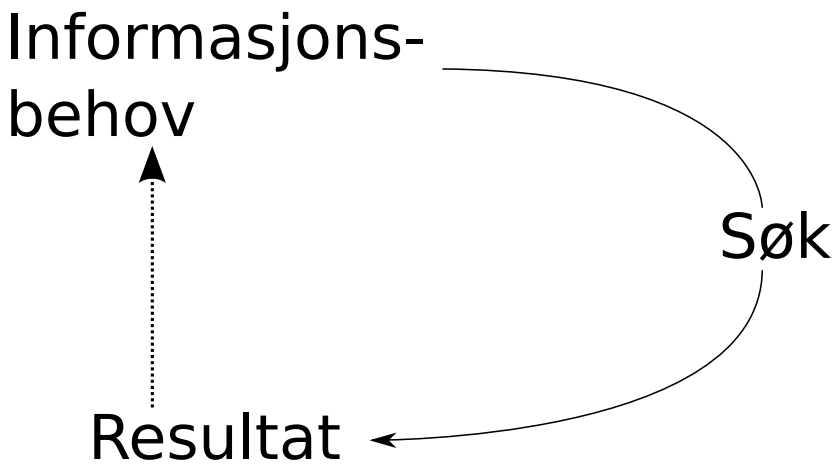
INFO221v12
IR VI

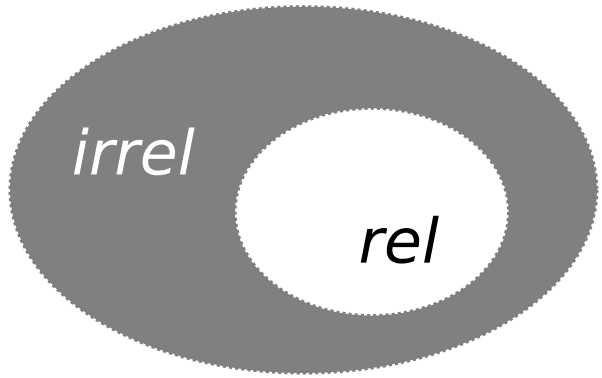
Truls Pedersen
Institutt for informasjons- og medievitenskap
Universitetet i Bergen

Oversikt

- Husker: Evaluering av IR systemer
  - Urangerte resultatmengder
- Rangerte resultatmengder
- Relevansvurdering
- Brukernytte
- Raffinere søkeresultatene
- Relevance feedback (*RF*)
  - Rocchio algoritmen
- Lokal vs. global
- Spørringutvidelse

IR systemer

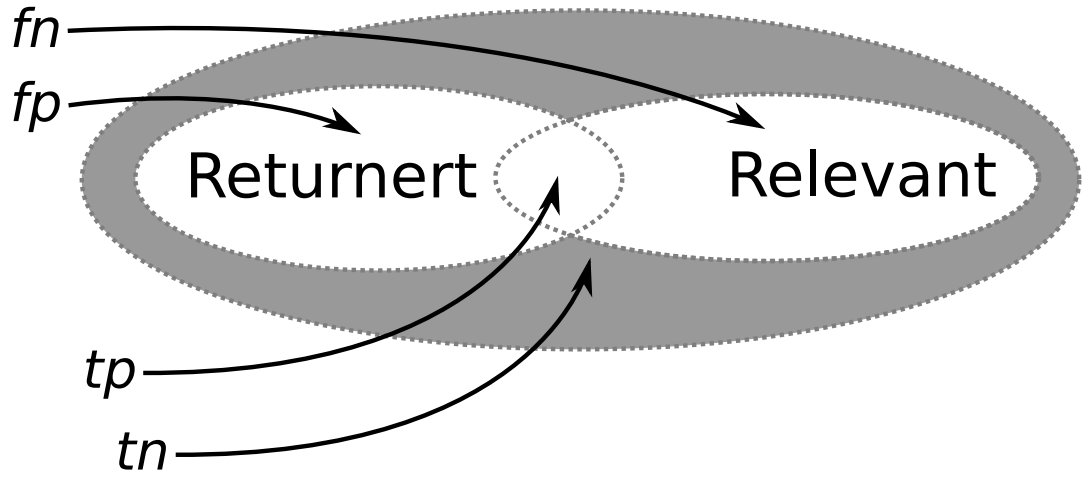




Vi utfører et søk for å møte et *informasjonsbehov*.

Et resultat er *relevant* hvis, og bare hvis, det svarer til *informasjonsbehovet*, og er ikke avhengig av hvorvidt det er et rimelig resultat for et søk.

Urangerte resultatmengder



Nøyaktighet

$$P = \frac{tp}{tp + fp}$$

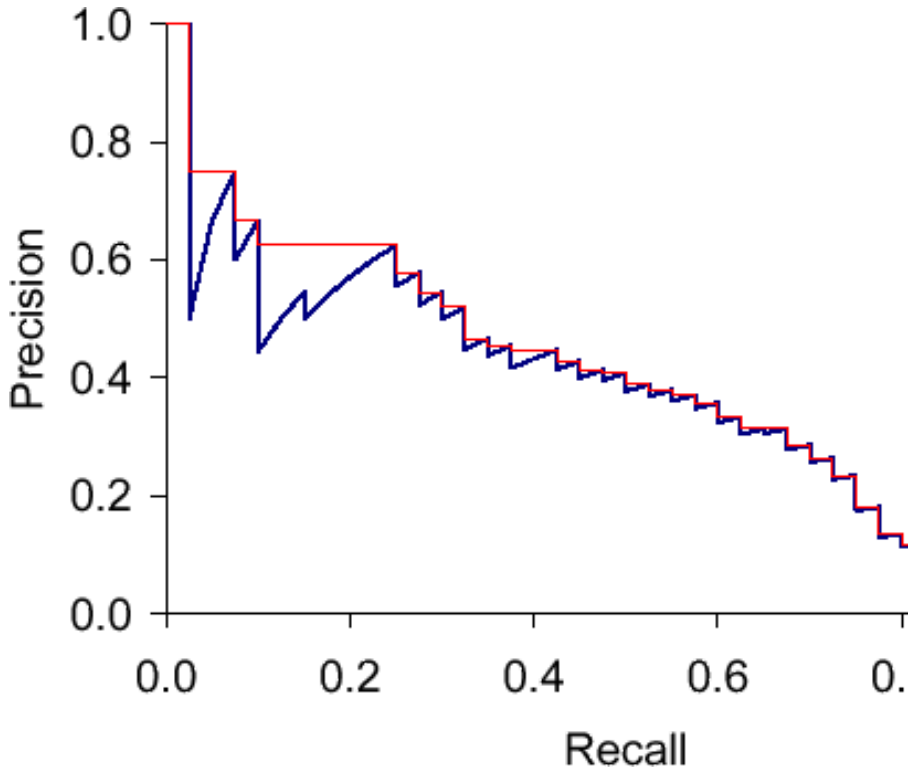
Gjennkalling

$$R = \frac{tp}{tp + fn}$$

Rangerte datamengder

Hvis vi rangerer *alle* dokumentene, vil mengden av dokumenter med de k høyest poengene ha et P og R mål.

