INF1820: Introduksjon til språk-og kommunikasjonsteknologi

Syvende forelesning

Lilja Øvrelid

6 mars, 2017

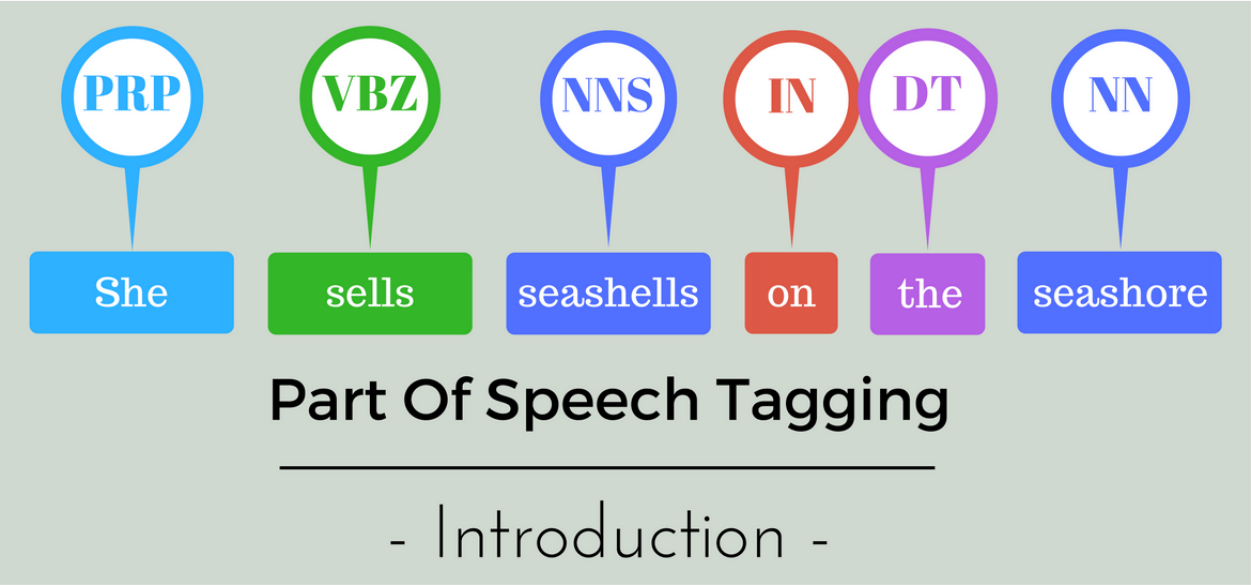
Ordklassetagging

Ordklasser?

- Bindeledd mellom ordet og setningen (syntaks):
  - Sier noe om hva slags kontekster et ord forekommer i
  - Sier noe om uttale (*record*, *content*, *discount*)
- Helt essensiell i en rekke språkteknologiske applikasjoner:
  - Talesyntese
  - Morfologisk analyse
  - "Chunking", syntaktisk parsing
  - Word Sense Disambiguation
  - Informasjonsekstraksjon

Ordklassetagging

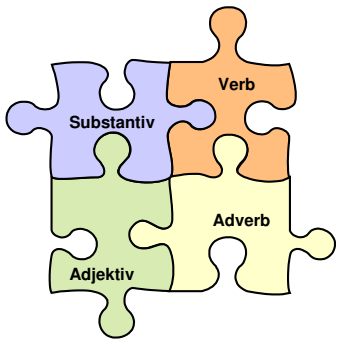
- Input: streng av ord og en spesifisert mengde tagger (taggsett)
- Output: en tagg per ord



Flertydighet

- De fleste engelske ord er entydige
- Men mange av de mest frekvente ordene er flertydige
- Tall fra Brown-korpuset:
  - 11.5% av engelske ordtyper er flertydige
  - 40% av Brown tokens er flertydige
- Heldigvis er ikke alle lesninger like sannsynlige
- Flertydighet:
  - Book/VB that/DT flight/NN ./.
  - Does/VBZ that/DT flight/NN serve/VB dinner/NN ?/.

