NTNU
Kunnskap for en bedre verden

Institutt for Datateknikk og informasjonsvitenskap

Eksamensoppgave i TDT4117 Informasjonsgjenfinning

Faglig kontakt under eksamen: Heri Ramampiaro

Tlf.: 990 27 656

Eksamensdato: 08.12.2015

Eksamenstid (fra-til): 09 - 13

Hjelpemiddelkode/Tillatte hjelpemidler: D: Ingen trykte eller håndskrevne hjelpemidler tillatt. Bestemt, enkel kalkulator tillatt.

Annen informasjon:

Sidene 5 til 7 skal fylles ut og leveres sammen besvarelsen din

Målform/språk: Sensurveiledning

Antall sider: 10

Antall sider vedlegg: 0

Kontrollert av:

Dato Sign

***Svar kort og konsist** på alle spørsmålene. **Stikkord foretrekkes fremfor lange forklaringer**. Les igjennom hele oppgavesettet før du begynner å løse oppgavene. Gjør rimelige antagelser der du mener oppgaveteksten er ufullstendig og skriv kort hva du antar. Lykke til!*

Oppgave I - Lett blanding (20%)

Ola Nordman er nettopp ferdig med masterutdannelsen sin. Han fikk en ide om å starte en egen bedrift som et enmannsforetak. Før han kunne starte opp må han sette seg inn i en del lover og regler i forhold til skatt, plikter og ansvarsforhold. Ola vet at Brønnøysundregistrene var et perfekt sted å starte, i stedet for å lese gjennom boka "Norges Lover".

Basert på denne lille historien skal du svare på følgende delspørsmål:

1. Ola ønsker å finne all relevant informasjon som omhandler skatteregler for enmannsforetak. Drøft kort hvorfor det er viktig for Ola at de som har utviklet søkesystemet for Brønnøysundregistrene har fokusert på høyst mulig "recall" fremfor "precision".

   Svar: Med søk i regler (lov og regler) er det viktig med presedens. Derfor er det viktig å finne alle relevante regler. Pga dette er det viktig med høyest mulig **recall** fremfor precision. Full pott hvis man forklarer hva recall betyr.

2. Ola vil etterhvert trenge å registrere bedriften sin i Enhetsregisteret, men før han gjør dette bestemte han seg for å finne ut hvordan andre har gjort dette før. Er problemet til Ola en "*datagjenfinning*" eller "*informasjonsgjenfinning*"? Begrunn svaret ditt.

   Svar: Ola ønsker å søke om tidligere erfaringer. Derfor er det viktig med **relevans**, **delvis match**, og at skal kunne klare å finne informasjon selv om ha ikke kjenner til *riktig* søkeord. Selv om registeret kunne vært implementert som lagring av strukturert informasjon som i en database er Olas problem "informasjonsgjenfinning". (Datagjenfinning blir ikke feil dersom svaret er velbegrunnet med karakteristikkene på datagjenfinning).

3. Ola finner ut at Brønnøysundregistrene lagrer informasjonen både veldig strukturert og ustrukturert. Han finner også ut at de har gjort det veldig enkelt å bla gjennom dokumenter via linker. Han mistenker at når han søker generelle informasjon om bedrifter så bruker de nettopp denne *linkinformasjonen* til å rangere søkeresultatene. Anta at den som er blitt brukt er "*PageRank*", som er angitt i følgende formell:

$$PR(p_i) = \frac{1-d}{N} + d \cdot \sum_{p \in M(p_i)} \frac{PR(p_j)}{L(p_j)}$$

   Anta at d = 0.15.

   a) Forklar hovedideene bak "PageRank" ved å tolke formelen ovenfor.

Svar: Hovedideene: rangering basert på hvor mye en side blir linket til og lenker til andre sider. Den simulerer sannsynligheten for at en bruker vil klikke på en lenke til en side.

d: er en såkalt **damping factor (**et tilfeldig weighting factor).

PR(pi): PageRank of page pi, p1, ..., pn: pages that point to the page a

L(pi) is the number of outbound links from page *pi*.

M(pi) is the set of links to page *pi*

b) Se for deg at du har en linkstruktur som vist på Figur 1 på de dokumentene som er returnert fra søket. Vis hvordan du regner ut og rangerer dokumentene ved hjelp av PageRank. Bruk formelen ovenfor. Gjør også de antakelsene du finner nødvendig.

Figur 1 Linker mellom dokumentene

Svar: Her er det viktigste at studentene viser hvordan de forstår prinsippet i praksis. Dvs. viser de at de forstår formelen og kan bruke den er svaret godkjent.

c) Nevn og forklar kort to andre metoder som utnytter linkstruktur til å rangere søkeresultater.