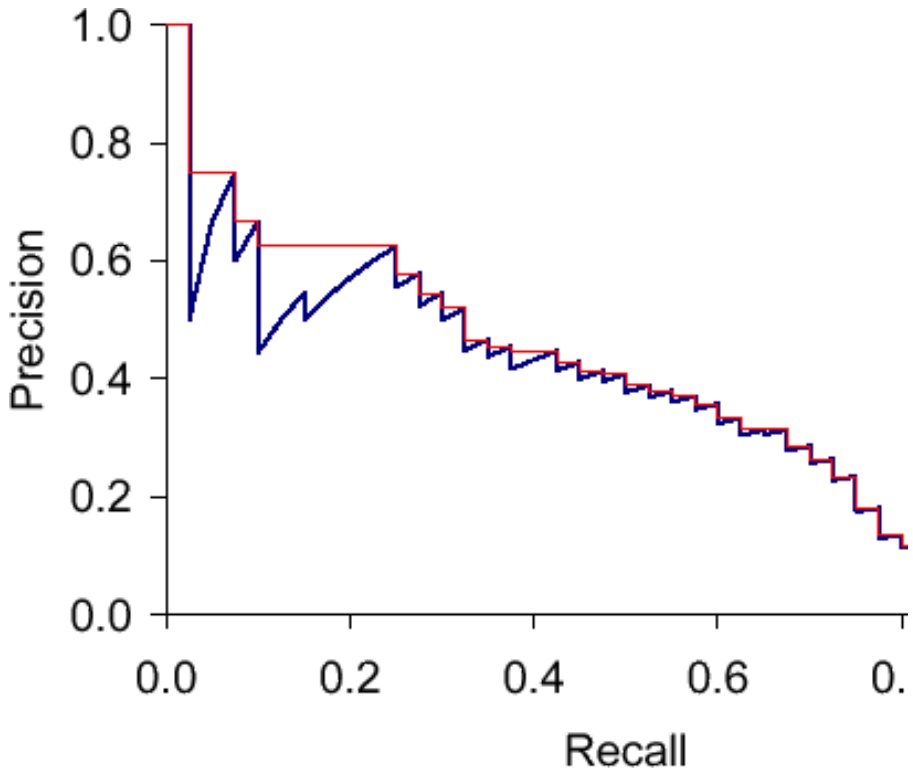
Vi kan gradvis øke k og se hvordan P og R utvikler seg (blå):



Rangerte datamengder

Interpolert nøyaktighet (en bruker klikker gjerne “vis flere” hvis det øker nøyaktighet, rød).

$$p_{\text{interp}}(r) = \max_{r' \geq r} p(r')$$

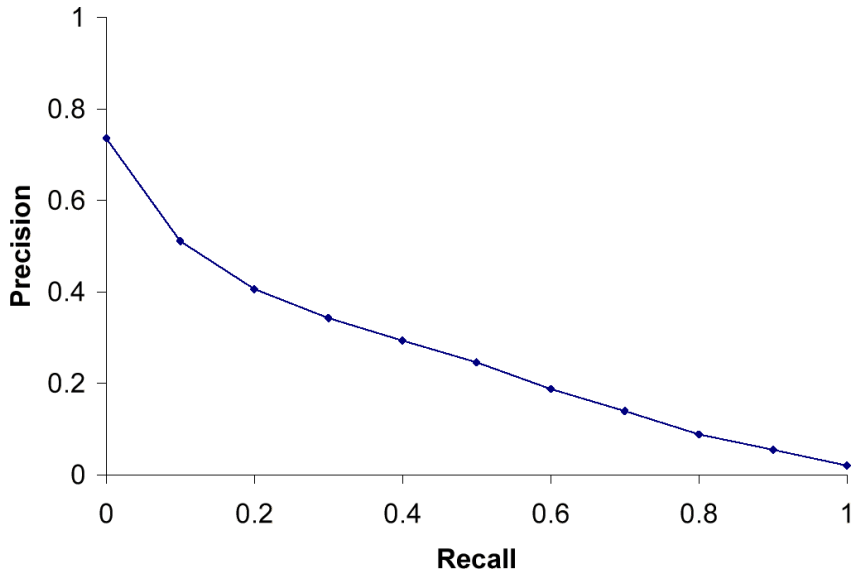


Rangerte datamengder

Eleven-point interpolated average precision

For å få et inntrykk av hvordan systemet i sin helhet fungerer, finner vi slike *P*-*R*-grafer for flere informasjonsbehov.

For hver av disse informasjonsbehovene utformer vi en spørring q_j og finner vi en *P*-*R*-graf.



Rangerte datamengder

Mean average precision (MAP)

La

- q være spørringen for et informasjonsbehov,
- $D_R = \{d_1, \ldots, d_l\}$ være de relevante dokumentene for q ordnet synkende etter rangeringen, og
- R_k være (den minste) mengden av søkeresultater for q som inneholder d_k

$$\text{"}AP\text{"} = \frac{1}{l} \sum_{k=1}^{l} P(R_k)$$

$P(R_j)$ er nøyaktigheten for (den minste) mengden som inneholder dokument d_j, eller 0 hvis det ikke finnes en slik mengde.

Rangerte datamengder

Mean average precision (MAP)

Vi har fått en mål “AP” for hvor godt systemet egner seg til å søke etter én spørring, q.

For å få et anslag for hvor godt systemet er generelt gjør vi samme beregning for en mengde med spørringer, Q, og tar gjennomsnittet av disse

$$MAP(Q) = \frac{1}{|Q|} \sum_{j=1}^{|Q|} \frac{1}{m_j} \sum_{k=1}^{m_j} \text{Precision}(R_{jk})$$

Rangerte datamengder

MAP estimatet kan variere veldig for et gitt system hvis vi velger andre søk.

Som regel er det mer varians for ett system med forskjellige søk, enn mellom to systemer med like søk.

Det finnes naturligvis mange mål for å estimere systemene:

Det er ofte ønskelig å måle nøyaktighet gitt en begrenset gjennkalling, men dette kan være et ustabilt mål.

Det finnes selvsagt andre tester som forsøker å være mer rettferdig, mer stabil, og pålitelig.

Relevansvurdering

Når vi skal teste systemene våre må vi ha

- informasjonsbehovtester som vil være relevante informasjonsbehov i det endelige systemet, og
- testdata som er representative for den endelige datamengden.

Før testdataene er klare må vi ha

- *relevansvurderinger* for dokumentene i forhold til informasjonsbehovene.

