Spisak pitanja koja su dolazila na usmenom

1. Hits algoritam
2. Cuvanje slika (indexsiranje)
3. Zivotni ciklus dokumenta
4. Kapa mera
5. F mera
6. Ldf mera
7. Page rang
8. Crawler
9. Preciznost (i formule)
10. Povrat (i formule)
11. Meta podaci
12. Seo
13. Google rate algoritmi
14. Razlika i slicnosti izmedju bulovog i vektorskog modela
15. Benchmark testovi (i testove sto se validiraju)
16. Ucestanost terma
17. Pretrazivanje - pokazivanje stranica od jedne ka drugoj (autoriteti iz odredjene oblasti)
18. Pretprocesiranje podataka

Skripta - izvuceno najvaznije iz knjige

Metapodaci

Metapodaci su podaci o dokumentu, odnosno podaci o podacima. Primeri metapodataka za tekstualni digitalni dokument su:

- autor,
- naslov,
- datum nastanka,
- kljucne reci.

Primeri metapodataka za digitalnu fotograju su:

- autor,
- datum i vreme fotograsanja,
- mesto fotograsanja,
- podesavanje aparata,
- objekti prikazani na slici.

Formalno, mogu postojati metapodaci za metapodatke (npr. Ko je uneo metapodatke, da li su menjani metapodaci, itd.).

Metapodaci se koriste kako bi se bolje organizovale kolekcije dokumenata, da se pregled dokumenta prilagodi korisnicima. Takodje, koriste se za klasifikaciju dokumenta i bolju pretragu dokumenata.
Metapodaci mogu nastati tako sto ce sam korisnik koji je kreirao dokument uneti podatke o dokumentu. Drugi nacin je da softver, na kojem je kreiran dokument, sam “generise” metapodatke.

Zivotni ciklus

Zivotni ciklus digitalnog dokumenta:

- Incijalizacija
- Priprema
- Uspostavljanje
- Koriscenje
- Revizija
- Arhiviranje
- Uklanjanje

Inicijalizacija - formiranje podataka potrebnih za kasniju pripremu i utvrdjivanje sadrzaja. U ovoj fazi se radi identifikacija dokumenta. Mogu se dodati i metapodaci koji se odnose na klasifikaciju dokumenta (funkcija dokumenta, prava pristupa, autorska prava, itd).

Priprema - predstavlja proizvodnju sadrzaja sve do trenutka uspostavljanja. Ovde dolazi do stvaranja dokumenta. Meta podaci koji se mogu dodavati jesu nivo razvoja dokumenta, kljucne reci, rezime, izvor dokumenta, itd.

Uspostavljanje - trazenje odobrenja od odredjenih ljudi/institucije ukoliko je ono potrebno za izdavanje dokumenta. Metapodaci koji se mogu dodati su ID podnosioca, rok za dobijanje odobrenja, komentari vezani za proveru i odobravanje, itd.

Koriscenje - dostupan dokument za koriscenje.Metapodaci mogu biti komentari, iskustva korisnika, itd. Distribucija je glavno pitanje, odnosno na koji nacin ce se dokument dostaviti korisnicima.

Revizije - promena sadrzaja ili promena namene dokumenta. Dokument treba da bude sacuvan kao nova verzija kada se menjaju informacije, a ne kada se menja prezentacija. Potrebno je obezbediti podrsku za upravljanjem verzija. Za svaku verziju postoji period formiranja i period vazenja (sekvencijalno ili konkurentno).

Aktiviranje - premestanje dokumenta u nepromenljivu formu.

Postojalo je nekad i uklanjanje kao faza, ali se u modernom dobu radi logicko brisanje, jer memorija vise nije problem.

Standardi u sistemima digitalnih dokumenata Formati metapodataka

Standardi u sistemima digitalnih dokumenata:

- Za uredjenje sistema za upravljanje dokumentima - ISO IEC 82045
- Formati za reprezentaciju metapodataka - MARC 21, Dublic Core, ETD-MS
- Protokoli
    - Za razmenu metapodataka - OAI-PMH
    - Za uadaljeno pretrazivanje - Z39.50 protokol, SRU protokol

Pronalazenje informacija

Data retrieval - bavi se pronalazenjem podataka koji zadovoljavaju precizno definisane kriterijume pri cemu se ocekuje od korisnika da poznaje strukturu podataka u bazi koju pretrazuje. Svodi se na pronalazenje kljucnih reci koje je korisnik naveo u upitu.