Ordklasser



- **Taksonomi** - uttømmende, gjensidig utelukkende, styrt av et prinsipp
- Alle ord havner i en klasse og ingen ord havner i mer enn én klasse

Ordklassekriterier

3 slags kriterier:

1. formelle eller morfologiske kriterier
2. funksjonelle eller syntaktiske kriterier
3. betydningsmessige eller semantiske kriterier

Ordklassetagging

Mulige innfallsvinkler til oppgaven:

- **Default-taggeren**: tildeler alle ord samme tagg, den mest frekvente (NN)
  - ca. 13% korrekt taggede ord (Brown)

Ordklassetagging

Mulige innfallsvinkler til oppgaven:

- **Default-taggeren**: tildeler alle ord samme tagg, den mest frekvente (NN)
  - ca. 13% korrekt taggede ord (Brown)
- **Regulære uttrykk**: tildeler tagger basert på matchende uttrykk
  - `r'.*ing$', 'VBG'`
  - `r'.*ed$', 'VBD'`
  - `r'.*es$', 'VBZ'`
  - `r'.*ould$', 'MD'`
  - `r'.*\'s$', 'NN$'`
  - `r'.*s$', 'NNS'`
  - `r'^-?[0-9]+(.[0-9]+)?$', 'CD'`
  - `r'.*$', 'NN'`
  - `...`
  - ca 20% korrekt taggede ord (men rom for forbedring!)

Ordklassetagging

- **Oppslagstaggeren**: tildeler hvert ord dets mest frekvente ordklassetagg (basert på en unigrammodell)
  - dersom vi lagrer de 100 mest frekvente ordene og deres tagger får vi
  - ca. 46% korrekt taggede ord (Brown)
- Kan kombinere taggere (s.k. "back-off")

Algoritmer for ordklassetagging

- To hovedkategorier:
  1. **Regelbaserte taggere**: stor database med håndskrevne regler. Eksempel: *book* er substantiv, og ikke verb, dersom etterfølger en determinativ
     - Constraint Grammar-taggere
  2. **Probabilistiske taggere:** bruker et ordklassetagget korpus ("treningskorpus") til å beregne sannsynlighet for en gitt tagg i en gitt kontekst
     - Hidden Markov Models (HMM-taggere)

Regelbasert tagging

Regelbasert tagging

1. Preprosessering: tokenisering, setningssegmentering
2. Morfologisk analyse: hvert ord tildeles en liste av mulige tagger
3. Disambiguering: Håndskrevne regler **disambiguerer**

Regelbasert tagging

1. Preprosessering: tokenisering, setningssegmentering
2. Morfologisk analyse: hvert ord tildeles en liste av mulige tagger
3. Disambiguering: Håndskrevne regler **disambiguerer**

- Morfologisk analyse (transduser) mapper mellom ordform og og mulige trekk
- Constraint Grammar – sentral regelformalisme som har resultert i taggere for en rekke språk, deriblant engelsk og norsk

Morfologisk analyse

- Tildeler hvert ord alle mulige tagger
  - Fullformsleksikon
    - Alle ord: løp, løper, løpt, ...
    - Med tilhørende tagger
  - To-nivå morfologi: leksemleksikon + morfologisk analysator
    - Finsk: 2000 forskjellige former for substantiver, 12000 for verb!

Regelbasert disambiguering

- Håndskrevne regler (som regel flere tusen)
  - Eksempel: fjerner alle lesninger av *that* untatt ADV-lesningen, feks *It isn't* **that** *strange*

```
Given input: "that"
if
  (+1 A/ADV/QUANT)  ;if next word is adj/adverb/quant
  (+ 2 SENT-LIM)    ;followed by sent limit
  (NOT -1 SVOC/A)   ;previous word is not a verb
                    ;like 'consider'

then eliminate non-ADV tags
else eliminate ADV
```

Regelbasert tagging

- Norsk: Oslo-Bergen taggeren
- Test taggeren i LAP: `http://www.mn.uio.no/ifi/english/research/projects/clarino/`

```
"<som>"  SELECT:3261 (prep) IF
    (1 pron-akk)
    (NOT 1 pron-nom)
;
#  "Ei jente som (prep) meg"))
```

HMM-tagging

Probabilistisk tagging

- **Probabilistiske taggere:** bruker et ordklassetagget korpus ("treningskorpus") til å beregne sannsynlighet for en gitt tagg i en gitt kontekst
  - Hidden Markov Models (HMM-taggere)

Sannsynlighet (repetisjon)

- **Distribusjon**: mengde (ikke-negative) tall som summerer til 1.
  - Eksempel: terningkast
  - Seks resultater, antar typisk at disse er like sannsynlige: $1/6$ (**uniform** distribusjon)
- **Utfallsrommet ("sample space")**: mengden Ω av mulige utfall, hvert utfall er assosiert med en sannsynlighet, summerer til 1.
  - $\Omega = \{1, 2, 3, 4, 5, 6\}$

