

# INF1820: Introduksjon til språk-og kommunikasjonsteknologi

Sjette forelesning

---

Lilja Øvrelid

27 februar, 2017

1

## Sannsynlighet

- Sannsynlighet spiller en svært viktig rolle i språkteknologi
- ... og også i dette kurset!
- Språklig input inneholder mye “bråk” (noise), er flertydig og usegmentert
- Sannsynlighetsteori gir gode modeller for problemløsning og valg under usikre betingelser
- Bruker sannsynlighet for å kvantifisere graden av sikkerhet/usikkerhet vi innehar

2

- Gir oss mulighet til å modellere språklige sammenhenger:
  - Gitt kontekst  $Y$ , hva er sannsynligheten for at vi nettopp hørte ord  $X$  (og ikke ord  $Z$ )?
  - Gitt at vi har struktur  $X$  i input'en, hva er sannsynligheten for at vi også har struktur  $Y$ ?

- Terningkast-eksempler
  - Hva er sannsynligheten for å kaste en 1?
  - Hva er sannsynligheten for å kaste en 2?
  - Hva er sannsynligheten for å kaste en 3 eller 4?



- Men hva har dette med språklige fenomener å gjøre?
- Talegjenkjenning: hvilket ord ble nettopp ytret?
  - Vi har etablert at taleren snakker engelsk
  - Kan beregne sannsynligheten for at ordet er *the* og f.eks. *telephone*
  - Vi kan være ganske sikre på at det mest sannsynlige ordet er *the* (ca 5%)

## Litt terminologi

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

## Litt terminologi

- **Hendelser** ( $A, B, C, \dots$ ) (“events”) er delmengder av utfallsmengden
- Enhver **mengde** utfall har en sannsynlighet: summen av medlemmenes sannsynlighet
  - Sannsynligheten for å slå et partall: 0.5
- En sannsynlighetsfunksjon  $P$  er en funksjon fra hendelser til intervallet  $[0, 1]$
- $P(A)$  gir sannsynligheten for hendelsen  $A$ 
  - Med terning:  $P(\{2\}) = \frac{1}{6}$ ,  $P(\{2, 4, 6\}) = \frac{1}{2}$

7

## Aksiomer

- Sannsynlighetsteoriens 3 aksiomer:
  - $P(A) \geq 0$  for alle hendelser  $A$
  - $P(\Omega) = 1$
  - $P(A \cup B) = P(A) + P(B)$  (additivitet for disjunkte hendelser)

8

- Dette er grunnlaget for alle sannsynlighetsmodeller:
  - vi lager en modell over en mengde data
  - tildeler en distribusjon til mengden av mulige utfall
- Notasjon: sannsynligheten skal alltid summere til 1:

$$\sum_i p_i = 1.0$$

## Felles sannsynlighet

- **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)

- Vi begynner å nærme oss...
  - 1000-siders terning (Hva er sannsynligheten for 567?)
  - Velger ut de 1000 mest frekvente ordene fra Brown-korpuset
  - Ved hvert terningkast velger vi ordet med den rangeringen (1=*the*, 2=*of* osv.)
    - 320 990 646 94 756
    - whether designed passed must southern
  - En svært enkel ordgenerator

- Men ikke helt slik vi tenker oss modellering av språklige fenomener
- Hva er sannsynligheten for en setning noen faktisk har ytret?
  - *In the beginning was the word*
  - 6 ord:  $1/1000 * 1/1000 * 1/1000 * 1/1000 * 1/1000 * 1/1000$
- Men tildeler samme sannsynlighet til enhver sekvens av seks ord
- Bra eller dårlig grammatikk for engelsk?
  - Modellen vil gi sannsynlighet 0 til enhver sekvens med ord som ikke er blant de 1000 mest frekvente
  - + Men vil også gi sannsynlighet 0 til en setning som ikke er på engelsk

## Sannsynlighet og språk

- Et bedre forslag:
  - tildele hvert ord en sannsynlighet som tilsvarer dets frekvens i et korpus
    - F.eks. *the* forekommer 69903 ganger ut av 1159267 ord, gir en sannsynlighet på ca. 0.0603
- Utfallsrom ( $\Omega$ ) = mengden av engelske ord (funnet empirisk, dvs i korpuset)
- Tildeler dem en sannsynlighet som tilsvarer frekvens i korpuset
- Dette er en **unigram** ordmodell
- Sannsynlighetene summerer til 1.0

13

## Sannsynlighet og språk

Toppen av unigramdistribusjonen for Brown-korpuset

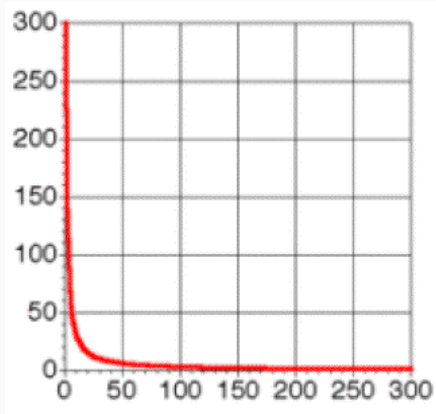
ord	telling	frekvens
the	69903	0.068271
of	36341	0.035493
and	28772	0.028100
to	26113	0.025503
a	23309	0.022765
in	21304	0.020807
that	10780	0.010528
is	10100	0.009864
was	9814	0.009585
he	9799	0.009570
for	9472	0.009251
it	9082	0.008870
...	...	...