Information retrieval - pronalazenje dokumenata, ali se od korisnika ne ocekuje da poznaje kolekciju koju pretrazuje. Podrazumeva se nepreciznost upita, kao i nepreciznost u listi pronadjenih rezultata. Mogu se pojaviti nerelevantni dokumenti. Neophodno je obezbediti rangiranje rezultata zbog velikih rezultata.

Bulov model pretrazivanja je zasnovan na teoriji skupova i Bulovoj algebri. Trazeni pojam se ili nalazi ili ne nalazi u dokumentu. Ne postoji mogucnost rangiranja. Rezultat je binarna vrednost (0 - ne postoji i 1 - postoji).

Vektorski model pretrazivanja uvodi da je stepen slaganja upita i dokumenta vrednost koja je >=0, ali nije celobrojna. Moze se vrsiti rangiranje rezultata. Zasnovano na vektorima u n-dimenzionalnom prostoru.

Probablisticki model pretrazivanja zasnovan na teoriji verovatnoce. Navodi se inicijalni idealni skup i kroz korake se taj skup pomera ka konacnom skupu rezultata pretrage. Problem je kako odrediti inicijalni idealni skup.

Pretraga tekstualnih dokumenata

Tekstualni digitalni dokument se sastoji od reci.
Instanca reci koja se pojavljuje u tekstu je token.
Term - klasa ekvivalencije reci. Odnosno, term je normalizovana rec.

Pretprocesiranje teksta

1. Pretprocesiranje - kao ulaz se dovodi tekst, a na izlazu se dobija lista termova.

Prvo se radi tokenizacija teksta, sto nekad predstavlja veliki problem (brojevi telefona, datumi, viserecne fraze).

2. Normalizacija - svesti termove u indeksiranom tekstu i u upitima na isti oblik (USA i U.S.A). Definisu se klase ekvivalencije termova. Alternativa je asimetricno prosirivanje.

Dijaktrik - glif koji je dodat na neko osnovno slovo (akcenti npr.).

Soundex algoritam - citaj kako je napisano - nije bitno kako je napisano ako citajuci daju istu zvucnost.

Lematizacija - Redukovanje reci na baznu formu (bih, bio, bismo -> biti).

Steming - odsecanje krajeva reci sa ciljem da se postigne rezultat sto slicniji onome koji postize pravilna lematizacija bazirana na lingvistickom znanju.
Moze biti zasnovan na algoritmu ili na recniku.
Steming zasnovan na algoritmu obicno skida sufikse (a-sufiks, i-sufiks, d-sufiks).

Potrebno obratiti paznju na greske koje se mogu desiti:
- Under-stemming - skidanje previse malog sufiksa
- Over-stemming - skidanje previse velikog sufiksa
- Mis-stemming - skidanje necega sto je deo stema

Snowball - pogodan za opis steminga zasnovanog na algoritmu.

Stemeri zasnovani na recniku - tezi za kreiranje ali mogu davati bolje rezultate. Najpoznatiji algoritam ovog tipa je Porterov algoritam.

Bulov model pretrazivanja

Zasnovan na teoriji skupova i bulovoj algebri. Trazene informacije se izrazavaju upitom, koji su logicki izrazi. Svaki dokument se posmatra kao skup termova. Pojam se ili nalazi ili ne nalazi u dokumentu. Nema mogucnost rangiranja. Primer jednog pretrazivaca zasnovanog na bulovom modelu je Westlaw. Koriste ga profesionalni korisnici koji vole preciznosti.

Invetrovani indeks

Bolje resenje od Bulovog modela jeste kreiranje invertovanog indeksa koji predstavlja strukturu podataka koja mapira reci i brojeve iz sadrzaja dokumenta na dokumente u kojima se javljaju, a u boljem slucaju i na lokaciju na kojoj se te reci i brojevi javljaju u dokumentima.

