetelmien tapaan, ottaa huomioon relevanssin, mutta nyt relevanssia annetaan sille, että dokumentit, jotka sisältävät useimmat kyselyn hakutermeistä, rankataan muita korkeammalle.

Neljäs liitännäismenetelmä on kolmitasoisen saalistuksen menetelmä (three-level scoring method TLS). Se nojaa tutkimukseen ihmisestä manuaalisena web-sivujen etsijänä [LSZ02].

4.4 PageRank-algoritmi

PageRank-algoritmi on kaupallistunut tuote. Se kehitettiin Google-hakukonetta ja WWW:tä varten. Idean keksijä kuvaa algoritmin ja laskennan ydintä seuraavasti:

> ”Academic citation literature has been applied to the web, largely by counting citations or backlinks to a given page. This gives some approximation of a page's importance or quality. PageRank extends this idea by not counting links from all pages equally, and by normalizing by the number of links on a page. PageRank is defined as follows:
> We assume page A has pages TI ... Tn which point to it (i.e., are citations). The parameter d is a damping factor which can be set between 0 and 1. We usually set d to 0.85. There are more details about d in the next section. Also C(A) is defined as the number of links going out of page A. The PageRank of a page A is given as follows:
>
> **PR(A) = (1-d) + d (PR(TI)IC(TI) + ... + PR(Tn)IC(Tn))**
>
> Note that the PageRanks form a probability distribution over web pages, so the sum of all web pages' PageRanks will be one.
> PageRank or PR(A) can be calculated using a simple iterative algorithm, and corresponds to the principal eigenvector of the normalized link matrix of the web. Also, a PageRank for 26 million web pages can be computed in a few hours on a medium size workstation. There are many other details which are beyond the scope of this paper [BrP98].

Google-hakukoneen järjestelmäarkkitehtuuri (system anatomy) on julkaistu, samoin osia algoritmista. Merkittävä osa algoritmista on pidetty salassa. Siihen lienee syynä bisnes-näkökulma ja huijausten esto. Viimeksi mainitusta syystä algoritmia lienee aika ajoin muuteltu, muutoinhan kuka tahansa markkinoija voisi antaa itsestään väärän kuvan WWW:ssä järjestämällä sivuja, joilla siteerataan häntä.

4.5 TF-IDF – kaavio

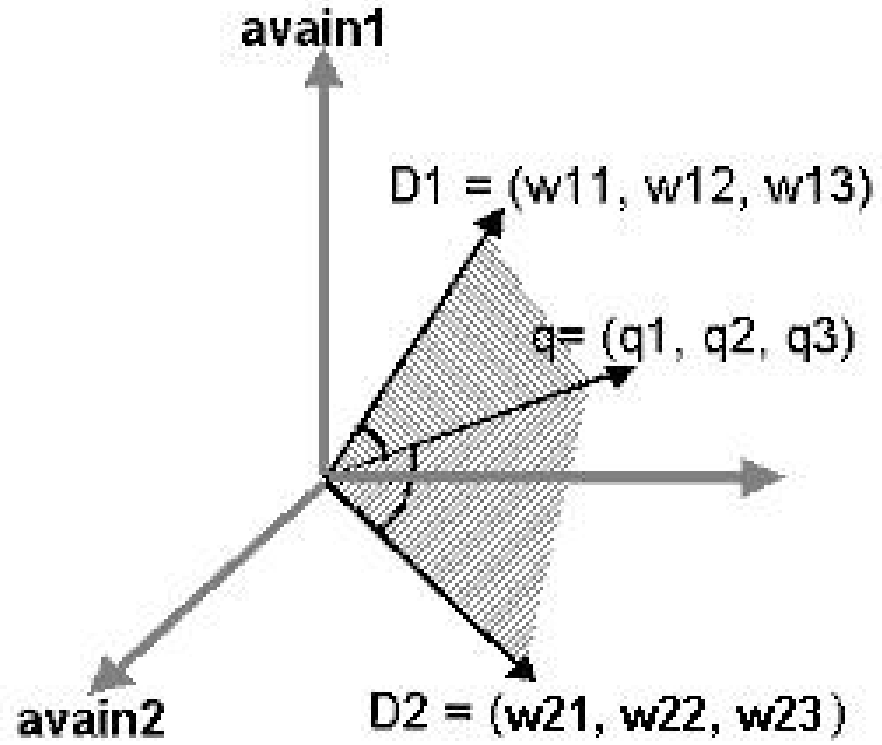
TF-IDF – kaavio on dokumenttivektorimallin eräs sovellus. Dokumenttivektorimalli nojaa siihen, että kysely voidaan esittää q:n hakutermin avaruusvektorina ja dokumentin kuvaaja voidaan muodostaan dokumentin t:n termin avaruusvektorina. Täsmäytyksessä tutkitaan näiden vektoreiden kohtaantoa. Vektoreiden pistetulon suuri skalaariarvo yleensä kertoo hyvästä kohtaannosta, koska samansuuntaisille alkeisvektoreille saadaan osatulo $1 \cdot 1\cos(0 \text{ rad}) = 1$ ja erisuuntaisille alkeisvektoreille osatulo $1 \cdot 1\cos(\pi/2 \text{ rad}) = 0$. Vektorimalli kaavoina ja kuviona voidaan esittää seuraavasti [Aho04].

