SMT—Statistisk maskinoversettelse

INF5820 – H2008

Jan Tore Lønning

Institutt for Informatikk
Universitetet i Oslo

2. oktober

Outline

Outline

Støyete kanal

The noisy channel model

- Det vi oppfatter er en forvanskning av det originale budskapet.
- Gjett hva som egentlig er ment.
- J&M Fig. 5.23, 9.2 og 25.15

Example

- Talegjenkjenning: Det vi hører er en forvanskning av det vi skriver.
- Tagging: Det vi leser er en forvanskning av en tagg-sekvens.
- Oversettelse: Det vi hører/leser i kildespråket er en forvanskning av målspråket

Støyete MT

- Vi er interessert i å finne den beste engelske oversettelsen $\hat{E}$ av en norsk (fremmed) setning F.
- $\hat{E} = \arg\max_E P(E \mid F)$
- For å gjøre det trenger vi:
  1. En modell for dette. Vi har antagelig ikke sett F før. Vi må prøve å finne en approksimasjon.
  2. Vi må lære parametrene i modellen.
  3. Vi må ha en metode for å bruke modellen, for å finne den beste kandidaten, dekoding
- Disse 3 trinnene finner vi igjen i flere former for statistisk språkbehandling.

Vi bruker Bayes

- $$\begin{aligned}\hat{E} &= \arg\max_E P(E \mid F) \\ &= \arg\max_E \frac{P(F \mid E)}{P(F)} P(E) \\ &= \arg\max_E P(F \mid E) P(E)\end{aligned}$$
- Hvorfor? Er $P(E \mid F)$ enklere enn $P(F \mid E)$?
- Utgangspunkt for forenkling/approksimasjoner.
- Tilgang til $P(E)$ = språkmodell.

Outline

Stokastisk tagging

- Gitt en sekvens av ord $w_1 w_2 \ldots w_n$ (som vi vil skrive w_1^n)
- Hva er den mest sannsynlige taggsekvensen $t_1^n = t_1 t_2 \ldots t_n$?
-

$$\begin{aligned} \arg\max_{t_1^n} P(t_1^n \mid w_1^n) &= \arg\max_{t_1^n} \frac{P(w_1^n \mid t_1^n) P(t_1^n)}{P(w_1^n)} \\ &= \arg\max_{t_1^n} P(w_1^n \mid t_1^n) P(t_1^n) \end{aligned}$$

Ordsannsynlighetene

- Uttrykket vi ser på: $P(w_1^n \mid t_1^n)P(t_1^n)$
- $P(w_1^n \mid t_1^n) =$
  $P(w_n \mid w_1^{n-1} t_1^n)P(w_{n-1} \mid w_1^{n-2} t_1^n) \cdots P(w_1 \mid t_1^n)$
- Antar $P(w_i \mid w_{1,i} t_1^n) = P(w_i \mid t_i)$
- Altså (feilaktig) at et ord ikke avhenger av ordene før og etter, bare av sin tagg.
- $P(w_1^n \mid t_1^n) =$
  $P(w_n \mid t_n)P(w_{n-1} \mid t_{(n-1)}) \cdots P(w_1 \mid t_1) = \prod_{i=1}^{n} P(w_i \mid t_i)$

Tagg-sannsynlighetene

- Antar Markov-egenskapen $P(t_{(j+1)} \mid t_1^j) = P(t_{j+1} \mid t_j)$
- $$\begin{aligned} P(t_1^n) &= P(t_1)P(t_2 \mid t_1)P(t_3 \mid t_1^2)\cdots P(t_n \mid t_1^{(n-1)}) \\ &= P(t_1)P(t_2 \mid t_1)P(t_3 \mid t_2)\cdots P(t_n \mid t_{n-1}) \\ &= P(t_1)\prod_{i=1}^{n-1} P(t_{i+1} \mid t_i) \\ &= \prod_{i=1}^{n} P(t_i \mid t_{i-1}) \end{aligned}$$
- (Hvis $P(t_1 \mid t_0)$ er sannsynligheten for å starte i t_1)

Som gir

- $$\begin{aligned} P(w_1^n \mid t_1^n)P(t_1^n) &= \prod_{i=1}^{n} P(w_i \mid t_i) \prod_{i=1}^{n} P(t_i \mid t_{i-1}) \\ &= \prod_{i=1}^{n} P(w_i \mid t_i)P(t_i \mid t_{i-1}) \end{aligned}$$
- $$\begin{aligned} \hat{t}_1^n &= \arg\max_{t_1^n} P(t_1^n \mid w_1^n) \\ &= \arg\max_{t_1^n} P(w_1^n \mid t_1^n)P(t_1^n) \\ &= \arg\max_{t_1^n} \prod_{i=1}^{n} P(w_i \mid t_i)P(t_i \mid t_{i-1}) \end{aligned}$$

Outline

Oversettelse som tagging

- Anta at
  - Kildesetning og målsetning er like lange.
  - Det er en ord-til-ord korrespondanse.
  - Ordrekkefølgen bevares.
- Da blir oversettelse som tagging, med

Oversettelse	Tagging
ord i kildespråket	ord
ord i målspråket	tagg
n-grammodell for målspråket	taggsannsynlighet
kildespråkessetning	setning som skal tagges
sannsyn. for ordoversettelse	sannsyn. for ord gitt tagg

- Se på eksempel

Anvendt på oversettelse

Norsk: Han gikk til en bredd.
Engelsk1: He went to a bank.
Engelsk2: He went to a brim.

