Norges teknisk-naturvitenskapelige universitet
Institutt for datateknikk og informasjonsvitenskap

EKSAMENSOPPGAVE I FAG TDT4117 – INFORMASJONSGJENFINNING

Faglig kontakt under eksamen: Heri Ramampiaro

Tlf.: 93440

Eksamensdato: ***1. desember 2010***

Eksamenstid / varighet: *15.00-19.00 / 4 timer*

Tillatte hjelpemiddel: *D: Ingen trykte eller håndskrevne hjelpemiddel tillatt. Bestemt, enkel kalkulator tillatt.*

Språkform: Bokmål /SENSURVEILEDNING

Sensurdato: 22. desember 2010

Det ønskes **korte** og **konsise** svar på hver av oppgavene.
Les oppgaveteksten meget nøye og vurder hva det spørres etter i hver enkelt oppgave.

Begrunn svar på alle oppgaver.

Dersom du mener at opplysninger mangler i oppgaveformuleringene, beskriv de antagelsene du gjør.

Oppgave I (20%)

a) Nevn de tre viktigste karakteristikker som skiller *datagjenfinning* fra *informasjonsgjenfinning*.

Her er det nok at studentene nevner: datagjenfinning (DR) handler ofte om å finne dokumenter (records) som inneholder nøkkelord mens IR handler ofte om å finne dok. som beskriver et tema. DR har veldefinert semantikk mens i IR er semantikken ofte løs. DR tolerer ikke feil mens i IR er småfeil tillatt. Hvis studentene nevner noe om full vs. delvis match, ikke vs. med rangering er det også ok. (5%)

b) Hvorfor er valg av "Index terms" viktig informasjonsgjenfinningssammenheng? Begrunn svaret ditt.

Valg av Index termer er viktig fordi de skal **representere** dokumentene. Valg av termer som skal inngå i indeksen bør derfor være de som man oppnår dette med. Dette er blant annet de termene som bidrar mest i diskriminering mellom dokumenter. Valg indextrermer skal også lette gjenfinning av relevante dokumenter. (4%)

c) Forklar hvilke steg eller operasjoner man *bør* utføre før man kan indeksere dokumenter?

Her skal studentene få med minst tokanization, stemming og/eller lemmatization, stoppordfjerning, casefolding. Det kreves kort forklaring av hver operasjon. (6%)

d) Hva menes med "Feature extraction" og i hvilke sammenhenger brukes dette begrepet? Hvilke tre krav bør uttrekning av features (feature extraction) oppfylle?

Feature extraction handler om å hente ut de egenskaper i et multimedia objekt som skal representere dette objektet ifm. multimedia informasjongjenfinning.

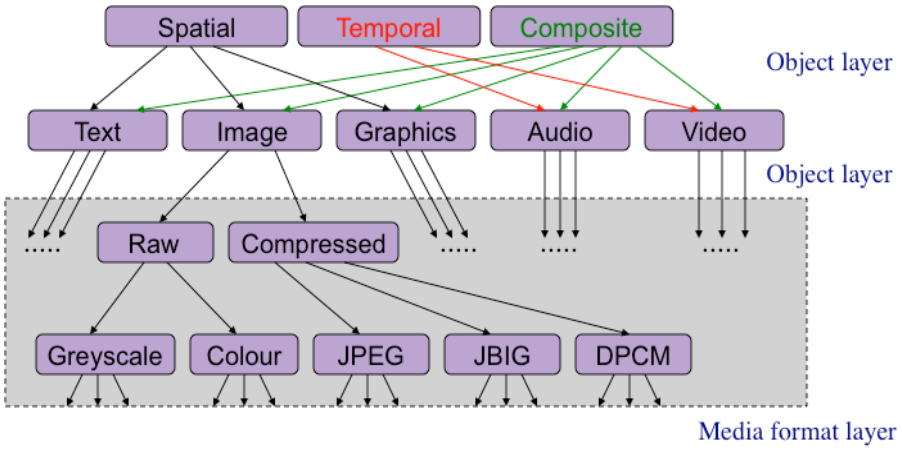
Krav: (1) Kompletthet: det skal dekke informasjonsbehovet, (2) Skal ha kompakt representasjon og lagring og (3) Skal være mest mulig effektiv ifm beregning av likhet mellom forskjellige multimedia objekter.