Vektorimallissa tehdään yksinkertaistava oletus, että termit ovat riippumattomia toisistaan, jolloin dimensiot ovat ortogonaalisia toisiinsa nähden. Määritellään t-dimensioisen kyselyvektorin **q**j ja t-dimensioisen dokumenttivektorin **d**i samanlaisuusfunktio *sim*, joka kuvaa dokumentin ja kyselyn tai kahden dokumentin välistä samanlaisuutta. Se, kuten useat vektorimallissa käytettävät samanlaisuusfunktiot, perustuu vektorien sisätuloon. Sim-funktio on määritelty kaavassa 3:

(kaava 3) $$sim(d_i, q_j) = \sum_{k=1}^{t} d_{ik} \cdot q_{jk}$$

missä funktion parametreina ovat dokumenttivektorin d i:s termivektori ja kyselyvektorin q j:s termivektorivektori ja iteraattori k käy läpi koko t:n laajuisen termiavaruuden.

Vektorimallin termiavaruutta luonnehtii kuva 4. Siinä termiavaruuden dimensio t = 3. Kuvaan on piirretty dokumenttien Di osalta ensimmäisen dokumentin D1 dokumenttivektori, jonka komponentit hakuavainten suunnassa ovat arvoiltaan w11, w12 ja w13 ja vastaavasti dokumentin D2 dokumenttivektori sekä vielä yksi kyselyvektori, jonka komponentit ovat q1, q2 ja q3. Dokumenttivektorien komponenttien arvot w, joita kutsutaan myös painoiksi, voidaan asettaa arvoiltaan myös välille (0,1).



Kuva 4. Havainne-esitys dokumenttivektorimallista. [Aho04]

Jotta laskettaisiin juuri nollilla ja ykkösillä, kosinifunktio pitää normeerata kaavalla 4:

(kaava 4)

$$\cos(d_i, q_j) = \frac{\sum_{k=1}^{t} d_{ik} \cdot q_{jk}}{\sqrt{\sum_{k=1}^{t} (d_{ik})^2 \cdot \sum_{k=1}^{t} (q_{jk})^2}}$$

missä parametrit, muuttujat ja iteraattori ovat samat kuin edellisessä kaavassa [Aho04], [LSZ02].

Hakutermit kannattaa valita siten, että toivotut dokumentit erottuisivat ei-toivotuista. Termin *erottelukykyä* kuvaa käänteinen dokumenttifrekvenssi (inverse document frequency eli *idf*, kaavion nimessä yleensä muodossa IDF). IDF voidaan laskea esimerkiksi seuraavasti: Olkoon dokumenttifrekvenssi dfj niiden dokumenttien lukumäärä, joissa termi Tj esiintyy vähintään kerran. Tällöin käänteinen dokumenttifrekvenssi idf saadaan kaavasta 5:

(kaava 5)

$$idf\ (T_j) = \log \frac{N}{df_j}$$

missä N on dokumenttien kokonaismäärä. Sekä saantia että tarkkuutta voidaan parantaa, kun otetaan huomioon sekä termin frekvenssi dokumentin sisällä (TF) että termin esiintymien jakautuminen eri dokumenttien välillä (IDF) [Aho04], [Sug03].

Dokumenttien kuvaajien käyttökelpoisuutta voidaan vielä lisätä, jos TF-IDF -kaaviota tarkennetaan kuhunkin solmuun linkitetyn naapuruston tietoihin nojaavalla lisälaskennalla. Lisälaskenta kannattaa ulottaa kohdesivusta katsoen kaksi linkkiä taaksepäin olevaan naapurustoon asti. Lisälaskenta on raskasta, mutta kokeellisesti se todettiin mahdolliseksi eikä se kestoltaan heikentänyt hakuprosessia. Seuraavassa esitetään kolme TF-IDF -kaavioon kokeiltua lisäpiirrettä [Sug03].