Sannsynlighet (repetisjon)

- **Hendelser (A, B, C, ...)** ("events") er delmengder av denne mengden
- En sannsynlighetsfunksjon P er en funksjon fra hendelser til intervallet $[0, 1]$
- $P(A)$ gir sannsynligheten for hendelsen A
  - Med terning: $P(\{2\}) = \frac{1}{6}$, $P(\{2, 4, 6\}) = \frac{1}{2}$

Sannsynlighet (repetisjon)

- **Felles** sannsynlighet ("joint probability") for to hendelser A og B er sannsynligheten for at begge hendelser finner sted, $P(A \cap B)$
  - kaster to mynter, første er kron, andre er mynt

Sannsynlighet (repetisjon)

- **Felles** sannsynlighet ("joint probability") for to hendelser A og B er sannsynligheten for at begge hendelser finner sted, $P(A \cap B)$
  - kaster to mynter, første er kron, andre er mynt
- **Uavhengighet**: sannsynligheten for en påvirker ikke sannsynligheten for den andre. To hendelser er uavhengige hvis

$$P(A \text{ og } B) = P(A)P(B)$$

(Multiplikasjonsregelen)

Sannsynlighet (repetisjon)

- **Betinget** sannsynlighet ("conditional probability")
- Lar oss håndtere **avhengige** hendelser

Sannsynlighet (repetisjon)

- **Betinget** sannsynlighet ("conditional probability")
- Lar oss håndtere **avhengige** hendelser
- Multiplikasjonsregelen:

$$P(A \cap B) = P(A)P(B|A) = P(B)P(A|B)$$

Bayes regel

Bayes regel

- Betinget sannsynlighet

$$P(A|B) = \frac{P(A \cap B)}{P(B)}$$

Bayes regel

- Betinget sannsynlighet

$$P(A|B) = \frac{P(A \cap B)}{P(B)}$$

- Multiplikasjonsregelen:

$$P(A \cap B) = P(A|B)P(B)$$

$$P(A|B)P(B) = P(B|A)P(A)$$

Bayes regel

- Betinget sannsynlighet

$$P(A|B) = \frac{P(A \cap B)}{P(B)}$$

- Multiplikasjonsregelen:

$$P(A \cap B) = P(A|B)P(B)$$

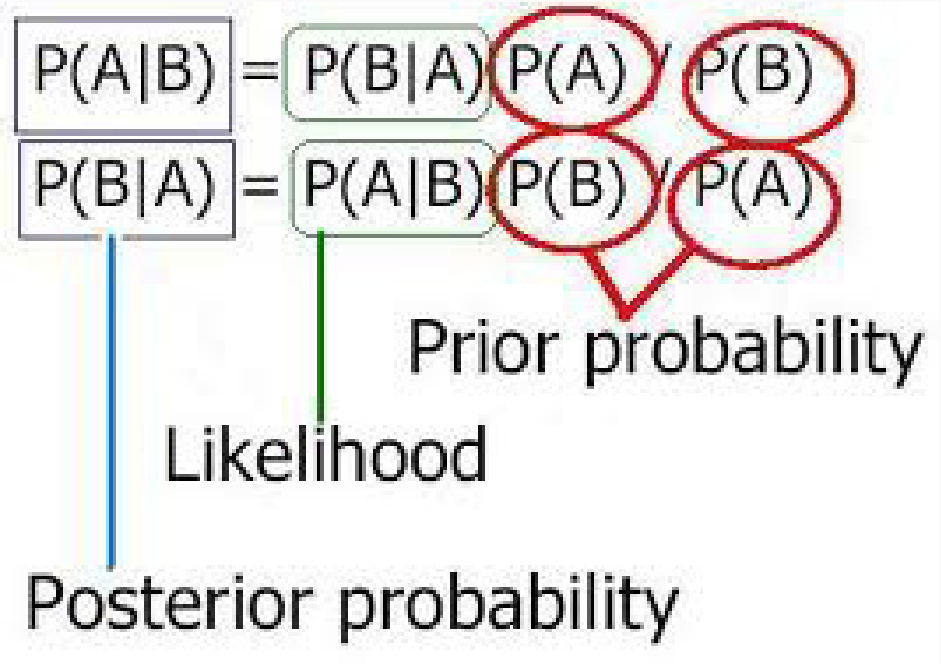
$$P(A|B)P(B) = P(B|A)P(A)$$

- Bayes regel

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

Bayes regel

- Bayes regel viser hvordan vi kan beregne inverse sannsynligheter, dvs dersom vi vet $P(A|B)$, hvordan kan vi beregne $P(B|A)$?
- Noen begreper



Bayes regel

- Eksempel fra et sykehus
  - 10% av pasientene har en leversykdom
  - 5% av pasientene er alkoholiker
  - Av pasientene med leversykdom er 7% alkoholikere
  - Hva er sannsynligheten for at en pasient som er alkoholiker har en leversykdom?