Matrica incidencije term/dokument predstavlja matricu u kojoj se u zaglavlju kolona nalaze dokumenti, a u vrstama term-ovi, dok vrednost u matrici je 1 ako se term pojavljuje u dokumentu, a u suprotnom 0.
Vektor incidencije je jedna vrsta u ovoj matrici, tj. Vektor sa elementima 0 i 1.

Invertovani indeks - za svaki term t, cuvamo listu dokumenata koji sadrze term t.

Algoritam za konstrukciju invertovanog indeksa:

1. Prikupljanje dokumenata koje treba indeksirati
2. Tokenizacija teksta - pretvaranje svakog dokumenta u listu tokena
3. Pretprocesiranje teksta - formiranje liste normalizovanih tokena tj. Termova, koji ce biti u recniku
4. Indeksiranje dokumenata - formiranje invertovanog indeksa koji ima recnik i pojave

Procesiranje upita

Upit dva tokena: *Lucene AND multimedijalnih*

Algoritam:

1. Pretprocesiranje upita: *lucene AND multimedijalan*
2. Pronalazenje terma *lucene* u recniku termova
3. Ucitavanje liste pojava ovog terma iz fajla sa pojavama
4. Pronalazenje terma *multimedijalan* u recniku termova
5. Ucitavanje liste pojava ovog terma iz fajla sa pojavama
6. Izracunavanje **preseka** ove dve liste pojava
7. Vracanje rezultata korisniku - vracanje dokumenata koji se nalaze u prethodno izracunatom preseku

Ukoliko imamo upit sa tri tokena: *Lucene AND multimedijalnih AND tekstura* - potrebno je obratiti paznju na redosled izvrsavanja Bulovih operacija - sortirati liste po duzini i tek onda izvrsiti operacije.

Pointeri za preskakanje

Posto je procesiranje upita linearno zavisno od duzine liste, pokusavamo da zaobidjemo tu zavisnost.
Pointeri za preskakanje omogucavaju preskakanje pojava koje svakako nece biti u rezultatu.
Vise skokova - svaki pointer preskace malo elemenata, ali ga mozemo cesce koristiti;
Manje skokova - svaki pointer preskace puno elemenata, ali ga mozemo retko koristiti.

Upiti fraze

Upiti cesto mogu da sadrze celu frazu, koja se sastoji od dve ili vise reci.
Ako postoji upit “sive pantalone” onda recenica “Voli da nosi sive kosulje i plave pantalone” nije pogodak.

Pristupi:

1. Dvorecni indeks - pored termova indeksiraju i svaki susedni par reci u tekstu kao frazu: “sive markirane pantalone” -> “sive markiranje” i “markirane pantalone”. Upit se moze predstaviti kao sledeci *“sive markirane” AND “markirane pantalone”.* Lako je moguce da rezultati budu false positive, tj. Da se cela fraza ipak ne nalazi u dokumentu.
2. Pozicioni indeks - dobra alternativa za dvorecne indekse. Omogucavaju odgovore na upite fraze proizvoljne duzine. Za jedan term u nepozicionom indeksu je vezana lista pojava ovog terma u dokumentima, pri cemu je svaka pojava docID - identifikator dokumenta u kolekciji koji sadrzi term. Kod pozicionog indeksa svaka pojava je docID i lista pozicija. Pozicioni indeks se moze koristiti za blizinsku pretragu - implementacija preko dekartovog proizvoda.

Vekstorski model pretrazivanja

Omogucuje parcijalno poklapanje upita i dokumenata i samim tim omogucuje rangiranje i sortiranje. Upit i dokument se predstavljaju kao n-dimenzionalni vektor gde je n broj termova u recniku.

Ocena relevantnosti

Ocena je mera koliko se dokument i upit poklapaju.
Jaccard-ov koeficijent je uobicajena mera preklapanja dva skupa definisana na sledeci nacin:

1. Neka su A i B skupovi (bar jedan je neprazan),
2. Jaccard-ov koecijent:
   $$Jaccard(A, B) = |A \cap B| / |A \cup B|$$
3. Jaccard(A, A) = 1,

4. Jaccard(A, B) = 0 ako je A ∩ B = 0.

Mane su:

- Ne uzima u obzir frekvenciju terma
- Retki termovi su informativniji od cestih - Jaccard ovo ne uzima u obzir

Frekvencija terma tf

Koristimo matricu koja sadrzi informaciju o broju ponavljanja terma u dokumentu - brojacku matricu.
Svaki dokument je prikazan pomocu vektora broja pojavljivanja.
Za razmatranje vektorskog modela korisitcemo model “vrece sa recima” koji ne uzimamo u obzir redosled reci u dokumentu.