Relevansvurdering

Pooling

En ofte brukt metode for å finne en mengde relevante dokumenter for et informasjonsbehov er å samle inn de beste k resultatene fra en mengde forskjellige IR systemer.

Metoden kan suppleres med Boolsk søk eller dokumenter utpekt av eksperter.

Vurderinger gjort av mennesker er subjektive. Hvis hvert dokument bare klassifiseres av et menneske kan noen dokumenter utelukkes/innlemmes selv om andre ville vært uenig.

Relevansvurdering

I hvor stor grad er to mennesker enig om en samling relevansvurderinger?

$$\text{kappa}(\kappa) = \frac{P(A) - P(E)}{1 - P(E)}$$

hvor $P(A)$ er andelen av tilfellene hvor agentene er enige, $P(E)$ er sannsynligheten* for at de er enig om de velger helt tilfeldig.

Relevansvurdering - eksempel

Vi har 400 dokumenter som to mennesker skal vurdere. Er disse 2×400 vurderingene enige dersom ...

		Vurd.2		
		ja	nei	$\sum$
Vurd.1	ja	300	20	320
	nei	10	70	80
	$\sum$	310	90	400

Hvor stor andel er de enige, $P(A)$?

Relevansvurdering - eksempel

Vi har 400 dokumenter som to mennesker skal vurdere. Er disse 2×400 vurderingene enige dersom ...

		Vurd.2		
		ja	nei	$\sum$
Vurd.1	ja	300	20	320
	nei	10	70	80
	$\sum$	310	90	400

Hvor stor andel er de enige, $P(A)$?

$$P(A) = \frac{300 + 70}{400} = 0.925$$

Hvor mange relevante dokumenter er det?

Relevansvurdering - eksempel

Vi har 400 dokumenter som to mennesker skal vurdere. Er disse 2×400 vurderingene enige dersom ...

		Vurd.2		
		ja	nei	$\sum$
Vurd.1	ja	300	20	320
	nei	10	70	80
	$\sum$	310	90	400

Hvor stor andel er de enige, $P(A)$?

$$P(A) = \frac{300 + 70}{400} = 0.925$$

Hvor mange relevante dokumenter er det?

$$\left(\frac{310}{400} + \frac{320}{400}\right) \cdot \frac{1}{2} = 0.7878$$

Relevansvurdering - eksempel

Vi har 400 dokumenter som to mennesker skal vurdere. Er disse 2×400 vurderingene enige dersom ...

		Vurd.2		
		ja	nei	$\sum$
Vurd.1	ja	300	20	320
	nei	10	70	80
	$\sum$	310	90	400

Hvor mange irrelevante dokumenter er det?

Relevansvurdering - eksempel

Vi har 400 dokumenter som to mennesker skal vurdere. Er disse 2×400 vurderingene enige dersom ...

		Vurd.2		
		ja	nei	$\sum$
Vurd.1	ja	300	20	320
	nei	10	70	80
	$\sum$	310	90	400

Hvor mange irrelevante dokumenter er det?

$$\left(\frac{90}{400} + \frac{80}{400}\right) \cdot \frac{1}{2} = 0.2125$$

Gitt et tilfeldig dokument, hvor sannsynlig er det at begge ville svart “ja” eller begge “nei”?

Relevansvurdering - eksempel

Vi har 400 dokumenter som to mennesker skal vurdere. Er disse 2×400 vurderingene enige dersom ...

		Vurd.2		
		ja	nei	$\sum$
Vurd.1	ja	300	20	320
	nei	10	70	80
	$\sum$	310	90	400

Hvor mange irrelevante dokumenter er det?

$$\left(\frac{90}{400} + \frac{80}{400}\right) \cdot \frac{1}{2} = 0.2125$$

Gitt et tilfeldig dokument, hvor sannsynlig er det at begge ville svart “ja” eller begge “nei”?

$$P(E) = 0.2125^2 + 0.7878^2 = 0.665$$

Relevansvurdering - eksempel

Vi har 400 dokumenter som to mennesker skal vurdere. Er disse 2×400 vurderingene enige dersom ...

Vurd.1 \ Vurd.2	ja	nei	$\sum$
ja	300	20	320
nei	10	70	80
$\sum$	310	90	400

Hvor stor enighet er det mellom disse to?

Relevansvurdering - eksempel

Vi har 400 dokumenter som to mennesker skal vurdere. Er disse 2 $\times$ 400 vurderingene enige dersom ...

		Vurd.2		
		ja	nei	$\sum$
Vurd.1	ja	300	20	320
	nei	10	70	80
	$\sum$	310	90	400

Hvor stor enighet er det mellom disse to?

$$\kappa = \frac{P(A) - P(E)}{1 - P(E)} = \frac{0.925 - 0.665}{1 - 0.665} = 0.776$$

Relevansvurdering

Hvis agentene alltid er enig vil $\kappa = 1$.
$0,8 < \kappa < 1,0$: “enig”,
$0,67 < \kappa < 0,8$: “ganske enig”, og
$0,0 < \kappa < 0,67$: “uenig”.

Når disse tallene regnes på (binære) relevansvurderinger for systemer der ute i “virkeligheten” lander vi ofte på “ganske enig”.

Det indikerer at vi ikke trenger å ta i bruk mer detaljert skala for relevans når vi konstruerer testdata.

Brukernytte

Når vi nå har gått gjennom noen forskjellige kvalitetsmål for IR systemer... har det vært verdt det?

Brukernytte

Når vi nå har gått gjennom noen forskjellige kvalitetsmål for IR systemer... har det vært verdt det?

Hvis et system kommer godt ut av en test... betyr det at det er et godt system?

Brukernytte

Når vi nå har gått gjennom noen forskjellige kvalitetsmål for IR systemer... har det vært verdt det?

Hvis et system kommer godt ut av en test... betyr det at det er et godt system?

Disse målene kan hjelpe oss å justere parametre i systemet, og kan i det minste gi en pekepinn i forhold til kvalitet.

Brukernytte

Ingen av metodene vi har sett tar for seg at

- relevansen av et dokument kan være avhengig av de andre dokumentene i resultatmengden,
- vurderingene er subjektive, enkle, binære vurderinger,
- forskjellige mennesker vil vurdere relevans forskjellig (ikke minst mennesker fra forskjellige regioner/alder/...)

Brukernytte

Ingen av metodene vi har sett tar for seg at

- relevansen av et dokument kan være avhengig av de andre dokumentene i resultatmengden,
- vurderingene er subjektive, enkle, binære vurderinger,
- forskjellige mennesker vil vurdere relevans forskjellig (ikke minst mennesker fra forskjellige regioner/alder/...)

Det vi *egentlig* ønsker å måle er hvor *nyttig* systemet er.