Bayes regel

- Eksempel fra et sykehus
  - 10% av pasientene har en leversykdom
  - 5% av pasientene er alkoholiker
  - Av pasientene med leversykdom er 7% alkoholikere
  - Hva er sannsynligheten for at en pasient som er alkoholiker har en leversykdom?
- Løsning med Bayes regel
  - Pasient har leversykdom $P(A) = 0.10$
  - Pasient er alkoholiker $P(B) = 0.05$
  - Pasient som har leversykdom er alkoholiker $P(B|A) = 0.07$

Bayes regel

- Eksempel fra et sykehus
  - 10% av pasientene har en leversykdom
  - 5% av pasientene er alkoholiker
  - Av pasientene med leversykdom er 7% alkoholikere
  - Hva er sannsynligheten for at en pasient som er alkoholiker har en leversykdom?
- Løsning med Bayes regel
  - Pasient har leversykdom $P(A) = 0.10$
  - Pasient er alkoholiker $P(B) = 0.05$
  - Pasient som har leversykdom er alkoholiker $P(B|A) = 0.07$

$$P(A|B) = \frac{P(B|A)(P(A)}{P(B)} = \frac{0.07 \times 0.10}{0.05} = 0.14$$

Notasjon

- Argmax-notasjon:
  - argmax $f(x)$ verdien av x som maksimerer $f(x)$
  - $x_0 = \underset{x}{\text{argmax}}\ f(x)$ betyr at $f(y) \leq f(x_0)$ for alle y

HMM-tagging

- Tagging som klassifiseringsoppgave:
  - Gitt en sekvens med ord
  - Hva er den mest sannsynlige taggsekvensen?
- Utfallsrom: alle mulige taggsekvenser t_1^n
- Betinget på de observerte ordene w_1^n
- Den taggsekvensen som gir oss høyest sannsynlighet $P(t_1^n|w_1^n)$
- $\hat{t}_1^n = \underset{t_1^n}{\text{argmax}}\ P(t_1^n|w_1^n)$

HMM-tagging

- Men hvordan beregner vi $P(t_1^n|w_1^n)$?

HMM-tagging

- Men hvordan beregner vi $P(t_1^n|w_1^n)$?
- Vi bruker Bayes regel til å transformere denne formelen til sannsynligheter det er lettere å beregne
- Bayes regel lar oss bryte ned en betinget sannsynlighet til tre andre sannsynligheter (likelihood, prior og prior):

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

HMM-tagging

- Men hvordan beregner vi $P(t_1^n|w_1^n)$?
- Vi bruker Bayes regel til å transformere denne formelen til sannsynligheter det er lettere å beregne
- Bayes regel lar oss bryte ned en betinget sannsynlighet til tre andre sannsynligheter (likelihood, prior og prior):

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

- Da får vi

$$\hat{t}_1^n = \underset{t_1^n}{\text{argmax}} \frac{P(w_1^n|t_1^n)P(t_1^n)}{P(w_1^n)} = \underset{t_1^n}{\text{argmax}} P(w_1^n|t_1^n)P(t_1^n)$$

HMM-tagging

- Beregner altså to sannsynligheter:
  1. $P(t_1^n)$ – prior-sannsynligheten for taggsekvensen
  2. $P(w_1^n|t_1^n)$ – likelihood for ordsekvensen
- Fremdeles vanskelig å beregne

HMM-tagging

- HMM gjør 2 forenklende antagelser:
  1. sannsynligheten for et ord avhenger bare av ordets egen tagg, dvs uavhengig av de andre ordene og taggene rundt:

$$P(w_1^n|t_1^n) = \prod_{i=1}^{n} P(w_i|t_i)$$

  2. sannsynligheten for en tagg er kun betinget av foregående tagg (bigram/Markov-antagelsen)

$$P(t_1^n) \approx \prod_{i=1}^{n} P(t_i|t_{i-1})$$

HMM-tagging

- Disse setter vi inn i den tidligere formelen:

$$\hat{t}_1^n = \underset{t_1^n}{\text{argmax}} \prod_{i=1}^{n} P(w_i|t_i)P(t_i|t_{i-1})$$