Merkitään haettavaa kohdedokumenttia eli kohdesivua (target page) tunnuksella p_{tgt}. Olkoon i sivusta p_{tgt} alkavan lyhimmän polun askelten määrä. Olkoon i:nnellä tasolla sivusta p_{tgt} lukien web-sivuja **N**i kappaletta, merkitään näitä sivuja p vastaavilla alaindekseillä. Olkoon termejä t m kappaletta ja merkitään termit vastaavasti alaindekseillä. Tällöin, olettaen että kaikki m termiä t ovat riippumattomia (unique), sivun p_{tgt} dokumenttivektori kuuluu seuraavasti:

$$\boldsymbol{w}^{p_{tgt}} = (w_{t_1}^{p_{tgt}}, w_{t_2}^{p_{tgt}}, \cdots, w_{t_m}^{p_{tgt}})$$

.

Normeerataan vielä TF-IDF -kaavion mukaisesti dokumenttivektorin kuhunkin termiin liittyvää komponenttia (element) seuraavalla kaavalla 6:

(kaava 6)

$$w_{t_k}^{p_{tgt}} = \frac{tf(t_k, p_{tgt})}{\sum_{s=1}^{m} tf(t_s, p_{tgt})} \cdot \log \frac{N_{web}}{df(t_k)}$$

,

missä tf kuvaa kohdesivun p_{tgt} kunkin termin t frekvenssiä sillä sivulla ja N kokoelman web-sivujen kokonaismäärää ja df kuvaa dokumenttifrekvenssiä k:nnen termin osalta.

Jalostetaan sitten kuhunkin termiin liittyvää w-elementtiä ja valitaan dokumenttivektoriksi näistä jalostetuista elementeistä muodostettu vektori. Jalostustoimenpidettä merkittäköön yläpilkulla ´. Silloin jalostettu dokumenttivektori kuuluu seuraavasti:

$$\boldsymbol{w}^{\prime p_{tgt}} = (w_{t_1}^{\prime p_{tgt}}, w_{t_2}^{\prime p_{tgt}}, \cdots, w_{t_m}^{\prime p_{tgt}})$$

.

Ensimmäinen jalostusmenetelmä:

Ensimmäinen jalostusmenetelmä, kuten kaikki kolme vaihtoehtoista jalostusmenetelmää, nojaa lisälaskentaan kohdesivun p_{tgt} naapurustosivujen tiedoista. Merkitään tunnuksella in liikkumista kohdesivusta p_{tgt} taaksepäin linkkiketjua pitkin haluttuun tasoon ja vastaavasti tunnuksella out etenemistä linkkiketjun suunnassa haluttuun tasoon. Merkitään saavutettuja tasoja L (level) vastaavasti $L_{(in)}$ ja $L_{(out)}$.

Tehdään oletus, että kohdesivun p_{tgt} kanssa sisällöltään samanlaisia sivuja on olemassa. Otetaan vielä huomioon, että samanlaiset sivut ovat toisaalta lähellä mutta toisaalta ne on saatettu vektoriavaruuden mielessä siirtää kauaksikin kohdesivusta. Lasketaan nyt kohdesivun p_{tgt} k:nnetta termipiirrettä t_k (k = 1, 2, …, m) vastaavalle elementille $w_{t_k}^{p_{tgt}}$ jalostettu arvo $w_{t_k}^{\prime p_{tgt}}$ seuraavasta kaavasta 7:

(kaava 7)

$$w_{t_k}^{\prime p_{tgt}} = w_{t_k}^{p_{tgt}} + \frac{1}{Dim}\left(\sum_{i=1}^{L_{(in)}}\sum_{j=1}^{N_{i_{(in)}}} \frac{w_{t_k}^{p_{ij_{(in)}}}}{dis(\boldsymbol{w}^{p_{tgt}}, \boldsymbol{w}^{p_{ij_{(in)}}})}\right) + \frac{1}{Dim}\left(\sum_{i=1}^{L_{(out)}}\sum_{j=1}^{N_{i_{(out)}}} \frac{w_{t_k}^{p_{ij_{(out)}}}}{dis(\boldsymbol{w}^{p_{tgt}}, \boldsymbol{w}^{p_{ij_{(out)}}})}\right),$$

missä *Dim* tarkoittaa riippumattomien termien lukumäärää web-kokoelmassa, ja *dis*-funktioiden eli in-suunnan ja out-suunnan etäisyysfunktioiden arvot lasketaan kaavoista 8 ja 9:

(kaavat 8 ja 9)

$$dis(\boldsymbol{w}^{p_{tgt}}, \boldsymbol{w}^{p_{ij(in)}}) = \sqrt{\sum_{k=1}^{m} (w_{t_k}^{p_{tgt}} - w_{t_k}^{p_{ij(in)}})^2},$$