Brukernytte

Noen kvantitative målinger vi kan gjøre spiller en direkte rolle for hvordan brukeren oppfatter systemet:

- hvor raskt indekserer systemet dokumenter?
- hvor raskt søker systemet?
- hvor uttrykksfullt er spørrespråket?
- hvor mange dokumenter er tilgjengelig for gjennomsøking?

Brukernytte

Noen kvantitative målinger vi kan gjøre spiller en direkte rolle for hvordan brukeren oppfatter systemet:

- hvor raskt indekserer systemet dokumenter?
- hvor raskt søker systemet?
- hvor uttrykksfullt er spørrespråket?
- hvor mange dokumenter er tilgjengelig for gjennomsøking?

Andre interessante mål vil gjerne være kvalitative egenskaper;

- hvordan bruker en bruker systemet?
- hvordan oppfatter brukeren systemet?

Brukernytte

A/B testing

Etter vi har kjørt alle testene våre på systemet og føler vi er "i mål" gjør vi systemet tilgjengelig for brukerne.

Vi har nå en *stor* mengde informasjonsbehov som vi kan bruke for testing!

Brukernytte

A/B testing

Etter vi har kjørt alle testene våre på systemet og føler vi er "i mål" gjør vi systemet tilgjengelig for brukerne.

Vi har nå en *stor* mengde informasjonsbehov som vi kan bruke for testing!

Vi kan forsøke å gjøre videre fin-justeringer ved å gjøre *A/B-testing*.

Gjør én endring i systemet slik at det finnes to versjoner, og la 1%-10% tilfeldig utvalgte brukere bruke den eksperimentelle versjonen.

Hvis f.eks. flere brukere nå er fornøyd med første søkeresultat, betrakter vi endringen vi gjorde som en forbedring.

Brukernytte

Utsnitt

I mange tilfeller er ikke brukeren interessert i å gå nøye gjennom alle søkeresultatene.

Vi kan minske antall dokumenter brukeren må gå nøye gjennom ved å tilby utsnitt som del av resultatpresentasjonen.

Brukernytte

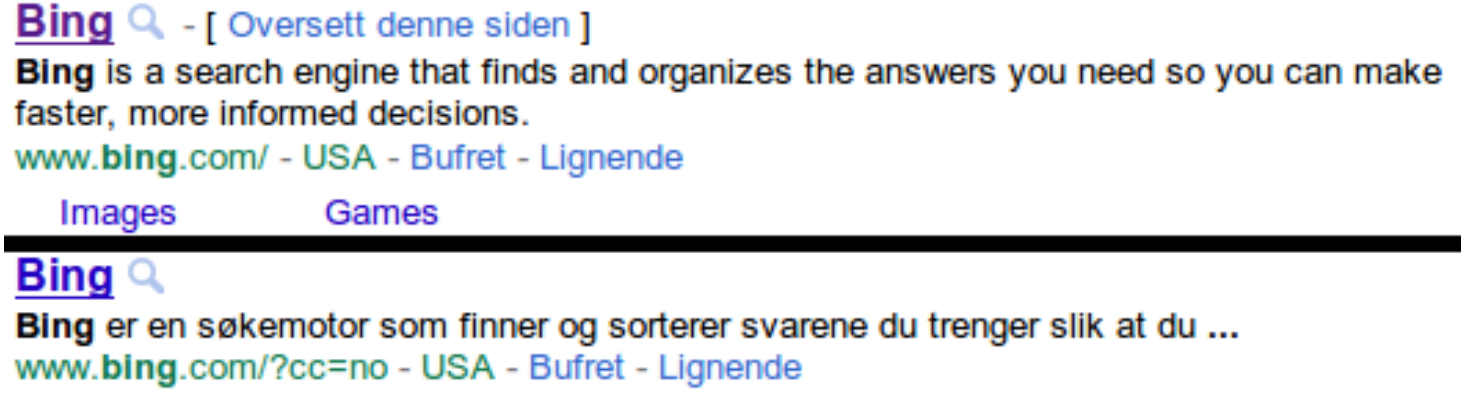
Utsnitt

I mange tilfeller er ikke brukeren interessert i å gå nøye gjennom alle søkeresultatene.

Vi kan minske antall dokumenter brukeren må gå nøye gjennom ved å tilby utsnitt som del av resultatpresentasjonen.

Vi har hovedsakelig to måter å gjøre dette:

- statisk, og
- dynamisk



Brukernytte

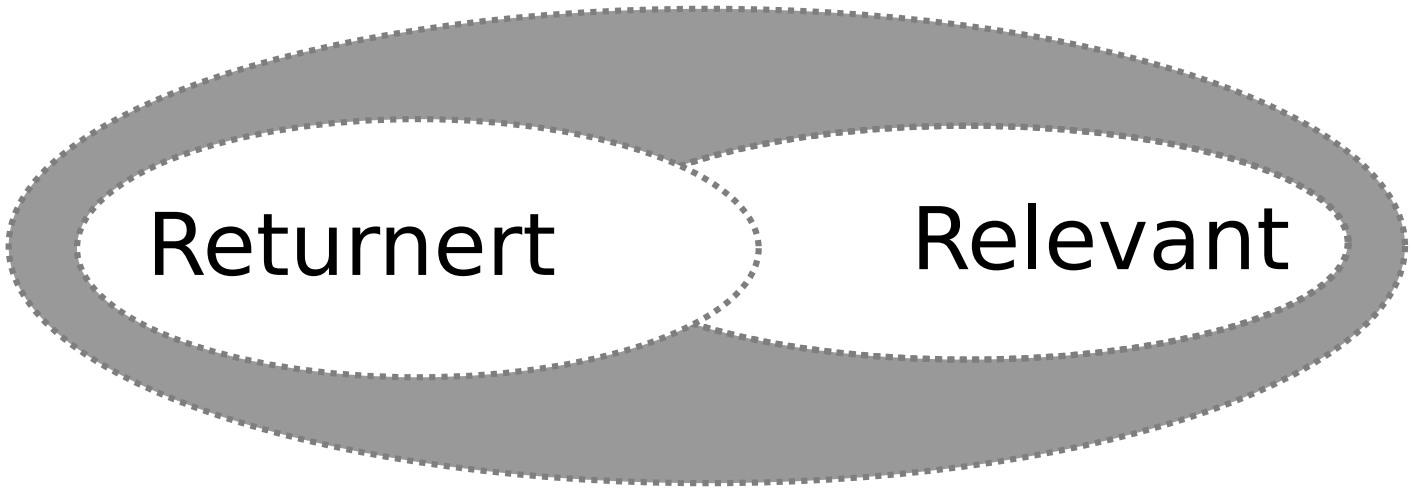
Utsnitt

Det er ingen enkel sak å få en datamaskin til å konstruere et kort sammendrag av den relevante delen av et dokument.