Frekvencija terma tf terma t u dokumentu d definise se kao broj pojavljivanja t u d. Zelimo da koristimo tf kada racunamo upit/dokument ocene.

Koristimo logaritamsku tezinu frekvencije terma t u dokumentu d koja se definise kao:

wt,d = { 1 + log10tf ako je tft,d > 0
0 inace

Pomocu log10 smanjujemo relevantnost, posto relevantnost ne raste proporcionalno sa frekvencijom terma.

Frekvencija dokumenta df

Retki termovi su informativniji od cestih. Zelimo da postignemo za retke termove veliku tezinu, dok za ceste termove zelimo takodje da postignemo pozitivne tezine ali manje tezine od onih retkih.

dft je oznaka za frekvenciju dokumenta i predstavlja broj dokumenata u kojima se pojavljuje term **t**.
df je inverzna mera informativnosti terma, zbog cega definisemo **idf** tezinu terma **t** kao:

idft = log10 (N / dft)

idf je mera srazmerna informativnosti terma.

Frekvencija kolekcije cf

Frekvencija kolekcije terma t je broj pojavjivanja t u klekciji i oznacava se sa cf. Ova mera se retko koristi za utvrdjivanje informativnosti terma, mnogo se cesce koristi frekvencija dokumenta.

Tf-idf

Jedna od najpoznatijih tezina u oblasti pronalazenja informacija je ft-idf tezina.
Predstavlja proizvod njegove tf tezine i njegove idf tezine:

$$wt,d = (1 + \log tf) \cdot \log N/df$$

Ova tezina raste sa brojem ponavljanja terma u dokumentu i sa retkoscu terma u kolekciji.

Tezinska matrica

Binarna matrica -> brojacka matrica -> tezinska matrica (tf-idf tezine)

Algoritam za pretragu putem vektorskog modela:

1. Predstavljanje svakog dokumenta u formi normalizovanog tf-idf vektora
2. Predstavljanje upita u formi normalizovanog tf-idf vektora
3. Racunanje kosinusnih slicnosti izmedju upita i svakog dokumenta
4. Rangiranje dokumenata prema slicnosti
5. Prikazivanje najboljih K (npr. K = 10) dokumenata korisniku

Pretraga struktuiranih tekstualnih dokumenata

U pitanju je pretraga sadrzaja koji ima strukturu, ali su elementi u strukturi bogati tekstualnim sadrzajima, pa je potrebno omoguciti normalizaciju upita i tekstova, relevantnost dokumenta za odredjeni upit, sortiranje, itd.

Pretraga struktuiranih sadrzaja se deli na:
- Pretraga po parametrima i zoanama
- Pretraga slozenijih struktura - najcesce XML

Pretraga po parametrima i zoanama

Omogucuje pretragu po:
- Parametrima: datumu izmene, pripadnosti nekoj grupi, redni broj, itd.
- Zoanama: naslov, apstrakt, uvod, kljucne reci, itd.

Sadrzaj koji se naalzi u parametrima se obicne ne pretprocesira ali se moze pretrazivati po njima - filtriranje.
Zone mogu biti svi metapodaci nekog dokumenta.

Pretraga tekstualnih sadrzaja slozenijih struktura

XML je pomogdna struktura za opis struktuiranog ili “polustruktuiranog” sadrzaja.

Za opis seme postoje dva standardizovana nacina:

- DTD Scheme
- XML Scheme

XPath - standar za izraze kojima se vrsi selekcija elemenata.
Document Object Model - DOM - predstavlja objektnu reprezentaciju XML-a koja se cesto koristi u programskoj obradi XML-a.

Data-centric XML predstavlja XML koj ise koristi kao sredstvo za komunikaciju i najcesce sadrzi podatke iz relacione baze.

Document-centric XML - XML ciji je sadrzaj bogat tekstom - polustruktuiran.