14

## Zipfs lov

Empirisk lov innenfor matematisk statistikk

- Ranger alle ordene etter frekvens
- Det andre ordet vil opptre ca halvparten så ofte som det første ordet, det tredje ordet en tredjedel så ofte som det første, osv.



<https://www.youtube.com/watch?v=fCn8zs9120E>

15

## Sannsynlighet og språk

- Hva er sannsynligheten for setningen *The woman arrived*?
- Fra unigrammodellen:  $P(\textit{the}) = 0.068271$ ,  $P(\textit{woman}) = 0.00023$ ,  $P(\textit{arrived}) = 0.00006$
- Sannsynligheten til vår 3-ords setning (S) blir da  $P(S) = P(\textit{the}) * P(\textit{woman}) * P(\textit{arrived})$
- Antar at sannsynligheten til et ord er uavhengig av posisjon

16



## Betinget sannsynlighet

- Noen ganger ønsker vi å begrense utfallsrommet
- Vi har mer informasjon som vi vil inkludere i modellen
  - Trekker et kort og får vite at kortet er rødt
  - Gjetter et ord og får vite at ordet er et substantiv
  - Gjetter et ord og får vite hva det foregående ordet er
- Skal gjette noe, men har informasjon som gjør at vi kan gjette bedre
- **Betinget** sannsynlighet (“conditional probability”)

17

## Betinget sannsynlighet

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

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

18

- **Kjederegelen** generaliserer multiplikasjonsregelen til flere hendelser:

$$P(A \cap B \cap C \cap D \cap E \dots) = P(A)P(B|A)P(C|A \cap B)P(D|A \cap B \cap C) \dots$$

- Språkmodeller ( $n$ -grammodeller): sannsynlighetsmodeller som angir sannsynligheten for neste ord gitt de  $n - 1$  foregående ordene
- Kan også brukes til regne ut sannsynligheten for en hel setning

- Brukes innenfor en rekke språkteknologiske områder:
  - talegjenkjenning
    - *det vil jeg gjerne* er mer sannsynlig enn *det vil jeg hjerne*
  - stavekontroll
    - *Their are problems wit this sentence*
  - maskinoversettelse
    - mer sannsynlige setninger er antagelig bedre oversettelser
- Sentral kobling til nye metoder for ML: “deep learning”

- Et  $n$ -gram er en sekvens av  $n$  ord (tokens)
  - 2-gram (bigram;  $n = 2$ ) er en sekvens av to ord, feks *Johaug blir, blir utestengt, utestengt i, i ett, ett år*
  - 3-gram (trigram;  $n = 3$ ) er en sekvens av tre ord, feks *Johaug blir utestengt, blir utestengt i, utestengt i ett, i ett år*
- En språkmodell ( $n$ -grammodell) er en modell som forutsier det siste ordet i et  $n$ -gram gitt de foregående ordene

## Notasjon

- Sum og produkt
  - $\sum_{i=1}^n a_i = a_1 + a_2 + \dots + a_n$
  - $\prod_{i=1}^n a_i = a_1 \cdot a_2 \cdot \dots \cdot a_n$
- Eksempler
  - $\sum_{i=1}^7 1 + 2 + 3 + 4 + 5 + 6 + 7 = 28$
  - $\prod_{i=1}^7 1 \cdot 2 \cdot 3 \cdot \dots \cdot 7 = 5040$

- Vi ønsker å finne sannsynligheten for et ord  $w$  gitt en historie  $h$ :  $P(w|h)$ , for eksempel:  
 $P(\text{år} \mid \text{Johaug blir utestengt i ett})$
- Hvordan kan vi beregne denne sannsynligheten?
- En mulighet er frekvens i et korpus (som i den tidligere unigram-modellen)

$$\frac{C(\text{Johaug, blir, utestengt, i, ett, år})}{C(\text{Johaug, blir, utestengt, i, ett})}$$

- Hvorfor er denne metoden problematisk?

24

## Kjederegelen

- Finne en smartere metode for å beregne sannsynligheten for  $w$  gitt  $h$ , eller en sekvens av ord ( $P(w_1, w_2, w_3, \dots)$ )
- Kan benytte oss av kjederegelen:

$$P(X_1 \dots X_n) = P(X_1)P(X_2|X_1)P(X_3|X_1^2) \dots P(X_n|X_1^{n-1})$$

$$= \prod_{k=1}^n P(X_k|X_1^{k-1})$$

- For sekvenser av ord:

$$\prod_{k=1}^n P(w_k|w_1^{k-1})$$

25

- Ideen bak  $n$ -grammodellen: i stedet for å beregne sannsynligheten for et ord gitt alle de foregående ordene, ser vi kun på noen få av de foregående ordene (avhengig av  $n$ ):
  - Bigrammodellen:  $P(w_n|w_{n-1})$
  - $P(\text{år} | \text{ett})$
- **Markov-antagelsen:** vi kan forutsi sannsynligheten til en framtidig hendelse ved å se på en begrenset historie