(5%)

Oppgave II (10%)

a) Tegn opp en figur av multimedia datamodell. Hvilke tre lag (layers) består denne modellen av? Gi minst et eksempel på noe som hører til hvert lag.

Studentene skal tegne dette:

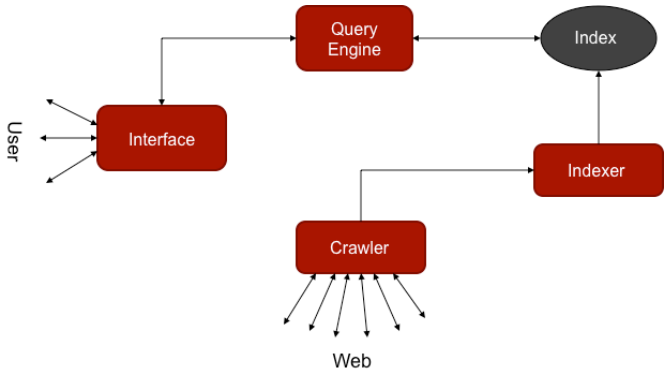


(3%)

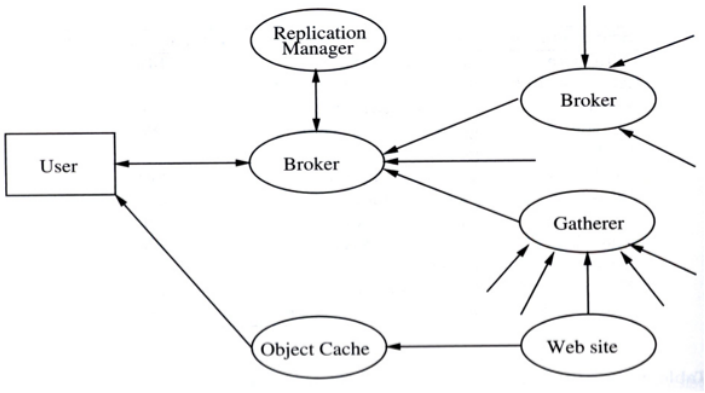
b) Sammenlikn sentralisert og distribuert web søkesystem. Bruk blant annet tegninger av arkitekturene til å støtte forklaringen din.

Det holder at studentene tegner opp forklarer kort de to arkitekturene her:

Sentralisert:



Distribuert:



Hovedforskjellene er håndtering av indekseringen av sidene. Det sentraliserte systemet benytter seg av Crawlere mot et sentralisert indexer (en server) mens det distribuerte systemet bruker flere gatherers som kummuniserer med brokers som kan kommunisere med andre brokers. Disse tar seg da av i felleskap indekseringen.

Det er et pluss hvis man nevner fordelene og ulempene med hvert av systemene også som for eksempel har noe med belastning på server, feilhåndtering etc.

(7%)

Oppgave III (20%)

Tabell 1 viser resultat av dokumentsøk Ola har utført.

Rang	Dok ID	Score	Relevans
1	232	1534	
2	111	1240	REL
3	2343	1111	
4	332	1022	REL
5	441	1001	
6	112	0999	REL
7	22	0555	
8	1	0233	REL
9	23	0220	

10	334	0100	

Tabell 1

Anta at vi i alt har 6 relevante dokumenter i samlingen vår.

a) Forklar begrepene "*precision*" og "*recall*"?
Precision: andelen av gjenfunnet dokumenter som er relevante

Recall: andelen av alle relevante dokumenter som er gjenfunnet

(4%)

b) Regn ut verdiene av precision og recall ut fra resultatet i tabell 1.
Precision= 4/10 = 40%

Recall = 4/6 = 66%

(3%)

c) Hva er R-precision? Regne ut verdien av R-precision for resultatet over.
R-precision er andelen av R-antall gjenfunnet dokumenter som er relevante, hvor R her er det totale antallet relevante dokumenter, i.e., R=6.