- Med denne kan vi beregne en taggsekvens for en gitt ordsekvens

HMM-tagging

To slags sannsynligheter:

1. "tag transitions": sannsynligheten for en tagg gitt den foregående taggen
   - Feks $P(NN|DT)$ vs $P(DT|NN)$

   Disse kan vi beregne fra et korpus (MLE)

$$P(t_i|t_{i-1}) = \frac{C(t_{i-1}, t_i)}{C(t_{i-1})}$$

HMM-tagging

To slags sannsynligheter (forts.):

2. "word likelihood": sannsynligheten for et ord, gitt en tagg
   - Feks *is* er sannsynlig gitt taggen VBZ

   Beregner fra et korpus

$$P(w_i|t_i) = \frac{C(t_i, w_i)}{C(t_i)}$$

HMM-tagging

- Et eksempel: *race* kan være både VB og NN
  (1) Secretariat/NNP is/BEZ expected/VBN to/TO **race/VB** tomorrow/NR
  (2) People/NNS continue/VB to/TO inquire/VB the/AT reason/NN for/IN the/AT **race/NN** for/IN outer/JJ space/NN
- La oss tagge ordet *race* i setning (1) over.

HMM-tagging

Antar at de tidligere ordene har blitt tagget:
Secretariat/NNP is/BEZ expected/VBN to/TO **race/??**
tomorrow

- P(race|**VB**) $\times$ P(VB|TO)
  - Gitt at vi har et verb, hvor sannsynlig er det at ordet er *race*?
  - Gitt at forrige tagg er TO, hvor sannsynlig er det at taggen er VB?
- P(race|**NN**) $\times$ P(NN|TO)
  - Gitt at vi har et substantiv, hvor sannsynlig er det at ordet er *race*?
  - Gitt at forrige tagg er TO, hvor sannsynlig er det at taggen er NN?

HMM-tagging

- Fra treningskorpuset har vi at:
  - $P(t_i|t_{i-1})$: P(VB|TO) = 0.34; P(NN|TO) = 0.021
  - $P(w_i|t_i)$: P(*race*|NN) = 0.00041; P(*race*|VB) = 0.00003

HMM-tagging

- Fra treningskorpuset har vi at:
  - $P(t_i|t_{i-1})$: P(VB|TO) = 0.34; P(NN|TO) = 0.021
  - $P(w_i|t_i)$: P(*race*|NN) = 0.00041; P(*race*|VB) = 0.00003
- Vi multipliserer og finner den mest sannsynlige tolkningen:
  - P(VB|TO) $\times$ P(*race*|VB) = .34 $\times$ .00003 = .00001
  - P(NN|TO) $\times$ P(*race*|NN) = .021 $\times$ .00041 = .000009
- VB er mest sannsynlig!

HMM-tagging

- Forresten er *race* NN i 98% av tilfellene:
  - P(VB|race) = 0.02
  - P(NN|race) = 0.98
- Hvordan finner vi sannsynlighetene?
  - treningskorpus med manuelt annotert tekst
  - tidkrevende og nøyaktig arbeid utført av lingvister