Pretraga veba

Veb crawling

Crawler (Spider ili Web robot) je program za automatsko kretanje kroz graf veba i preuzimanje veb stranica radi neke dalje obrade. Obicno se ta dalja obrada odnosi na indeksiranje zarad pretrazivanja.
Crawler pocinje sa poznatim seed URL-ovima. Preuzima ih i parsira. Nakon toga ekstrahuje URL-ove iz ovih fajlova koje postavlja u listu. Za svaki URL iz liste ponavlja prethodno nabrojane zadatke.

Nemoguce je raditi crawling sa jednim racunarom - sve operacije su distribuirane na vise racunara.

Crawler mora biti uctiv, postovati implicitne i eksplicitne zahteve veb sajtova.

Eksplicitni zahtevi veb sajtova su izrazeni upotrebom robots.txt fajla i u njemu se navodi sta crawler sme, a sta ne sme da preuzme.

Implicitno uctivost podrazumeva cak iako nema specifikacije da crawler mora izbegavati preuzimanje puno strana sa jednog veb sajta u kratkom vremensom roku.

Crawler mora biti robustan, mora reagovati na nenormalne situacije, maliciozne strane, klopke za crawler-e.
Distribuiranost - crawler mora biti dizajniran da se izvrsava na vise racunara.
Skalabilnost - mogucnost da je lako dodati jos racunara u sistem.
Perfomanse (efikasnost) - moraju biti maksimalne.
Prioritetnije preuzimanje - pozeljna karakteristika dobrog crawler-a. Ponovo preuzimanje novog sadrzaja sa veb strana koje su u medjuvremenu izmenjene.
Prosirivost novim formatima i protokolima - pozeljna osobina crawler-a.

URL frontier - sadrzi URL-ove koje je potrebno preuzimati.
Fetch komponenta - preuzima naredni URL iz URL frontier-a (upotrebom DNS komponente - vraca IP adresu).

Parse komponenta - radi parsiranje sadrzaaja koji je dobavljen kako bi se ustanovilo da li je sadrzaj duplikat.
Content seen komponenta - utvrdjuje da li je sadrzaj duplikat ili priblizno duplikat.
URL filter - proverava za svaki estrahovani URL da li prolazi sve filtere - filteri crawler-a i veb sajta definisani su pomoci robots.txt.
Dupl URL elim - komponenta koja proveravas za sve URL-ove koji su proslii filtere, da li oni vec postoje u URL fontier-u, da li ih treba dodati ili ne.
Host splitter - za sve URL-ove koji su prosli filter utvrdjuje koji od racunara u distribuiranom sistemu je zaduzen za taj URL i da pozove njegovu Dupl URL elim komponentu.

Analiza linkova

Dokumenti su povezani izmedju linkova.
Graf - cvorovi su dokumenti, a relacije su linkovi izmedju njih.

Cvorovi:

- Dobri - nemaju linkove ka losim - ako pokazuju na neki cvor i taj cvor je dobar
- Losi - Ako cvor ima linkove ka losem cvoru i on je los
- Ne zna se da li su dobri ili losi

Tekst linka - moze se iskoristiti za kvalitetniji opis stranice na koju link upucuje.
Konekcioni serveri skladiste i omogucuju pretragu inlinks i outlinks URL-ova i mogu da odgovore na dve vrste upita vezanih za graf veba:

1. Ko upucje na dokument ciji je URL u upitu?
2. Na koga upucuje dokument cije je URL u upitu?

Prilikom neke pretrage, web browser se seta kroz graf vega. Krece se od slucajno izabrane veb strane i ide ka ostalim grafovima. Zelimo da svaka strana ima tezinu za poseti - page score. Ukoliko se naidje na mrtav cvor, teleportujemo se u proizvoljni cvor pri cemu je tezina za prelaz jednaka 1 / UkupanBrojCvorova. Prelazak na novu veb stranicu cini teleportovanje iz bilo kog cvora koji nije mrtav u neki drugi cvor, gde je tezina za prelaz
Alpha / UkupanBrojCvorova, ili u neki drugi cvor prema kojem postoji link i tezina iznosi (1-Alpha) / BrojIzlaznihLinkova, gde je Alpha parametar u intervalu 0-1 koji predstavlja verovatnocu da korisnik unese veb adresu direktno u polje za adresu.