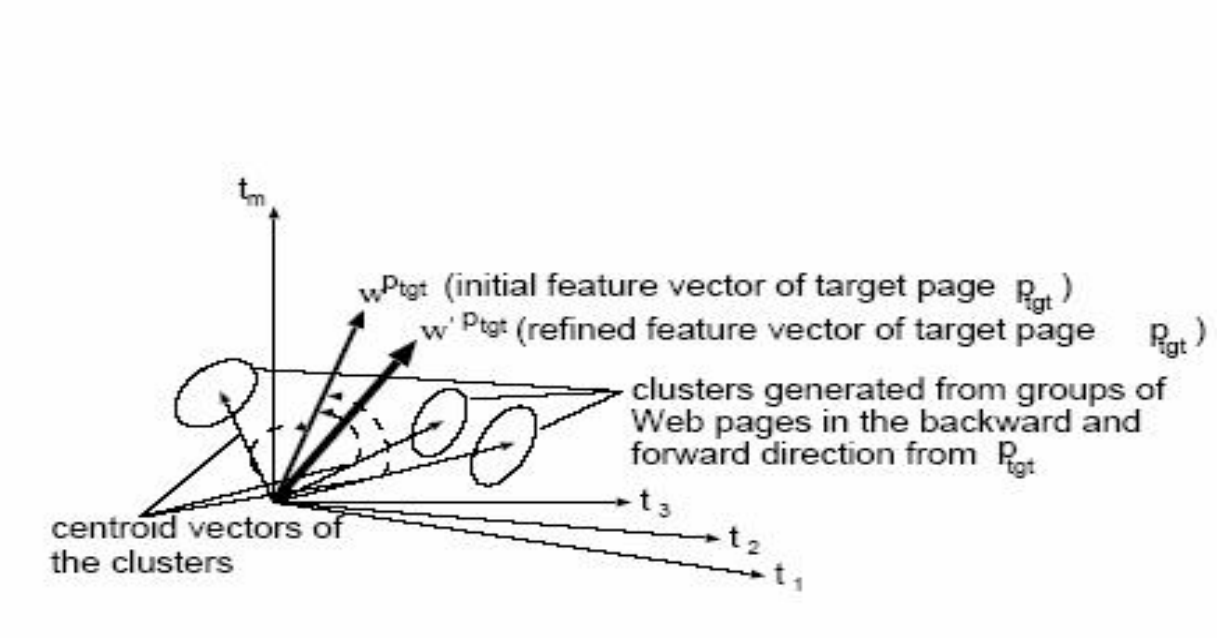
$$dis(\boldsymbol{w}^{p_{tgt}}, \boldsymbol{w}^{p_{ij(out)}}) = \sqrt{\sum_{k=1}^{m} (w_{t_k}^{p_{tgt}} - w_{t_k}^{p_{ij(out)}})^2}.$$

Siis jalostettu termipaino $w_{t_k}^{\prime p_{tgt}}$ saa lisäpainoa summalausekkeista, mutta lisäpainoa normeerataan termikokoelman termien lukumäärällä. Sekä in- että out-suunnassa olevilta tasoilta L otetaan lisää termipainoa. Kullakin tasolla kaikki N web-sivua käydään läpi ja kultakin tällaiselta web-sivulta otetaan lisäpainoa siten, että lisäpaino otetaan jokaisen termipiirteen osalta, mutta tässäkin painoarvoa normeerataan normeerausperusteen ollessa nyt se, että kohdesivuun nähden vektoriavaruudessa etäiseltä sivulta tulevan termin painoa heikennetään.

Toinen jalostusmenetelmä:

Toinen jalostusmenetelmä nojaa ryvästykseen ja niin sanottuun keskusvektoriin. Ryvästyksessä samankaltaiset dokumentit voidaan koota rypääksi. Rypäitä käytetään apuna kyselyä käsiteltäessä. Kullekin dokumenttirypäälle lasketaan keskusalkio, kukin keskusalkio edustaa tiivistetysti rypästä. Keskusalkioiden samankaltaisuutta voidaan vuorostaan vertailla keskenään. Sen perusteella tehdään korkeamman tason rypäitä, joilla on taas omat keskusalkionsa, eli muodostuu ryväshierarkia. Ryväshierarkiassa ylimmällä tasolla on yksi tai muutama ryväs, mutta dokumenttien termivektorit sijaitsevat alimman tason rypäissä. Kun rypään dokumentit myös talletetaan ryppäänä,, haku on nopeaa. Kyselyvektoria verrataan ensin ylimmän tason rypäiden keskusalkioihin. Kaikki ne rypäät, joiden keskusalkioiden samanlaisuusarvo ylittää kynnysarvon, otetaan tutkittavaksi. Sitten verrataan edelleen näiden rypäiden alirypäiden keskusalkioita kyselyvektoriin. Näin jatketaan, kunnes on päästy alimman tason rypäisiin. Näiden rypäiden dokumentit palautetaan hakutuloksena [Aho04]. Rypäitä (clusters), joita on generoitu eri tasoilla (level) näiden web-sivuista, ja

keskusalkiovektoreiden (centroid vectors) muodostamista sekä vielä lopullinen, jalostettu dokumenttivektori (refined feature vector) $w'^{p_{tgt}}_{t_k}$ on esitetty kuvassa 5.