R-precision = 3/6 = 50%

(6%)

d) Hva er MAP (Mean Average Precision)? Forklar.

MAP – mean average precision er et mål som blir brukt til å måle evnen til er IR-system til å finne relevante dokumenter basert på et sett av queries.

For å finne verdien av MAP må man beregne gjennomsnittsverdien av Precision for hver relevant og gjenfunnet dokument. For eksemplet vårt i tabell 1 betyr dette at vi beregner precision på når vi har hentet 2, 4,6, og 8 dokumenter. Deretter regnes snittet (AveP) av dette ut.

Dvs. P(2) = 0.5, P(4) = 0.5, P(6)=0.5, og P(8) = 0.5. AveP for denne spørringen er derfor 0.5. Man gjør tilsvarende for alle de andre spøringene og regner en snittverdi for dette og får MAP.

$$\mathrm{MAP} = \frac{\sum_{q=1}^{Q} \mathrm{AveP(q)}}{Q}$$

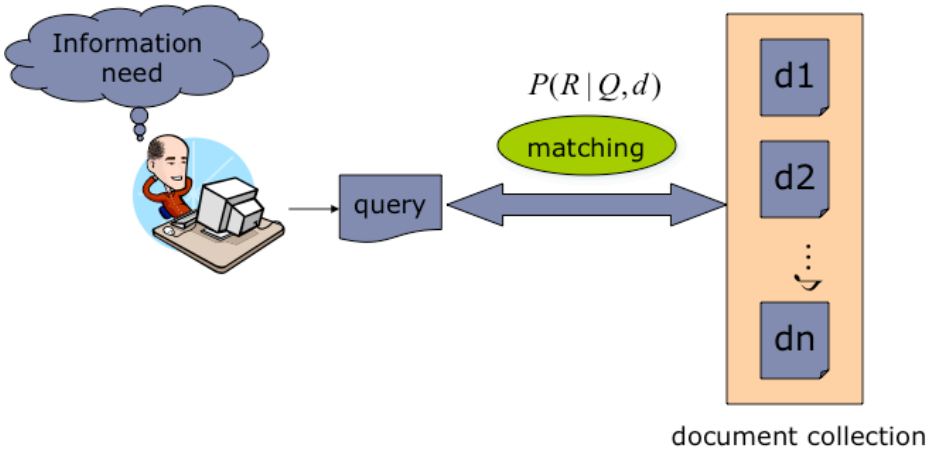
Hvor, Q er antall spørringer man har AveP verdien for.

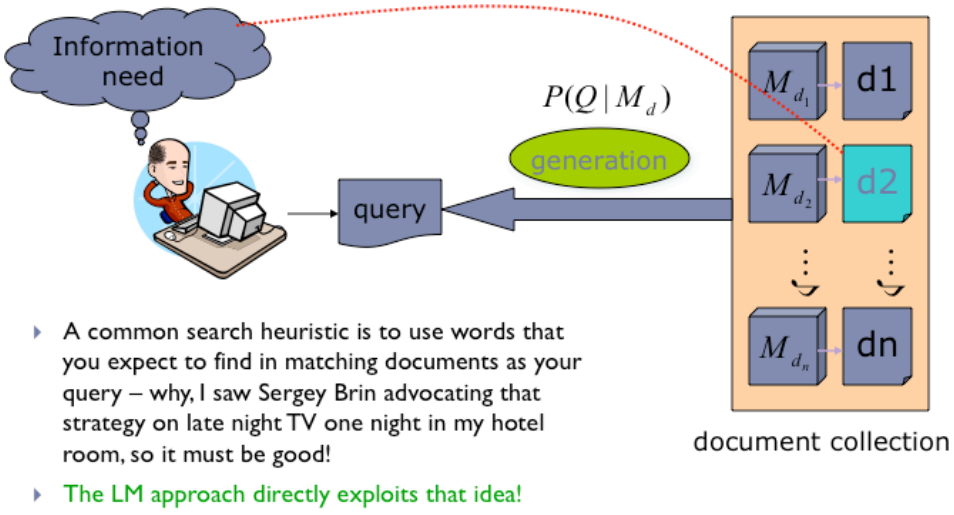
(7%)