Svar: HITS (bruker authority og hubs + forklaring av disse er forventet) og Most-Cited (tidlig metode basert på popularitet men bruker info om hvor mange sider som linker til en gitt side som basis for rangeringen).

4. Ola stusser litt på at man bruker typiske websøkemetoder til å håndtere søk i registrene. Han tenker at det ville være best med tradisjonelle informasjonsgjenningsbaserte metoder. Drøft kort hovedforskjellene mellom et typisk websøkesystem og et tradisjonell informasjonsgjenningssystem.

Svar: I Web: Kun indeksene er tilgjengelige (ikke dokumentene), rangering av søkeresultater basert på link info., mengden av informasjon er mye større, heterogenitet i innhold, etc. Et viktig poeng er at dokumenter indeksene er basert på plutselig kan forsvinne.

I tradisjonell IR: mer homogene dok.samlinger, dokumentene er stort sett tilgjengelige i tillegg til indeksene, rangeringsmetoder er i hovedsak basert på dokumentinnhold.

Oppgave II Tekstoperasjoner og Indekseringsteknikker (30%)

1. Forklar hvorfor man mener "stemming" er noe kontroversiell spesielt i forbindelse med websøk. Forklar tre andre tekstoperasjonsmetoder som man kan bruke.

Svar: I Websøk er hovedfokus å få **så høyst mulig precision** som mulig. Stemming øker mest recall. Stemming kan derfor koste mer enn den smaker.

Forventer gode forklaringer på "stoppordfjerning", "leksikalanalyse", "thersaurusbygging" og/eller valg av indekstermer.

2. Anta at vi har følgende tekst. Gjør dine antakelser med hensyn til stoppordfjerning.

"**President Barack Obama** warned his **Russian counterpart** Tuesday against **intervening** in **Syria**'s **civil war**, suggesting that **Vladimir Putin** is aware of the **dangers** his **country** faces by entering the **conflict**"

I svarene under antar vi at vi fjerner stoppord først.

a) Konstruer ***invertert file/liste*** for teksten ovenfor.

Svar: Her skal man lage en vocabulary-occurence oppsett.

Vocabulary	Occurrences
barack	11
civil	94
conflict	194
counterpart	43
country	164
dangers	152
intervening	71
obama	18
president	1
putin	130
russian	35
syria	86
vladmir	121
war	100

b) Bruk teksten over til å konstruere invertert liste med blokkadressering.

Svar: Anta blokkstørrelse på 4 ord.

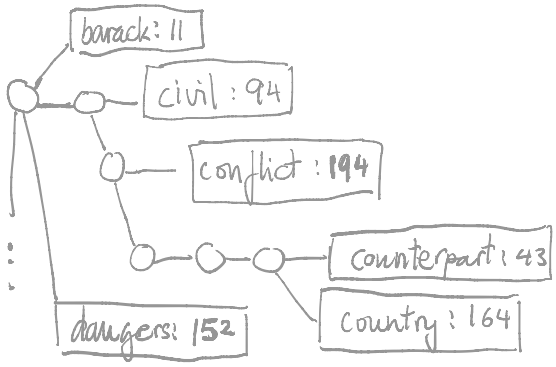
Da får vi:

President Barack Obama warned | his **Russian counterpart** Tuesday | against **intervening** in **Syria**'s | **civil war**, suggesting that | **Vladimir Putin** is aware | of the **dangers** his | **country** faces by entering | the **conflict**"

Dvs. 8 blokker, feks. blokk 1 vil nå bestå av "president, barack, obama, og warned". Basert på dette får vi:

Vocabulary	Occurrences
barack	1
civil	4
conflict	8
counterpart	2
country	7
dangers	6
intervening	3
obama	1
president	1
putin	5
russian	2
syria	3
vladmir	5
war	8

c) Konstruer et partielt vokabular "***trie***" av teksten over.



NB: Denne kan blandes med "Suffix Trie" som i dette tilfellet ***er feil***!

3. Forklar hvordan signaturfilindekseringsteknikken fungerer. Bruk gjerne eksempel til å støtte forklaringen din.

Svar: Her skal man forklare hvordan man 1. bestemme hashfunksjonene til indekstermene vi har valgt over. Deretter velge en blokkstørrelse (feks. som før 4) og lage en maske/ signatur for hver blokk vha. bitvis "or-ing" av hashfunksjonene til hver blokk, etc. Full pott dersom man viser forståelsen med et godt eksempel.

Oppgave III Similaritetsmodeller og Evaluering (30%)

1. Anta at vi har følgende dokumenter:

```
D1 = "George Bush is former American President, but he is still
called president"
D2 = "President Barack Obama's presidential period will soon be
over, and yet another Bush may well become a new president"
```

Anta at spørresetningen q= "`american president bush`" er brukt til å søke på dokumentene.

a) Finn alle nøkkelordene (som kan brukes som index terms) i de to dokumentene og sett opp vokabularsettet K. Gjør de antakelsene du finner nødvendige.