Kuva 5. Jalostetun dokumenttivektorin laskeminen keskusalkiomenetelmällä [Sug03].

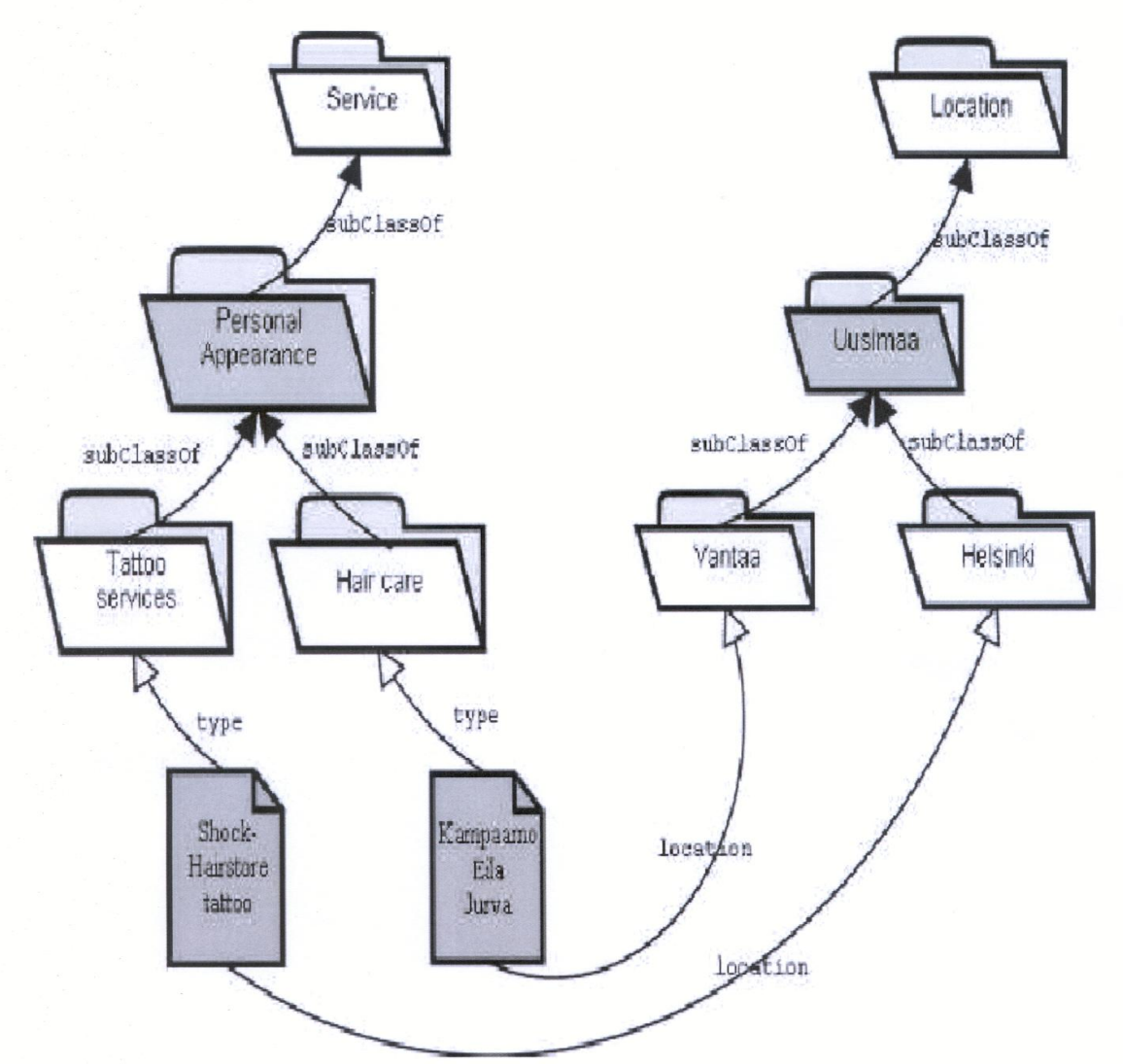
Kolmas jalostusmenetelmä:

Kolmas jalostusmenetelmä muistuttaa toista menetelmää, mutta siinä annetaan painoa myös aiheryhmille. Koeajoissa todettiin, että web-sivut, jotka linkillä osoittavat kohdesivuun p_{tgt} koostuvat yleensä vain kolmesta aihepiiristä. Kovin paljon lisäparannusta saaliiseen kolmas menetelmä ei enää tuonut verrattuna ensimmäiseen ja toiseen menetelmään. Yhteisenä piirteenä kaikille menetelmille koeajoissa ilmeni, että kannattaa panna enemmän painoa in-suunnan sivuille kuin out-suunnan sivuille [Sug03].