Oppgave IV (30%)

a) Sammenlikn den probabilistiske modellen og den språkmodellen ("Language Model for information retrieval"). Fokuser på prinsipp, fordelene og ulempene.

Her skal studentene forklare prinsippet med at man rangerer dokumentene i forhold til en spørring ved hjelp av å finne verdiene på P(R | Q, d) i sannsynlighetsmodellen og P (Q | M_d) i språkmodellen.





- A common search heuristic is to use words that you expect to find in matching documents as your query – why, I saw Sergey Brin advocating that strategy on late night TV one night in my hotel room, so it must be good!
- The LM approach directly exploits that idea!

Hovedforskjellene er dermed hvordan modellene "representerer" relevans. LM gjør dette ved å bake dette eksplisitt i dokumentmodellen mens PM modellerer dette mer implisitt. LM er lettere å beregne og er mer intuitivt tiltalende.

Hovedproblemet med LM er:

- LM antar at det er likhet mellom dokument og informasjon. Dette er i seg selv en svakhet
- LM baserer seg på en veldig forenkling av språkmodellen
- Vanskeligere å bruke URF enn i PM
- Vanskeligere å integrere frasesøk, spørsmål og svar spørringer, og boolsksøk.

(6%)

b) Utvidelse av spørringer.

1. Forklar hensiktene med utvidelse av spørringer (queries).

Hensiktene er å forbedre spørringen slik at man får med seg flere svardokumenter, og på den måten forbedrer evnen til å systemet til å finne relevante dokumenter.

(2%)

2. Forklar Standard Rocchios søkeforbedringsformell.

Her skal følgende formel forklares:

$$\vec{q}_m = \alpha\vec{q} + \frac{\beta}{|D_r|}\sum_{\vec{d}_j \in D_r}\vec{d}_j - \frac{\gamma}{|D_n|}\sum_{\vec{d}_j \in D_n}\vec{d}_j$$

vectors for the **relevant** documents

vectors for the **irrelevant** documents

(4%)

c) Hva er hovedforskjellen mellom ”automatic local analysis” og ”automatic global analysis”?

Begge er brukt i forbindelse med søkefordring ved hjelp av bruken av thesaurus. Hovedforskjellen er at den ALA bruker resultatet fra søket som grunnlag for lage denne thesaurus’en mens AGA bruker hele samlingen.

(4%)

d) Forklar prinsippene bak invertert indeks (inverted index). Bruk gjerne eksempel og figur for å støtte forklaringen din.

Her er det nok at studentene forklarer hvordan man lager en posting list… Det er også riktig hvis studentene heller vil bruke den ”gamle” metoden fra Modern Information Retrieval boka.

(6%)

e) Hva er suffiksstrengene for følgende tekst. Konstruer et ”suffix trie” og ”suffix tre” basert på denne tekseten.
”South Korea says it has returned fire after North Korea fired dozens of artillery shells at one of its border islands, killing two marines”

Her skal det legges mer vekt på at studentene viser at de har forstått prinsippet enn at de viser detaljekunnskap. Det er nok å vise deler av strengene, trie’et og treet

Her antas at vi allerede har fjernet stoppord.

Suffix strings:

South Korea … marines

Korea … marines

says … marines

returned … marines

fire … marines

North … marines

Korea … marines

.

.