Pagerank stranice se moze iskoristiti za rangiranje rezultata. To je mera kvaliteta stranice.
Pagerank algoritam:

- Kreiramo matricu P upotrebom izlaznih linkova sa veb strana i unapred definisanog parametra Alpha
- Izracunamo vektor Alpha koji predstavlja (levi) eigenvector matrice P
- Ai je pagerank stranice i

Pagerank ne zavisi od upita, on sluzi samo da se izvrsi rangiranje rezultata, a tf-idf bi trebao da ragnira rezultate shodno relevantnosti u odnosu na korisnikovu potrebu predstavljenu upitom.

Za rangiranje rezultata koristi se i HITS (Hyperlink-Induced Topic Search).

Potrebno je uociti dva skupa cvorova:

1. Hub - stranica (cvor) koja ima linkove ka puno drugih stranica
2. Authority - stranica ka kojoj linkuje puno Hub-ova.

Dobri cvorovi odredjene teme imaju puno linkova ka autoritativnim stranama. Dobre autoritativne strane se spominju na puno cvorova. Potrebno je iterativno utvrdjivanje autoratitivnih strana i cvorova. Potrebno je odabrati pocetni skup cvorova i autoriteta. I iterativno menjati skup.

Korenski skup (Root set) je sve sto je odgovor na upit.
Pocetni skup je Root set + in-links + out-links.

Inicijalna vrednost hub score prve iteracije za svako x je h(x) = 1, a authority score je a(x) = 1. Nakon jedne iteracije hub score i authority skor su sledeci.

$$h(x) = \sum a(y)$$
$$a(x) = \sum h(y)$$

Nakon jedne iteracije uzimamo samo najboljih n cvorova i autoriteta na osnovu prethodno izracunatih vrednosti i pocnemo narednu iteraciju, ali ne vracamo vrednost na 1. U praksi 5 iteracija dovoljno je da cvorovi i autoriteti konvergiraju.
HITS je zavistan od upita. Iteracije ne zavisi od upita, samo osnovni skup.

Search engine optimization - SEO

Da bi izbegli placanje mesta na vebu za Ads (reklame), alternativa je koriscenje Search Engine Optimization-a (SEO).

SEO - tehnika da se izmeni veb strana tako da ona bude dobro rangirana u algoritamskoj pretrazi veb pretrazivaca za odgovorajuce kljucne reci u upitu.
Potrebno je platiti SEO strucnjaka.
Koristili su se trikovi poput teksta u boji pozadine u HTML stranici, zatim u CSS navodjenjem taga koji se nije koristio nigde, itd.

Cloaking - popularna nelegitimna SEO radnja koja predstavlja poturanje specijalno pripremljenih sadrzaja crawler-ima.

Doorway pages - nelegitimna SEO tehnika koja predstavlja kreiranje stranica optimizovanih za jednu kljucnu rec od interesa i te stranice samo redirektuju do prave stranice.

Nove nelegitimne SEO tehnike - skriveni linkovi, Domain flooding - gomila domena koji imaju link ili cak redirektuju na ciljnu stranicu.

Pretraga multimedijalnih dokumenata

Prema zavisnosti od vremena:

- Vremenski zavisni mediji - menjaju se tokom vremena: video, zvuk, animacije.
- Staticni mediji - ne menjaju se kroz vreme: slika, tekst.

Prema nacinu konzumiranja:

- Linearni mediji - uvek se konzumiranje radi ustaljenih redosledom
- Nelinearni mediji - korisnik bira put kojim konzumira medij - hitpertekst sa linkovima

Upit nad kolekcijom multimedijalnih sadrzaja:

1. Upit se izrazava recima za sta nam je potreban specifican upitni jezik i oslanjamo se da mediji imaju kvalitetne metapodatke
2. Upiti se izrazavaju uzorkom, trazimo medije slicne datom uzorku

Pretraga moze biti:

1. Text based - pretraga multimedijalnih sadrzaja bazirana na tekstu kojim je opisana - zasniva se na metapodacima koje je potrebno kreirati - opis visokog nivoa
2. Content based - pretraga multimedijalnih sadrzaja bazirana na sadrzaju - koristi se racunar - opisivanje sadrzaja se realizuje ekstrahovanjem osobna medija(slika, zvuk, video). Ove osobine se izracunavaju na osnovu prostog sadrzaaj medija - piksela slika npr.