## Markovantagelsen

- Vi kan generalisere bigrammodellen til  $n$ -grammodellen, som ser på  $n - 1$  av de foregående ordene
- Med en bigrammodell blir sannsynligheten for en streng  $w_1, \dots, w_k$  da produktet av de individuelle ordenes sannsynlighet, slik:

$$P(w_1 \dots w_k) = \prod_{i=1}^k P(w_i | w_{i-1})$$

- Med en trigrammodell:

$$P(w_1 \dots w_k) = \prod_{i=1}^k P(w_i | w_{i-2} w_{i-1})$$

## Markovantagelsen

- Vi kan beregne disse sannsynlighetene som tidligere: tellinger fra et korpus (normalisert slik at ligger mellom 1 og 0)

$$P(w_n|w_{n-1}) = \frac{C(w_{n-1}w_n)}{\sum_w C(w_{n-1}w)} = \frac{C(w_{n-1}w_n)}{C(w_{n-1})}$$

$$P(\text{\AA}r|\text{ett}) = \frac{C(\text{ett}, \text{\AA}r)}{C(\text{ett})}$$

- Sannsynlighetene gis ved relativ frekvens i et korpus
- Dette kalles **Maximum Likelihood Estimation**

28

## Trening

- Hva vi teller som et ord avhenger av applikasjon
- Tokenisering
- Normalisering av bokstavering, forkortelser, numeriske uttrykk, tegnsetting, etc.
- Lemmatisering. Baseformer vs fullformer

29

Et eksempel (J&M)

- Minikorpus:

<s> I am Sam </s>

<s> Sam I am </s>

<s> I do not like green eggs and ham </s>

- Hvilke bigrammer har vi i korpuset?
- Beregner bigramsannsynlighetene:

$$P(w_n|w_{n-1}) = \frac{C(w_{n-1}w_n)}{C(w_{n-1})}$$

Et eksempel (J&M)

- Minikorpus:

<s> I am Sam </s>

<s> Sam I am </s>

<s> I do not like green eggs and ham </s>

- Bigramsannsynligheter:

$$P(I|<s>) = 2/3 = .67$$

$$P(\text{Sam}|<s>) = 1/3 = .33$$

$$P(\text{am}|I) = 2/3 = .67$$

$$P(</s>|\text{Sam}) = 1/2 = 0.5$$

$$P(\text{Sam}|\text{am}) = 1/2 = 0.5$$

$$P(\text{do}|I) = 1/3 = 0.33$$



- Med disse sannsynlighetene kan vi beregne sannsynligheten for en setning

$$P(w_1 \dots w_k) = \prod_{i=1}^k P(w_i | w_{i-1})$$

Eksempel:

$$\begin{aligned} P(\langle s \rangle I am Sam \langle /s \rangle) \\ &= P(I | \langle s \rangle) P(am | I) P(Sam | am) P(\langle /s \rangle | Sam) \\ &= .67 \times .67 \times .5 \times .5 \\ &= 0.112225 \end{aligned}$$

## Trening og testing

- Sannsynlighetene i en språkmodell kommer fra korpuset den ble **trenet** på
- Vi bruker treningssettet til å beregne modellen og testsettet til å evaluere modellen
- Appliseres deretter på nytt datasett: **testdata**
- Evaluering:
  - **Intrinsisk** evaluering: modell trenes på treningssettet og testes på testsettet
  - viktig at disse ikke overlapper!
  - **Ekstrinsisk** evaluering: applikasjonsbasert, end-to-end
  - måler effekten av bruk i en applikasjon ved resultatene for applikasjonen (feks maskinoversettelsessystemet eller talegjenkjenneren som LM er del av)

- Dersom vi tester på treningsdataene får vi urealistisk inntrykk av resultater for en virkelig applikasjon
- Vil gi et drastisk fall ved applikasjon på usette datasett
- “Overfitting”: modellene inneholder mye korpusspesifikk informasjon
  - treningskorpus og testkorpus hentet fra samme sjanger
  - balansert korpus

### Problemer

- “Data sparseness”
  - uavhengig av korpsets størrelse, vil det alltid være fraser som mangler
  - språkets kreativitet
- Ukjente ord
- Ord med frekvens 0 gir oss problemer. Hvorfor?

- Mulige løsninger
  - Utjevning (“smoothing”): passe på at alle  $n$ -grammer får frekvens større enn 0
  - Metoder for å inkludere ukjente ord i modellen

- Ta noe av sannsynlighetsmassen fra frekvente hendelser og gi til usette hendelser
- Enkleste metoden: Add-one (Laplace) smoothing

$$P(w_n|w_{n-1}) = \frac{C(w_{n-1}w_n) + 1}{C(w_{n-1}) + V}$$

der  $V$  er antall ordtyper

- Andre metoder:
  - Back-off
  - Klasse-baserte modeller