5 Hypertekstin jalostamisesta kaupallis-hallinnollista tiedon tarjoamista varten

Hakutermien valinnan tukena voidaan käyttää valmiita sanastoja. Selittävät sanastot eli tesaurukset eli käsitemallit ovat vielä parempi tuki. Esimerkiksi hakutermi, joka on liian harvinainen, voidaan korvata yleisemmällä, tesauruksesta löytyvällä termillä. Tesaurus

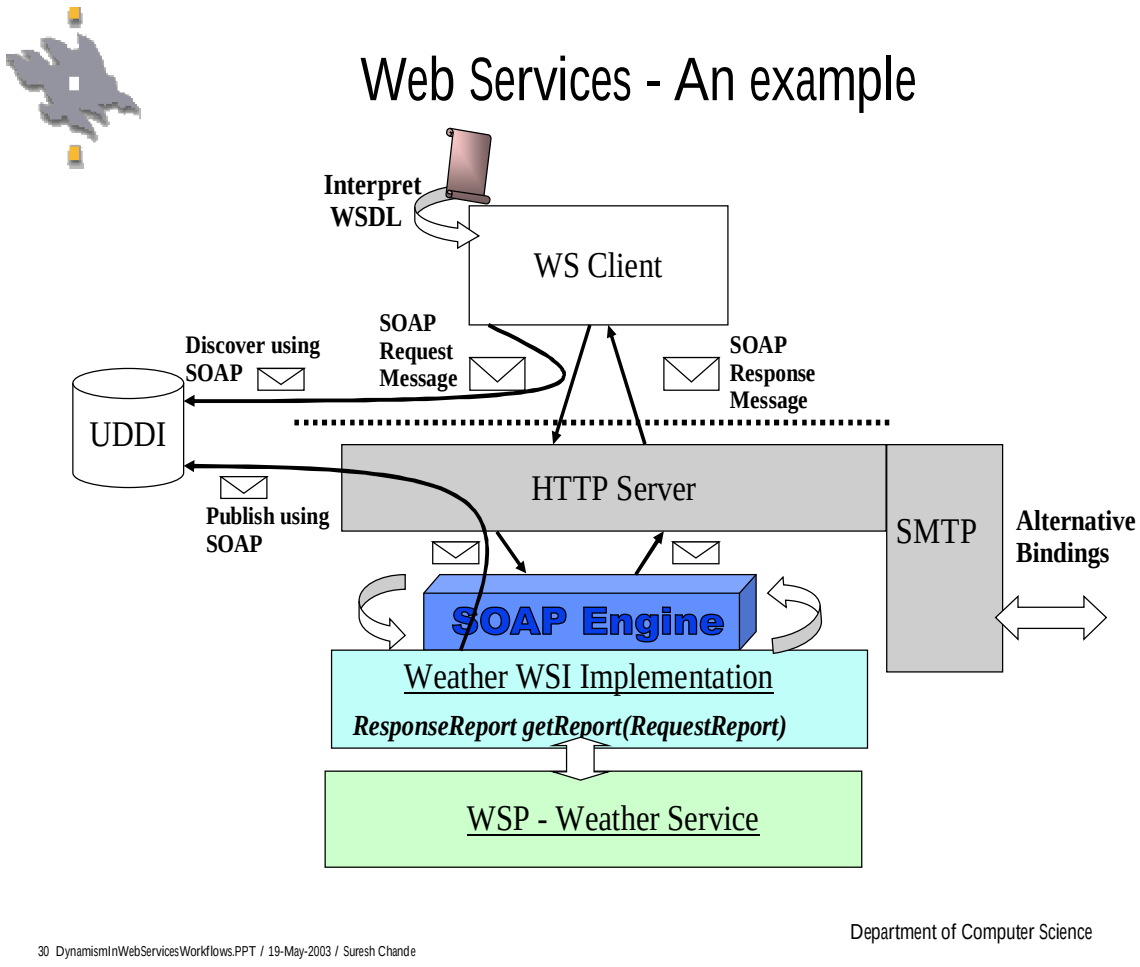
ryhmittelee kapea-alaisia termejä luokaksi. Luokan jäsenten esiintymät dokumenteissa voidaan korvata luokan tunnuksella tai yhdellä jäsenellä [JäK04]. Tesauruksen kehittynyt muoto on ontologia. Esimerkiksi voi olla olemassa lohikeittojen sanasto ja ravintoloiden sanasto. Ravintoladokumentin ja lohikeittodokumenttien välille on voitu asettaa linkki. Sitten kun kerrotaan vielä, että lohikeittoa saa ravintolassa, on tullut lisätyksi syvällinen tieto. Ontologia on käsitemalli, jossa on syvällistä tietoa. Syvällisen tiedon, merkityksen liittäminen verkkoon tarkoittaa, että sen asioille tai olioille on annettu merkitys. Eli on luotu semantiikka. Tällaista WWW:tä tai sen osaa kutsutaan semanttiseksi web´iksi. Termistö on lainattu tietojenkäsittelytieteen puolelle filosofiasta: sana *semantiikka* tai sen alkuperäinen muoto *sem(asi)ologia* tulee kreikan sanoista *semasia* ja *logos*, jotka tarkoittavat merkintää ja oppia jostakin. Myös sana *ontologia* on kreikkalaista juurta, siinä jälkiosa tarkoittaa oppia kuten edellä ja alkuosa *on* tarkoittaa olevaa [Siv60]. Kuvassa 6 on semanttiseen web´iin liittyvä esimerkki mallintamisesta ontologioiden avulla. Siinä kaksi ilmoitusoliota on yhdistetty palvelun (service) ja paikan (location) ontologioihin.