Slika

Upiti se mogu izraziti na dva nacina:

1. Upit izraziti recima - oslanjamo se na metapodatke
2. Upit uzorkom

Osobine niskog nivoa se mogu racunati za celu sliku - globalne osobine, ili po delovima slike - lokalne osobine.

Ekstrahovanje osobina je racunarski jednostavnije, ali ne odslikava covekovo poimanje slike. Predstavljaju prosek za celu sliku.
Za lokalno ekstrahovanje potrebno je uradiit segemntaciju slike i za tako dobijene delove izvrsimo ekstrahovanje osobina.
Dva nacina segmentacije:

1. Tile-based - podela slike na pravilan raspored delova.
2. Region-based - podela slike po regionima na osnovu sadrzaja, podela slike na vizuelno koherentne zone. Slozen i nepouzdan proces.

Potrebno je svakako ektrahovanje niskog nivoa:

Boja - moze se izracunati signatura regiona/slike.
Jedan od nacina da koristimo boju kao osobinu jete da kreiramo histogram boja.
Drugi nacin je da upotreba vektora koherencije boja.

Oblik - pomocu aktivne konture pronalazimo objekte na slici.
Gradient vector flow - usavrsene aktivne konture.

Furijeovi deskriptori - za prepoznavanje oblika. Otporna na geometrijske transformacije i sum.

Tekstura - matematicki opis ponavljajuceg sablona: glatko, peskovito, zrnasto,... Preko matrice ponavljanja.

Zvuk

Proizvodi se konverzijom energije u talase koji se prostiru kroz neki etar.

Vrste zvuka:
- Baziran na govoru
- Baziran na muzici
- Ostalo - alarm, zvuci iz prirode

Pretraga kolekcije zvucnih zapisa je vrlo korisna na kolekcijama zvucnih zapisa baziranih an govoru.
Osobine visokog nivoa se mogu podeliti na:
- Sadrzaj
- Identitet
- Jezik
- Ostalo

Pretrazivanje zvukova baziranih na muzici se zasniva na pretrazi po uzorku.

Performanse pretrazivanja

Faktori zadovoljstva korisnika ukljucuju sledece kriterijume:
- Brzinu dobijanja odgovora
- Velicina indeksa - sta sve mogu pronaci u kolekciji
- Nezatrpan korisnicki interfejs
- Relevantnost dobijenih odgovora - najvazniji kriterijum je da li je korisnik dobio ono sto je trazio
- Besplatan pristup

Relevantnost

Standardna metodologija za merenje relevantnosti rezultata pretraga ima sledece elemente:
- Test-kolekccija dokumenta
- Skup test-upita
- Binarna ocena relevantnosti svakog para upit-dokument

Zadovoljstvo korisnika se moze meriti samo prema releatnosti u odnosu na informacione potrebe, a ne upite.

Marginalna relevantnost dokumenta u rezultatu je dodatna informacija koju sadrzaj dokumenta donosi.

Preciznost - udeo pronadjenih relevantnih dokumenata u svim pronadjenim dokumentima, odnosno u listi rezultata pretrage.

Preciznost = pronadjeni relevantni / svi pronadjeni

Povrat/Odziv - udeo pronadjenih relevantnih dokumenata u svim relevantnim dokumentima koji postoje u kolekciji.

Povrat = pronadjeni relevantni / svi relevantni

	Relevantan	Nerelevantan
Pronađen	true positives (TP)	false positives (FP)
Nije pronađen	false negatives (FN)	true negatives (TN)

$$P = \frac{TP}{TP + FP}$$
$$R = \frac{TP}{TP + FN}$$

Tacnost - deo odluka koje su ispravne, u smislu prethodne tabele tacnosti .