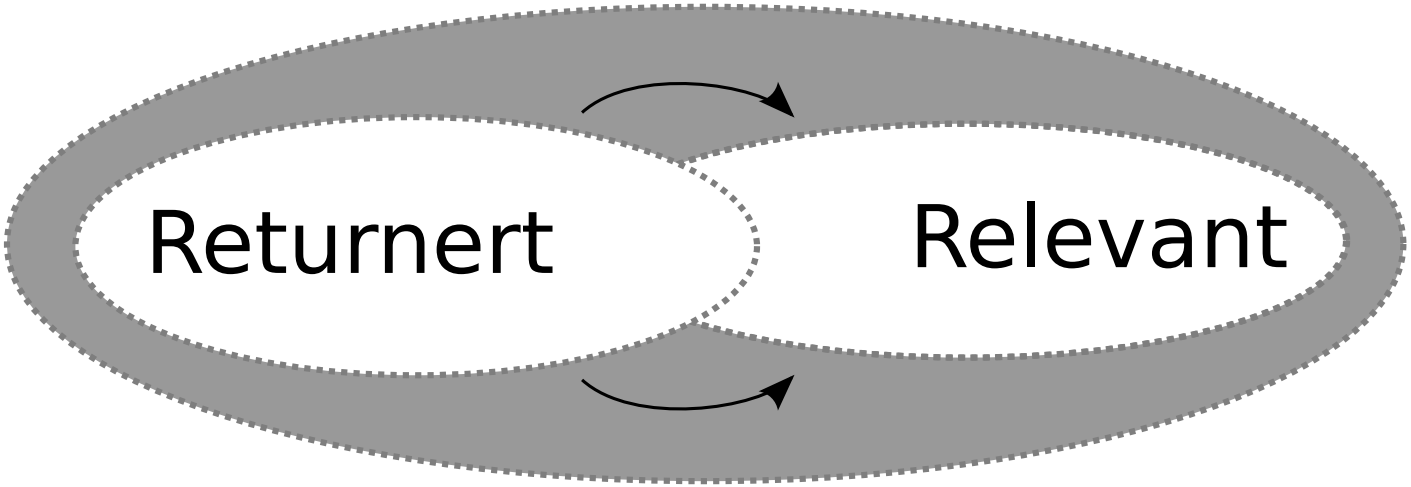
Problemstillingen kalles *text summarization* og er hovedsakelig et problem innen behanding av naturlig språk
(kunstig intelligens, lingvistikk, ...)

Generelt prøver vi å finne et, eller noen få, relevante *vinduer* i teksten.
(*keyword-in-context*, KWIC, utsnitt)

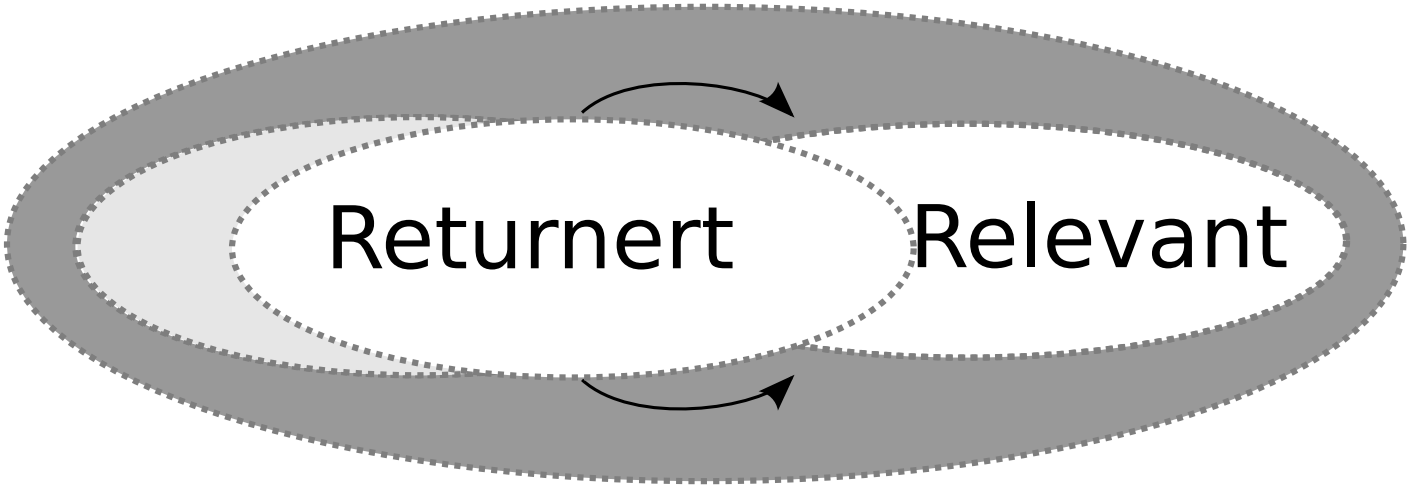
Raffinere søkeresultatene



Raffinere søkeresultatene



Raffinere søkeresultatene



Raffinere søkeresultatene

Vi har sett hvordan vi kan finne dokumenter som er *nær* en spørring.

Vi forsøker å komme så nært som mulig (P og R), men vi treffer aldri helt.

Raffinere søkeresultatene

Vi har sett hvordan vi kan finne dokumenter som er *nær* en spørring.

Vi forsøker å komme så nært som mulig (P og R), men vi treffer aldri helt.

Noen vanlige grunner for dette er

- Skrivefeil
- Synonymitet

Raffinere søkeresultatene

Vi har sett hvordan vi kan finne dokumenter som er *nær* en spørring.

Vi forsøker å komme så nært som mulig (P og R), men vi treffer aldri helt.