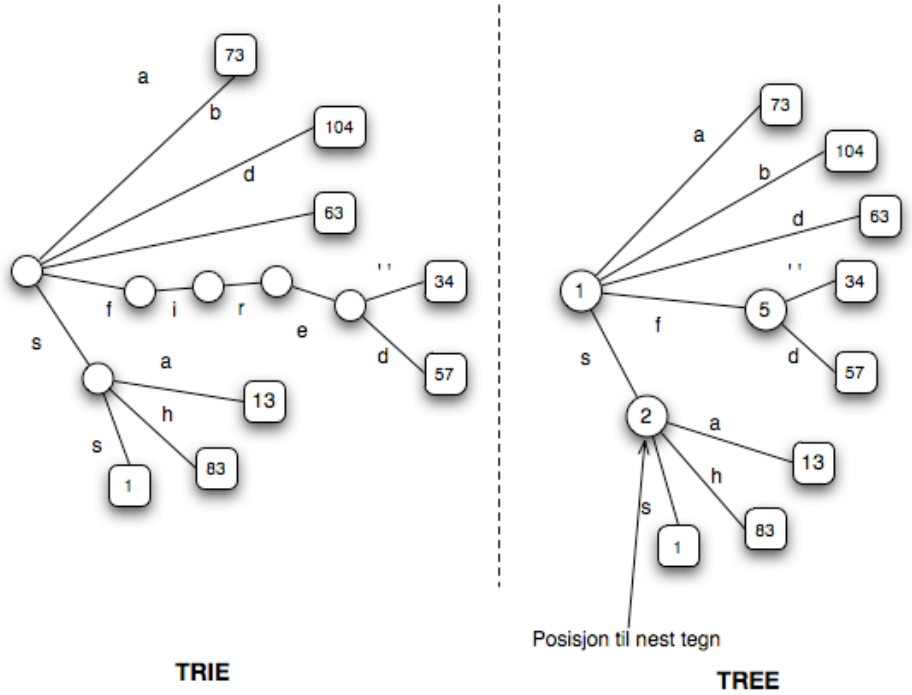
marines

Suffix Trie og Tree:

Her er oversikt over hvor orden finnes i teksten (occurrences)

artillery	73
border	104
dozens	63
fire	34
fired	57
…	
says	13
shells	83
south	1

Dette skal brukes til å lage trie og tree som følgende



(8%)

Oppgave V (20%)

a) Forklar hva vi må gjøre med indekseringen og indekstermene for å kunne få til wildcard-søk ("queries").

Her skal det forklares at dette kan gjøres mulig ved blant å indeksere permuterm, eg. hello$, ello$h, llohe, lohel, o$hell … Det er også et pluss hvis studentene nevner bruken av k-gram indexering.

(6%)

b) Forklar hva begrepet *"edit distance"* er. Hva brukes det til? Hvis vi har en tekststrengen s_i, hvor $len(s_i)$ er lengden på s_i. Vis at *"edit distance"* mellom s_1 og s_2 ikke kan være større enn max($len(s_1)$, $len(s_2)$).

Edit distance: Gitt to tekststreng s1 og s2, edit distance er antall operasjoner som trenges for å gjøre s1=s2, hvor en operasjon kan være sletting av tegn og/eller legge til et tegn. Jfr. defenisjonen kan ikke edit distance være større enn den lengste strengen.

(4%)

c) Forklar begrepet "Jaccard Coefficient". Hvorfor egner ikke denne seg så godt til rangering av søkeresultater?

Gitt to sett A og B, jaccard$(A,B) = |A \cap B| / |A \cup B|$
Det viktigste er at denne måler overlapp mellom to sett som kan i IR være to dokument med sett av termer. I så måte ville den gi en indikasjon på likhet mellom dokumentene, men den tar ikke hensyn til termfrekvenser eller termvekter. Den trenger dessuten en mer sofistikert normalisering av lengder…

(4%)

d) Forklar to måter å få til frasesøk ("phrase queries") på. Tips: Fokuset her er indeksering og posting list.

Her skal man forklare man kan få dette til ved å bruke bi-word index og lagre posisjon i posting listen.

(6%)