Tacnost = (TP + TN) / (TP + FP + FN + TN)

F mera - omogucava da se meri kompromis izmeu preciznosti i povrata:

$$F = \frac{1}{\alpha\frac{1}{P} + (1-\alpha)\frac{1}{R}} = \frac{(\beta^2+1)PR}{\beta^2 P + R} \quad \text{gde} \quad \beta^2 = \frac{1-\alpha}{\alpha}$$

$$\alpha \in [0,1] \text{ pa prema tome } \beta^2 \in [0,\infty]$$

Kada je β > 1 povrat vrednujemo vise nego preciznost, a kada je β < 1 onda preciznost vrednujemo vise nego povrat.

Evaluacija performansi

Mere performansi:

Clickthrough - nije pouzdana za prvi pogodak, ako se posmatra jedan clickthrough. Ova mera je pouzdana u proseku za veliki broj kosnika i upita.

Laboratorijske studije ponasanja korisnika - posmatranje korisnika i njihovog ponasanja prilikom pregrage. Skupo.

A/B testiranje - ima za clij testiranje jednog unapredjenja sistema za pretrazivanje. Za ovo testiranje potreban je pretrazivac sa velikim brojem korisnika i svakodnenvih uptina.

Ocena relevantnosti su korisne samo ako su konzistentne medju ocenjivacima.
Konzistentnost medju ocenjivacima mozemo meriti kapa merom (k).
Kapa mera - mera koliko su medjusobno ocenjivaci slazu i ova mera je dzajnirana za kategoricne ocene.

P(A) - koji deo od ukupnog broja slucajeva se ocenjivaci slazu
P(E) - koji deo slaganja bismo dobili slucajno

$$K = (P(A) - P(E)) / (1 - P(E))$$

Rezultati prerage

Najcesce opis dokumenta u rezultatu ukljucuje naslov i neke metapodatke. Bilo bi dobro da rezultat sadrzi i sazetak.

Sazetak moze biti:

- Staticki - dokument je uvek isti bez obzira na upit. Obicno je sazetak podskup dokumenta. Uvek se nekim pravilom “generise”.
- Dinamicki - zavisni od upita. Cilj prikazati jedan ili vise fragmenata iz dokumenta koji sadrze termove iz upita.

Klasifikacija

Klasifikacija pokusava da utvrdi kojoj klasi ili klasama posmatrani objekat pripada.
Koristi se kao korak u pretprocesiranju teksta.
Moze se koristiti pomocu masinskog ucenja kao detekcija spam strana.
Klasifikacijom se mogu razdvojiti rezultati po unapred definisanim grupama.
Rangiranje rezultata se moze izvrsiti pomocu klasifikacije.

Vrste klasifikacije:

- Rucna klasifikacija
- Na osnovu pravila
- Na tehnikama masinskog ucenja

Klasterovanje

Klaster analiza je podela skup objekata na podskupove i cilj klastera je nalazenje grupe objekata takvih da su objekti iz grupe medjusobno slicni i da su razliciti od objekata iz drugih grupa.
Za razliku od klasifikacije koja je bazirana na masinskom ucenju, za klasterovanje se koristi nenadgledani metod obucavanja, nema definisanih klasa i nema obucavajuceg skupa kreiranog od strane eksperta.

Klasterovanje rezultata pretrage - grupise rezultate po klasterima, odnosno rezultati nisu prosta lista relevantnih odgoovra.

Scatter-Gather - tehnika razbijanja na klatere i spanjanje klastera koja se koristi u sistemima za pregragu. Cilj je bolji korisnicki interfejs.

Collection clustering - tehnika za klasterovanje kolekcije bez interakcije sa korisnikom i bez postavljenog upita.

Language modeling - koriscenje klasterovanja da bi se resio problem sinonima.

Cluster-based retrieval - da bi se ubrzala pretraga. Racunanje slicnosti vektora koji predstavlja upit i dokumenta u kolekciji moze biti sporo.

Relevance feedback

Relevance feedback ne radi na pricnipu klasterovanja, ali takodje moze unaprediti sistem. Radi na sledeci nacin:
- Korisnik postavlja upit
- Pretrazivac vraca skup dokumenata
- Korisnik oznacava neke dokumente kao relevantne, a neke kao nerelevantne
- Pretrazivac izracunava novu reprezentaciju informacione potrebe, bolju od inicijalnog upita
- Pretrazivac izvrsava novi upit i vraca rezulatte korisniku