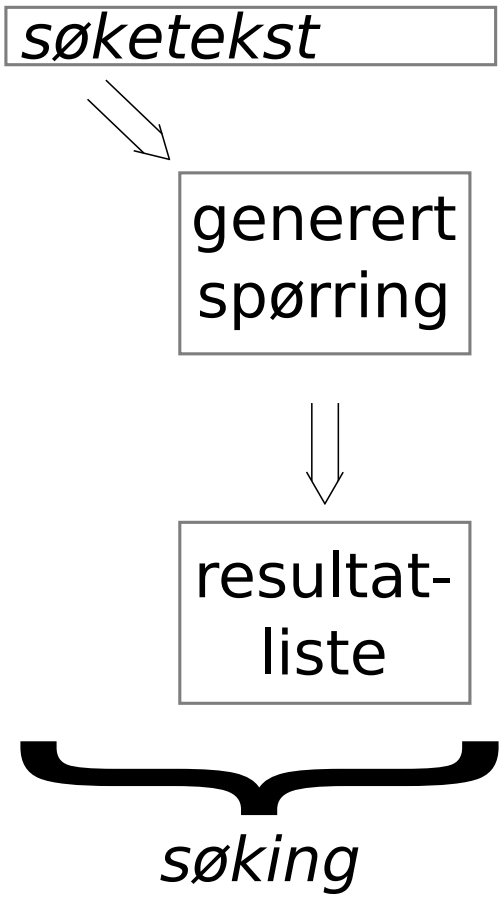
Noen vanlige grunner for dette er

- Skrivefeil
- Synonymitet

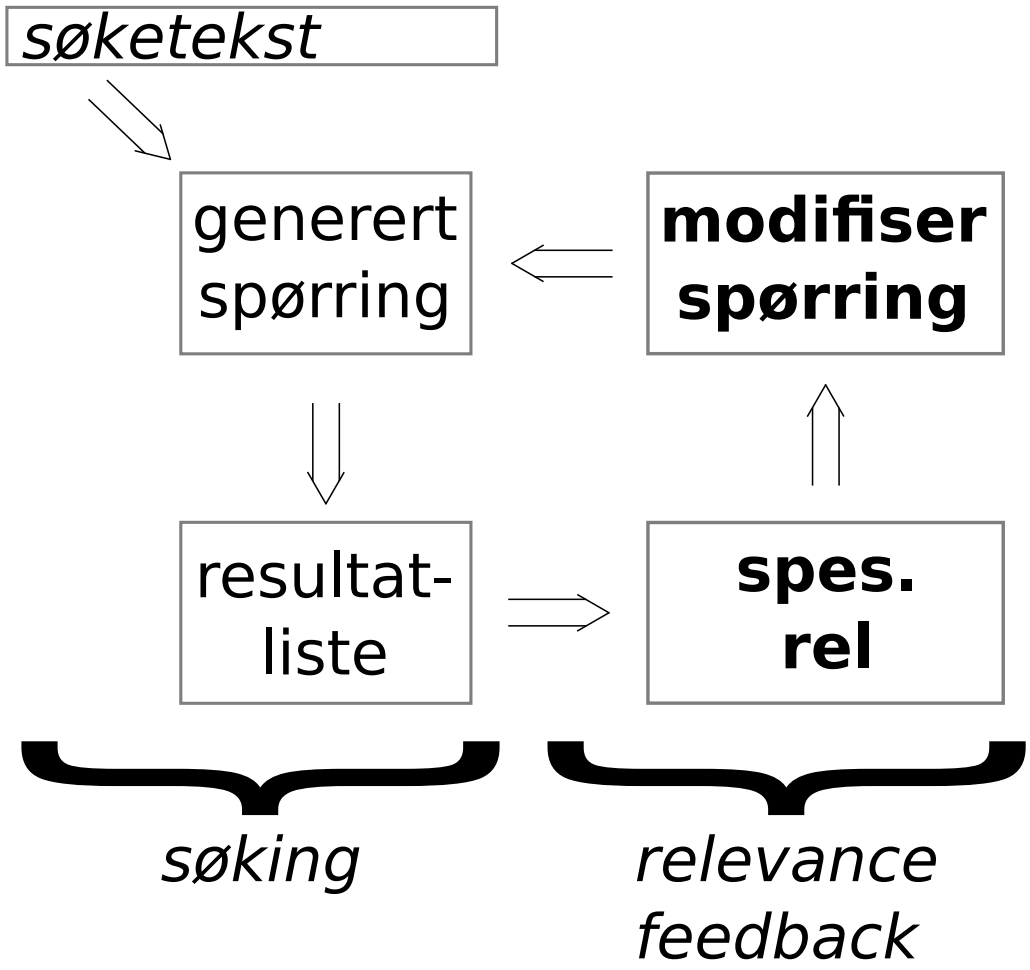
Vi skal se flere eksempler på løsningsforslag. Vi deler dem inn i

- *Globale metoder* er ikke avhengige av resultatet for spørringen, og
- *Lokale metoder* tar hensyn til spørringen.

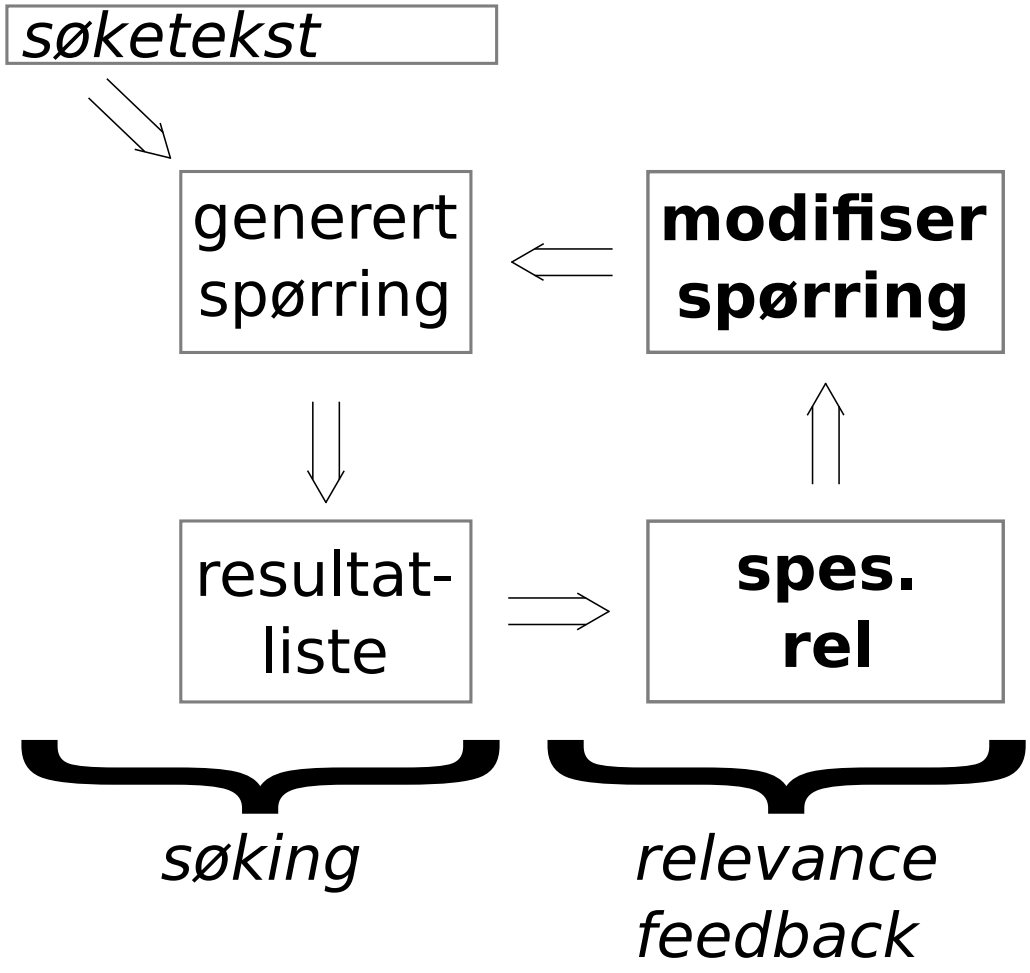
Relevance feedback (*RF*)