Kuva 6. Ontologiaesimerkki palvelusta ja sijainnista [HVH02].

Laajaa kysyntää sanastolle saati ontologialle ja semanttiselle web´ille tullee vasta, kun joku voi siitä kaupallisesti hyötyä. Eräs kaupallinen versio ovat UDDI-rekisterit. Olennaista UDDI-rekistereissä on, että ne mahdollistavat paitsi automaattisen tiedonhaun, niin myös automaattisen palvelun toteutuksen. UDDI on lyhennys sanoista Universal Description, Discovery & Integration. Se on eräänlainen ohjelmallinen keltaisten sivujen toteutus. Esimerkki säätiedotkin hyödyntävästä, vaikkapa lomamatkapaketin ostosta on kuvassa 7. Palvelun kolme ydinvaihetta ovat: 1) Palveluiden tarjoajat ovat lähettäneet (publish) UDDI´in palvelunsa kuvauksen. Kuvaukset on laadittu WSDL-kielellä. 2) Asiakas, itse asiassa asiakasta palveleva koneagentti, hakee palvelunkuvauksen ja oikean verkkopalvelimen tiedot UDDI´ista. Viestit kulkevat HTTP- tai muulla tietoliikennealustalla toimivan SOAP-tiedonvälitysprotokollan päällä, viestit on laadittu XML-kielellä. 3) Valittuaan

verkkopalvelimen, esimerkiksi Weather Service –palvelun, asiakasohjelma noutaa sitä kuvaavan WSDL-dokumentin ja tulkitsee sen. Luonnollisesti tilauksen teon automatisointi vaatii lisäyksiä järjestelmään, samoin kahden palvelun, esimerkiksi hotellihuoneen varauksen ja elokuvalipun varauksen sovittaminen, vaatii lisäyksiä järjestelmään [Jär02].



Kuva 7. Esimerkki ontogioiden ja semanttisen web ´in kaupallisesta hyödyntämisestä [Cha03].

6 Yhteenveto

Esityksessä kuvattiin hypertekstin asema hypermedian kentässä. Asema havaittiin markkinallisesti vähäiseksi, mutta tietojenkäsittelytieteen näkökulmasta suureksi ja olennaiseksi. Esitettiin tiedonhaun pseudokaavio. Siinä viimeisenä askeleena tehdään kyselynkuvaajan ja dokumentinkuvaajan täsmäytys. Tiedonhaun kaksi päälinjaa,

dokumenttien sisältöön perustuva haku ja dokumenttien linkitystopologiaan perustuva haku, esiteltiin toisaalta yleisesti ja toisaalta algoritmisesta näkökulmasta. Sen jälkeen esitettiin täsmäytyksen hallinta tarkemmin, tietomallin avulla. Sitten tuotiin esiin, että täsmäytyksen ydin on matemaattista laskentaa. Valotettiin laskentaa Boolen algoritmin, HITS-algoritmin, PageRank-algoritmin ja dokumenttivektoriavaruuden tapauksissa. Dokumenttivektroin ja kyselyvektorin kohtaannon laskeminen tuotiin esille keskeisen, erottelukykyä auttavan TF-IDF-kaavion esittelyn yhteydessä. Lopuksi esitelmöijä omana näkemyksenään veti hypertekstistä tesaurusten kautta välittömän jatkumon ontologioihin ja semanttiseen web ´iin. Ontologioiden ja semanttisen web´in lyhyessä esittelyssä palattiin sitten nojaamaan tieteelliseen lähdeaineistoon. Viimeisenä esiteltiin runko UDDI-rekistereitä hyödyntävästä kaupallisesta, täysautomaattisesta kokonaispakettien kuten vaikkapa lomamatkan varaamispalvelusta.