Svar: For enkelhetsskyld antar vi at vi har en aggressiv stoppordfjerner og bruker stemming.
K = {america, barack, bush, george, obama, period, president}.

b) Bruk formelen nedenfor og vis hvordan du regner ut likhetsfaktoren (similarity), *sim(q, D),* mellom dokumentene og spørresetningen ved hjelp av "vector space" modellen. Hvilket av de to dokumentene blir rangert først?

$$Sim(q,d_j) = \cos(\theta) = \frac{\vec{d_j} \bullet \vec{q}}{|\vec{d_j}| \cdot |\vec{q}|} = \frac{\sum_{i=1}^{t} w_{ij} w_{iq}}{\sqrt{\sum_{i=1}^{t} w_{ij}^2} \sqrt{\sum_{i=1}^{t} w_{iq}^2}}$$

Svar: Anta at vi bruker rå frekvensen som vekt siden dokumentene er ganske korte. Dette gir oss følgende vektorer: d1=[1, 0, 1, 1, 0, 0, 2], d2=[0, 1, 0, 0, 1, 1, 3], q=[1, 0, 1, 0, 0, 0, 1]

sim(q, d1) = (1*1 + 1*1 + 2*1)/(sqrt(1+1+1+4)*sqrt(1+1+1)=4/sqrt(7)*sqrt(3))= 0.87

sim(q, d2) = (3*1)/sqr(1+1+1+9)sqrt(3) = 3/sqrt(12)sqrt(3) = 0.50

d1 blir rangert før d2.

c) Hva tror du er grunnene til a denne modellen er mer populær enn sannsynlighetsmodellen?

Svar: Denne modellen er mer deterministisk og enklere å beregne. Sannsynlighetsmodellen er avhengig av at estimeringene og antakelsene man legger til grunn er gode.

2. Forklar hovedforskjellene mellom språkmodellen (Language model for information retrieval) og sannsynlighetsmodellen.

   Svar: Språkmodellen: sannsylighet for at et dokumentmodell genererer en spørring. Sannsynlighetsmodell: baserer seg på sannsynlighet for at et dokument man finner for et gitt spørring er relevant.

3. Du skal gjennomføre en evaluering av et gjenfinningssystem, og etablerer en eksperimentsamling på 1000 dokumenter. Evalueringen skjer ved at det utarbeides et sett med spørringer som kjøres mot eksperimentsamlingen, og at det utarbeides en såkalt "ground truth" eller "fasit" med dokumenter som anses for relevante for hver spørring. I forhold til et gitt spørsmål finner du at 15 av dokumentene i eksperimentsamlingen er relevante. Vi antar at spørringen har i alt 20 relevante dokumenter. Du finner disse dokumentene på rangnummer 1, 3, 6, 8, 9, 12, 15, 24, 36, 42, 43, 45, 50 , 54, og 60 i en rangert treffliste. Ta utgangspunkt i første 15 returnerte dokumenter i resultatlista og regn ut precision- og recall-punktene. Hva blir *f-measure* (harmonic means) verdien?

   Svar: Her skal studentene regne ut **precision og recall på punktene: 1, 3, 6, 8, 9, 12, og 15**.

Precision & Recall points

Doc		Precision	Recall
1	Rel	1.00	0.05
2			
3	Rel	0.67	0.15
4			
5			
6	Rel	0.50	0.3
7			
8	Rel	0.50	0.4
9	Rel	0.56	0.45
10			
11			
12	Rel	0.50	0.6
13			
14			
15	Rel	0.47	0.75

Eller: p@1=1/1 og r@1=1/20, p@3=2/3 og r@3=2/20=1/10, p@6 = 3/6 = 1/2 og r@6=3/20, p@8 = 4/8 = 1/2 og r@8=4/20=1/5, p@9 = 5/9 og r@9=5/20=1/4, p@12=6/12 = 1/2 og r@12=6/20=3/10, p@15=7/12 og r@15=7/20.

F-measure = 2PR/P+R
= 2(15/60)*(15/20)/(15/60+15/20)

= 2*0.25*0.75/(0.25+0.75) = **0.375**

Her antar vi at det er kun de 60 som er returnerte resultater. Dvs.
P=15/60 = 0.25, og R=15/20

4. Forklar kort hvorfor "recall" er viktigere enn "precision" for Lovdatasøk, mens “precision” er viktigere enn “recall” for søk i Gulesider o.l .

   Svar: I søk i regler (lov og regler) er det viktig med presedens. Derfor er det viktig å finne alle relevante regler. Pga dette er det viktig med høyest mulig **recall** fremfor precision. I Gulesider derimot er man mest opptatt av at minst de 10 første returnerte treffene er relevante. Derfor er precision viktigst.