Relevance feedback (*RF*)



Relevance feedback (*RF*)



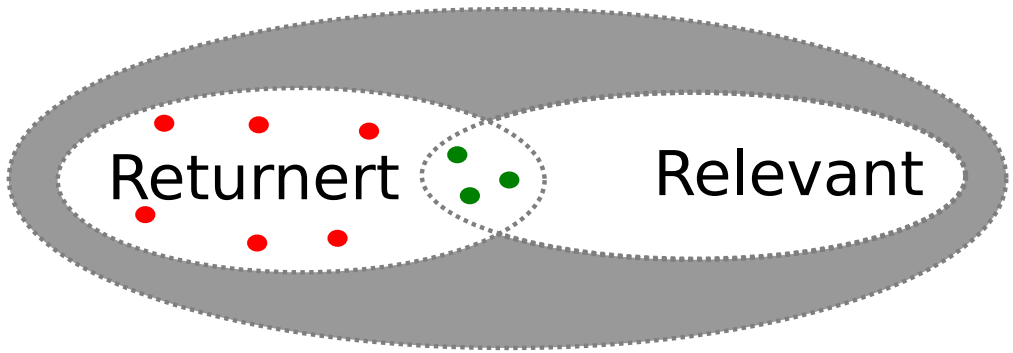
Dette har vist seg spesielt nyttig for bildesøk.

Relevance feedback (*RF*)

Vi har en spørring i form av en vektor, $\vec{q}_0$.

Brukeren får presentert en liste med resultater, C.

- Brukeren velger noen som relevante, C_r, og
- resten anser vi som irrelevante, C_{nr}.

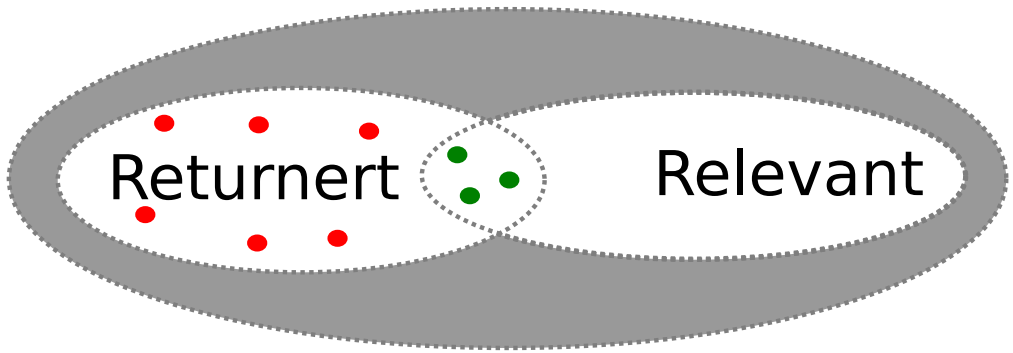


Relevance feedback (*RF*)

Vi har en spørring i form av en vektor, $\vec{q}_0$.

Brukeren får presentert en liste med resultater, C.

- Brukeren velger noen som relevante, C_r, og
- resten anser vi som irrelevante, C_{nr}.



Dette danner grunnlag for å avgjøre hvilken “retning” vi bør bevege oss.

RF - *Rocchio* algoritmen

Vi har $\vec{q}_0$, C_r og C_{nr}, og ønsker å finne $\vec{q}_{\text{opt}}$.

RF - *Rocchio* algoritmen

Vi har $\vec{q}_0$, C_r og C_{nr}, og ønsker å finne $\vec{q}_{\text{opt}}$.

$$\vec{q}_{\text{opt}} = \text{argmax}_{\vec{q}_0} \left\{ \text{sim}(\vec{q}_0, C_r) - \text{sim}(\vec{q}_0, C_{nr}) \right\}$$

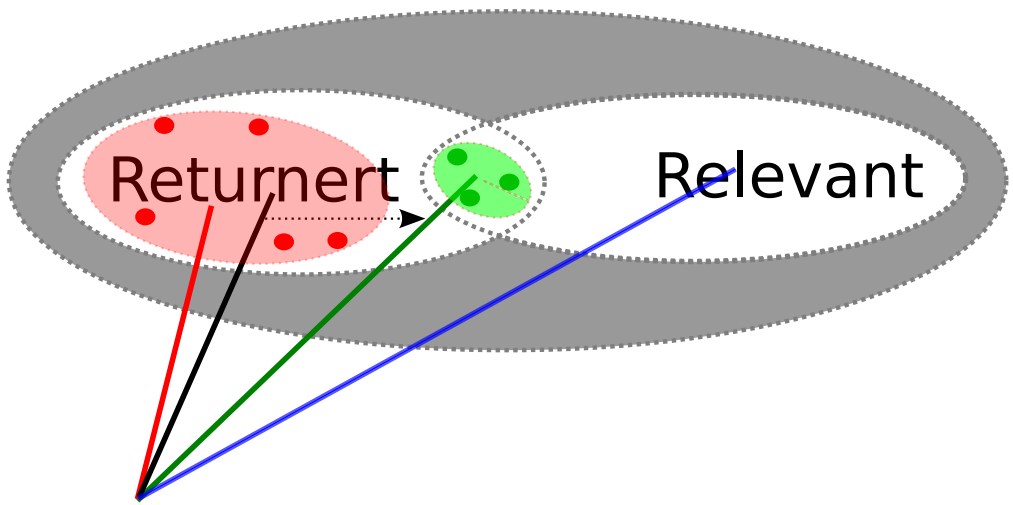
hvor $\text{sim}(\vec{q}_0, C)$ er funksjonen vi har sett tidligere. Mengden C er representert ved *tyngdepunktet* til dokumentene i C.

RF - *Rocchio* algoritmen

Vi har $\vec{q}_0$, C_r og C_{nr}, og ønsker å finne $\vec{q}_{\text{opt}}$.

$$\vec{q}_{\text{opt}} = \text{argmax}_{\vec{q}_0} \left\{ \text{sim}(\vec{q}_0, C_r) - \text{sim}(\vec{q}_0, C_{nr}) \right\}$$

hvor $\text{sim}(\vec{q}_0, C)$ er funksjonen vi har sett tidligere. Mengden C er representert ved *tyngdepunktet* til dokumentene i C.