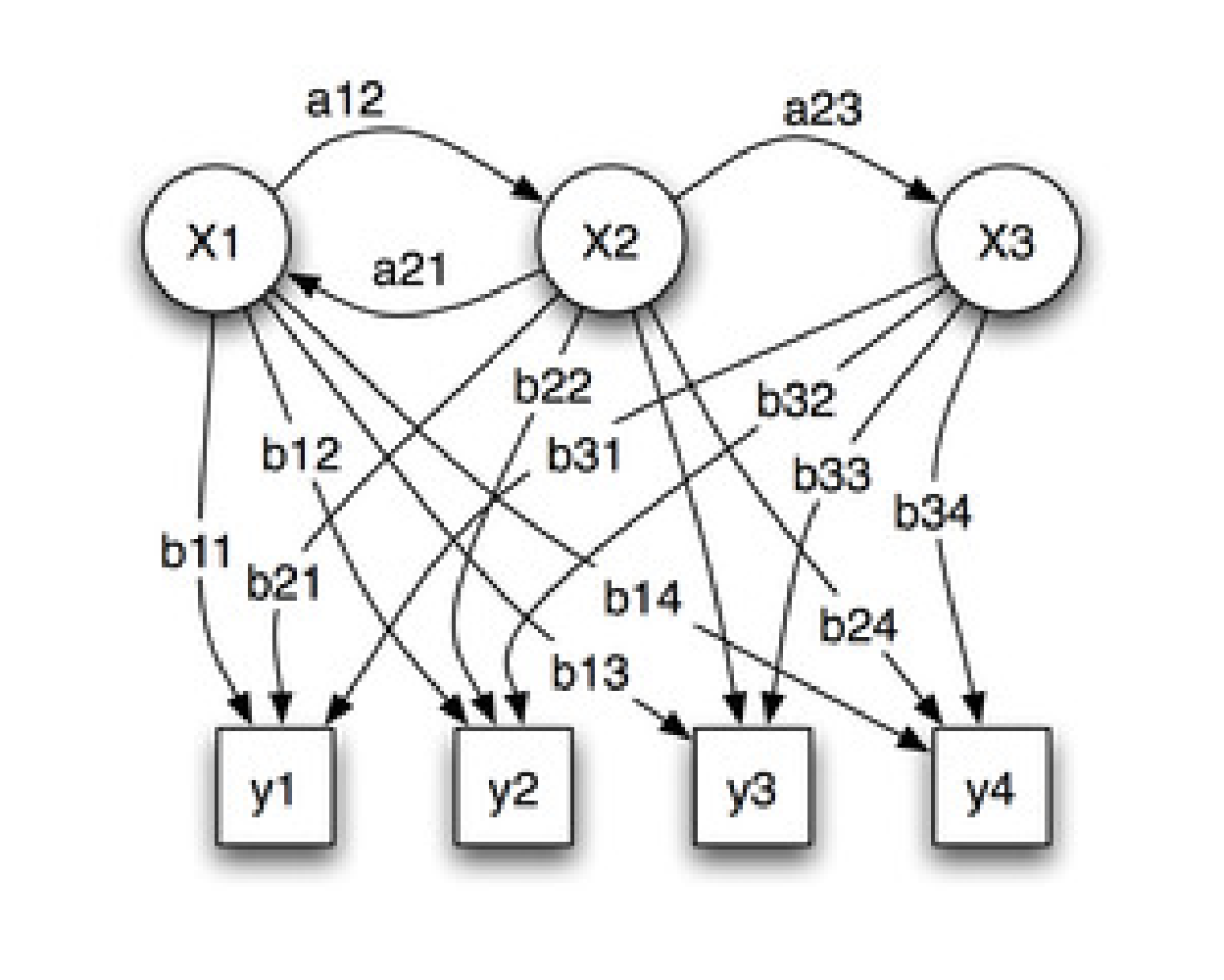
HMM-tagging

- HMM'er er en utvidelse av endelig tilstandsmaskiner (FSA'er)
- FSA'er defineres ved en mengde tilstander og en mengde transisjoner mellom disse i henhold til input-observasjoner
- En **vektet** FSA er en utvidelse av en FSA der hver transisjon er forbundet med en sannsynlighet: uttrykker hvor sannsynlig den overgangen er
- Sannsynlighetene for alle transisjoner fra en tilstand summerer til 1 (distribusjon)

HMM-tagging

- En HMM gir oss muligheten for å snakke om både observerte hendelser (ord) og skjulte hendelser (ordklasser)
- Definert ved:
  - $Q = \{q_0, q_1, q_2, \ldots, q_{n-1}\}$: en endelig mengde **tilstander**
  - $A = a_{11}a_{12}\ldots a_{n1}\ldots a_{nn}$: matrise over **transisjonssannsynligheter** A, der hver a_{ij} gir sannsynligheten for overgangen fra tilstand i til tilstand j, slik at $\sum_{j=1}^{n} a_{ij} = 1$
  - $O = o_1 o_2 \ldots o_T$: en sekvens av **observasjoner**
  - $B = b_i(o_t)$ **observasjonssannsynligheter** ("emission probabilities"), gir sannsynligheten for at en observasjon o_t blir generert fra tilstand i
  - q_0, q_F: en starttilstand, en sluttilstand

HMM-tagging



- $x =$ tilstander, $y =$ observasjoner, $a =$ transisjonssannsynligheter, $b =$ observasjonssannsynligheter

HMM-tagging

- To typer sannsynligheter
  1. transisjonssannsynligheter (A) – prior
  2. observasjonssannsynligheter (B) – likelihood

HMM-tagging

- Ordklasser (tilstander) genererer ord