Kandidatnr.:_____

Følgende ark skal du levere sammen med besvarelsen. Husk derfor å føre på ditt kandidatnummer.

Oppgave IV (20%)

I hver av følgende deloppgaver skal du krysse av ***et*** svar. Selv om du mener det kan være flere enn en påstand som er riktige skal du ***ikke krysse av mer enn et svar***. (**Alle delspørsmål teller likt. Riktig svar gir 2 poeng**)

1.

- [x] Micon og videogjenfinning hører sammen
- [ ] Micon og lydgjenfinning fra video shots hører sammen
- [ ] Micon kan brukes av PST til å lett gjenfinne bilder med terrorister
- [ ] Micon har ingen ting med informasjonsgjenfinning å gjøre

2.

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|} = \frac{|A \cap B|}{|A| + |B| - |A \cap B|}.$$

- [ ] Denne formelen er et likhetsmål som egner seg veldig godt til bruk i rangering av resultater med søk av store strukturerte dokumenter
- [ ] Denne formelen forteller noe om mengden av informasjon som er i dokumentene
- [ ] Denne formelen er en Jaccard likhetsmål som ikke kan brukes til å vurdere om to ord er like eller ikke
- [x] Denne formelen er en Jaccard likhetsmål som kan brukes til å vurdere om hvor stor grad to dokumenter er like

3.

- [ ] Et Huffman tre er en variant av et suffix tre
- [ ] Et Huffman tre er et komprimeringstre som kan være både ”bit” basert (binært) og ”byte” basert samtidig.
- [ ] Et Huffman tre en variant av et suffix trie
- [x] Et Huffman tre konstruksjon bruker informasjon om termdistribusjon i et dokument for å kunne lage en så effektiv komprimering som mulig

4.

- [ ] Automatic Global Analysis og Automatic Local Analysis bruker to motsatte metoder for utvidede spørringer
- [x] Pseudo relevance feedback er en metode for å utvide spørringer
- [ ] I Pseudo relevance feedback trenges tilbakemelding fra brukeren for at det skal fungere optimalt
- [ ] Med Pseudo relevance feedback trenger man alltid å bruke Rocchios standard metode for få det til å fungere

5.

- [ ] IDF står for “Information of Document Frequency”
- [ ] IDF står for “Invariant Document Frequency” og brukes til å måle hvor mye svingninger er det i antall termer per dokument
- [ ] IDF står for Inverse Document Frequency og kan brukes til å straffe termer som nevnes ofte i et dokument
- [x] IDF står for Inverse Document Frequency og kan brukes til å straffe termer som nevnes ofte i en samling av dokumenter

6.

- [x] Zipf's law kan brukes til å vurdere hvilke ord som kan være gode kandidater til stoppordlista
- [ ] Zipf's og Heap's law er viktig for genfinningsregler i websøkesystemer
- [ ] En crawler bruker Zipf's law til å avgjøre om den skal følge en link videre til neste dokument, for indeksering
- [ ] Zipf's law sier ingenting om termdistribusjon i en samling

7.

- [ ] Harverst websøkesystem er basert på et sentralisertsystemarkitektur med koordinerte crawlers
- [x] Harverst websøkesystem er basert på et distribuertsystemarkitektur med koordinerte gatherers
- [ ] Harverst er ingen websøkesystem men en proprietær systemarkitektur for datahøsting
- [ ] Harverst-arkitekturen er helt lik den arkitekturen som Google har brukt som basis for deres websøkearkitektur

8.

- [x] Et bildegjenfinningssystem består av en "feature extractor"-del som gjør det enkelt å automatisere lagring av informasjon om innholdet i et bilde
- [ ] Et bildegjenfinningssystem må bruke alle teknikker fra databaseteknikk for å kunne enkelt utføre spørringer
- [ ] Det er et krav at et bildegjenfinningssystem tilbyr støtte til nøkkelordsøk
- [ ] De fleste bildegjenfinningssystem bruker piksel-til-piskelsammenlikninger til å finne likhet mellom bilder og til rangering

9.

- [x] Okapi BM25 er en variant av sannsynlighetsmodellen men bruker både TF og IDF til å estimere sannsynlighetene
- [ ] Okapi BM25 er en variant av språkmodellen (Language model for information retrieval)
- [ ] Ifølge forskningen fungere Okapi BM25 mye dårligere enn boolskmodellen
- [ ] Okapi BM25 har ingenting med søkeresultatrangeringer å gjøre

10.

- [ ] Precision og recall er like viktige uavhengig av søkeapplikasjoner
- [ ] Recall er typisk viktigere enn precision for søk i Gulesider
- [ ] MAP bruker ikke recall i det hele tatt
- [x] Interpolering er nyttig dersom man har for få recall-punkter