RF - *Rocchio* algoritmen

Vi har $\vec{q}_0$, C_r og C_{nr}, og ønsker å finne $\vec{q}_{\text{opt}}$.

$$\vec{q}_{\text{opt}} = \text{argmax}_{\vec{q}_0} \left\{ \text{sim}(\vec{q}_0, C_r) - \text{sim}(\vec{q}_0, C_{nr}) \right\}$$

Vi kan finne denne $\vec{q}_{\text{opt}}$ relativt enkelt ved å løse

$$\vec{q}_{\text{opt}} = \underbrace{\frac{1}{|C_r|} \sum_{\vec{d}_j \in C_r} \vec{d}_j}_{\textit{tyngdepunkt}} - \frac{1}{|C_{nr}|} \sum_{\vec{d}_j \in C_{nr}} \vec{d}_j$$

RF - *Rocchio* algoritmen

Den ideelle løsningen er... ideell.

$$\vec{q}_{\text{opt}} = \alpha\vec{q}_0 + \beta\frac{1}{|D_r|}\sum_{\vec{d}_j \in D_r}\vec{d}_j - \gamma\frac{1}{|D_{nr}|}\sum_{\vec{d}_j \in D_{nr}}\vec{d}_j$$

- D_r er *kjente* relevante dokumenter (fra liste),

RF - *Rocchio* algoritmen

Den ideelle løsningen er... ideell.

$$\vec{q}_{\text{opt}} = \alpha\vec{q}_0 + \beta\frac{1}{|D_r|}\sum_{\vec{d}_j \in D_r}\vec{d}_j - \gamma\frac{1}{|D_{nr}|}\sum_{\vec{d}_j \in D_{nr}}\vec{d}_j$$

- D_r er *kjente* relevante dokumenter (fra liste),
- *negative* vekter blir satt til null,
- øker hovedsakelig *recall*,
- erfaring viser at positiv feedback er mest verdifull ($\beta > \gamma$),
- kanskje ignorerer vi negativ feedback ($\gamma = 0$) helt?
- *"Ide dec-hi"* bruker bare det "beste" negative for D_{nr} ($|D_{nr}| = 1$).

Relevance feedback generelt

RF metoder overkommer generelt ikke problemer som oppstår pga.

- Skrivefeil
  søketermen forekommer ikke, eller i feil dokumenter
- Flerspråklig dokumentsamling
  søking etter et ord på et språk finner ikke andre språk
- “Missforståelser”/synonymitet
  søker etter at ord som brukes lite(/ikke)

Relevance feedback generelt

RF/Rocchio antar at samlingen av relevante dokumenter utgjør en *klase*.
(Metoden forsøker å finne midten av denne.)

Relevance feedback generelt

RF/Rocchio antar at samlingen av relevante dokumenter utgjør en *klase*.
(Metoden forsøker å finne midten av denne.)

Dette er ikke alltid tilfellet.

- forskjellige deler av dokumentmengden kan bruke forskjellig vokabular,
- Noen søk er naturlig *disjunkte* (dekker mer enn en klase), *(f.eks. fra boken "pop stars who once worked at Burger King")*
- Svært generelle søketermer dekker naturlig mange klaser.

Relevance feedback generelt

Vi har generert en vektor $\vec{q}_{\text{opt}}$ som består av potensielt mange termer.

Dette fører til betraktelig mer beregning (i forhold til en søketekst på noen få termer).

Vi kan bare ta med “viktige” termer for å gjøre $\vec{q}_{\text{opt}}$ mindre, og, i følge noen undersøkelser, mer effektiv.

Relevance feedback generelt

Vi har generert en vektor $\vec{q}_{\text{opt}}$ som består av potensielt mange termer.

Dette fører til betraktelig mer beregning (i forhold til en søketekst på noen få termer).

Vi kan bare ta med "viktige" termer for å gjøre $\vec{q}_{\text{opt}}$ mindre, og, i følge noen undersøkelser, mer effektiv.

... og ikke minst:

Brukere orker ofte ikke en omfattende søkeprosess i flere steg.

Relevance feedback generelt

Søkemotorer på internett har stort sett gått bort fra RF.

Det finnes likevel *en* teknikk som ligner veldig på RF som vi har beskrevet det her, det er *“More like this”*-lenken som ofte vises ved siden av resultatene.

Brukeren har utpekt et dokument vi regner som eksemplarisk.

RF er fortsatt nyttig i andre, mer spesialiserte IR systemer.

Pseudo RF

En måte å anvende teorien vi nettopp har sett uten å bry brukeren med å velge ut relevante dokumenter:

Søk vanlig og *anta* at de beste k dokumentene er relevante.

Vi kan nå raffinere søket ved å finne $\vec{q}_{\text{opt}}$ som vi så tidligere.

Implisitt RF

Et klikk for å se et resultat kan tolkes som at dokumentet virker relevant.

Krever at resultatlisten viser nok informasjon for at brukeren kan vurdere relevans.

Denne informasjonen kan vi samle (f.eks. på tvers av brukermengden), og bruke for å vurdere relevans.

Implisitt RF

Et klikk for å se et resultat kan tolkes som at dokumentet virker relevant.

Krever at resultatlisten viser nok informasjon for at brukeren kan vurdere relevans.

Denne informasjonen kan vi samle (f.eks. på tvers av brukermengden), og bruke for å vurdere relevans.

Dette er *en* metode innen en strategi som generelt kalles *"clickstream mining"*.

Global vs. lokal

Metodene vi har sett har tatt hensyn til resultatene fra brukerens spørring.

Nå tar vi for oss globale metoder.

Felles for disse metodene er at de kan utføres før selve søkingen blir utført.

Assistert spørringskonstruksjon

IR systemet vårt kan assistere brukeren mens hun utformer en spørring.

Tiltak kan være bl. a.

- stavekontroll
- vise stoppord, stemming
- annen parseinformasjon
- forslag til synonymer (mest brukte synonym)
- om søketermen forekommer i termindeksen vår

Spørringsutvidelse

Spørringsutvidelse (*query expansion*) kan brukes for å forfine en spørring ved å

- foreslå videre innsnevring
  Eks (fra boken):
  Søketekst: “palm”
  Forslag: **palm** trees, **palm** springs, **palm** centro, **...**

- automatisk legge til synonymer
  Søketekst: “skin itch”
  Spørring:“ ’skin’ AND ’integumentary system’ AND ’itch’ AND ’pruritus’ “

Spørringsutvidelse

Hovedproblem: “hvordan finner vi nyttige forslag til forfininger?”

Tre fremgangsmåter er å

- normalisere søketeksten
  vi bruker et standard kontroll vokabular
  (vedlikeholdt av eksperter)

- bruke en synonymordbok
  Denne kan være manuelt eller automatisk konstruert

- foreslå forfininger basert på andre brukeres forfininger for samme/lignende spørring
  query log mining

Ferdig med pensum

Neste uke er forelesningsplanen enkel:

- oppsummering/repetisjon.

Send meg en mail **denne uken** om hva dere vil jeg skal ta for meg.