Lähteet

Aho04 Ahonen-Myka, H., Tiedonhakumenetelmät. Tietojenkäsittelytieteen laitos, Helsingin yliopisto, 2004, http://www.cs.helsinki.fi/u/hahonen/thm04/. [25.10.2004]

BrP98 Brin, S. ja Page, L., The anatomy of a large-scale hypertextual web search engine. *Proc.* WW7 Conference, Brisbane, Australia, 1998, http://www.site.uottawa.ca/~stan/csi5389/readings/google.pdf. [29.9.2004]

Cha03 Chande, S., Dynamism in web services. Työnkulun mallintamisen seminaariin liittyvä julkaisematon esitelmädiasto. Tietojenkäsittelytieteen laitos, Helsingin yliopisto, 2003.

Erk01 Erkiö, H., Tiedonhakumenetelmät. Tietojenkäsittelytieteen laitos, Helsingin yliopisto, 2001, http://www.cs.helsinki.fi/u/erkio/thm/k01/. [29.9.2004]

Erk04 Erkiö, H., Hypermediajärjestelmät. Tietojenkäsittelytieteen laitos, Helsingin yliopisto, 2004, http://www.cs.helsinki.fi/u/erkio/hmsem04/. [25.10.2004]

Huh04 Huhtala, Hypermedian perusteet. Matematiikan laitos, Tampereen teknillinen yliopisto, 2004, http://matriisi.ee.tut.fi/hmopetus/hypmed04/pruju/hp04-001-032-4-sivulla.pdf. [27.10.2004]

HVH02 Hyvönen, E., Viljanen, K. ja Hätinen, A., Yellow pages on the semantic web. *Proc.* Towards the semantic web and web services, Hyvönen, E. and Klemettinen, M., toimittajat. The XML Finland 2002 Conference, HIIT Publications 2002–03, Helsinki, 2002. http://www.cs.helsinki.fi/u/eahyvone/. [14.5.2003, 28.10.2004 esitystä ei löytynyt]

JäK02 Järvelin, K. ja Kekäläinen, J., Tiedonhaun menetelmät opintoaineisto. Tampereen yliopiston täydennyskoulutuskeskus ja informaatiotutkimuksen laitos ja Otavan opisto, 2002, http://www.internetix.fi/opinnot/opintojaksot/0viestinta/informaatiotutkimus/po4/. [27.10.2004]

Jär02 Järvinen, J., Hajautetut verkkopalvelut. Docendo Finland Oy, Tummavuoren kirjapaino - Dark, 2002.

Kle99 Kleinberg, J., Hubs, authorities, and communities. ACM Computing Surveys 31, 4 (1999), http://portal.acm.org/citation.cfm?id=345982&coll=portal&dl=ACM&CFID=27941032&CFTOKEN=99744840. [29.9.2004]

LiC99 Li, W. ja Candan, K., Integrating context search with structure analysis for hypermedia retrieval and management. ACM Computing Surveys 31, 4 (1999), http://portal.acm.org/citation.cfm?id=345999&coll=portal&dl=ACM&CFID=27941032&CFTOKEN=99744840. [29.9.2004]

LSZ 02 Li, L., Shang, Y. ja Zhang, W**,** Improvement of HITS-based algorithms on web documents. *Proc.* The eleventh international conference on World Wide Web, Honolulu, Hawaii, USA, 2003, 527 – 535, http://portal.acm.org/citation.cfm?id=511514. [27.10.2004]

Rod99 Rodríguez, H., Extraccion y recuperación de información. Departament d´Anglès i Linguisüística, Universitat de Lleida, 1999, http://www.udl.es/dept/dal/sepln/sepln99. [27.10.2004]

Siv60 Sivistyssanakirja, Hendell-Auterinen, A. ja Jääskeläinen, M., toimittajat. Kustannusosakeyhtiö Otava, Helsinki, 1960.

Sug03 Sugiyama, K. et al., Refinement of TF-IDF schemes for web pages using their hyperlinked neighboring pages. *Proc.* The fourteenth ACM conference on Hypertext and hypermedia, Nottingham, Englanti, 2003, 198-207, http://portal.acm.org/citation.cfm?id=900051.900096&coll=portal&dl=ACM&type=series&idx=900051&part=Proceedings&WantType=Proceedings&title=Conference%20on%20Hypertext%20and%20Hypermedia&CFID=14247157&CFTOKEN=63401439. [29.9.2004]