3-gram
Må se på
P(* * he)
P(* he went)
P(he went to)
P(went to a)
P(to a bank)/P(to a brim)
P(a bank .)/ P(a brim .)
P(bank . *)/ P(brim . *)
P(. * *)

Anta
P(to the bank) = 2*P(to the brim)
P(He went to a brim.)=b, vi kaller denne for *b*.
De andre er like.

$$\arg\max_{\mathbf{t}} P(\mathbf{t} \mid \mathbf{w}) = \arg\max_{\mathbf{t}} P(Hgteb \mid t_1 ... t_n) P(t_1 ... t_n)$$

$$\arg\max_{\mathbf{t}} P(\mathbf{t} \mid \mathbf{w}) = \arg\max_{\mathbf{t}} P(h \mid t_1) P(g \mid t_2) P(t \mid t_3) P(e \mid t_4) P(b \mid t_5) P(t_1 ... t_5)$$

$$P(han \mid he) P(gikk \mid went) P(til \mid to) P(en \mid a) = a$$

$$P(Hewenttoabank \mid \mathbf{w}) = (a * P(bredd \mid bank) * 2b = a * 0{,}02 * 2b = 0{,}04ab)$$
$$P(Hewenttoabrim \mid \mathbf{w}) = (a * P(bredd \mid brim) * b = a * 0{,}2 * b = 0{,}2ab)$$

Oversettelsen
C(bredd → bank) = 1000
C(bredd → brink) = 1000
C(bredd) = 5000
C(brim) = 5000
C(bank) = 50 000
P(bredd | bank) = 0,02
P(bredd | brim) = 0,2
P(bank | bredd) = 0,2
P(brim | bredd) = 0,2

Oversettelse er ikke tagging

- Ikke en-til-en korrespondanse mellom tokens i samme rekkefølge:
  - Et ord i et språk kan bli til flere i et annet: *bilen* $\rightarrow$ *the car*
  - Flere ord kan bli ett *the car* $\rightarrow$ *bilen*.
  - Rekkefølgen kan endres: *Jeg har sett ham* $\rightarrow$ *Ich habe ihn gesehen*.

Modeller for oversettelse

Utgangspunkt:

$$\arg\max_E P(F \mid E)P(E)$$

1. For språkmodellen $P(E) = P(e_1 e_2 \ldots e_n)$ kan vi bruke n-gram som ved tagging.
2. Men for $P(F|E)$ kan vi ikke bare gå ord for ord.

- Flere alternative modeller for $P(F|E)$.
- Problemstilling deler seg gjerne i to
  a) Ord-for-ord-delen. Sannsynligheten for at et ord er oversettelse av et annet.
  b) Plassering, "alignment" mellom et ord og dets oversettelse

Deler

Trinnene

- Modell
- Parameterlæring
- Dekoding

- Modell, to deler:
  1. Språkmodell, kan bruke *n*-gram
  2. Oversettelsesmodell: gjenstår
- Læring, kan gjøres uavhengig for de to
  1. Språkmodell, læring av *n*-grammodell: kjent
  2. Oversettelsesmodell: gjenstår
- Dekoding: må gjøres samlet, gjenstår.

Diverse oversettelsesmodeller

1. Ordbaserte
   1. IBM-modellene: 1, 2, 3, 4, 5
      - IBM-modell 1
      - IBM-modell 3
   2. HMM
2. Frasebaserte

Videre

- Knight & Koehn, 2003: 11–24.
- IBM-model 1 og litt model 3.
- EM-algoritmen

Outline

Alignment

Hvordan representerer vi oversettelsesdata for en ord-basert modell?

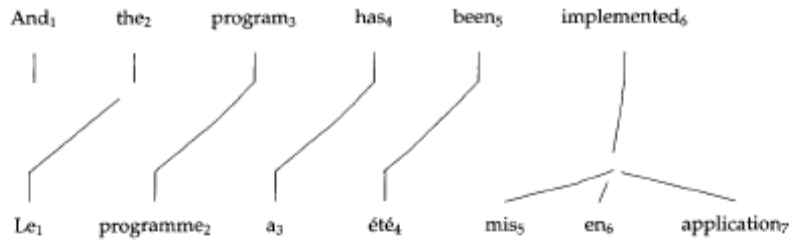


Figure 1
An alignment with independent English words.

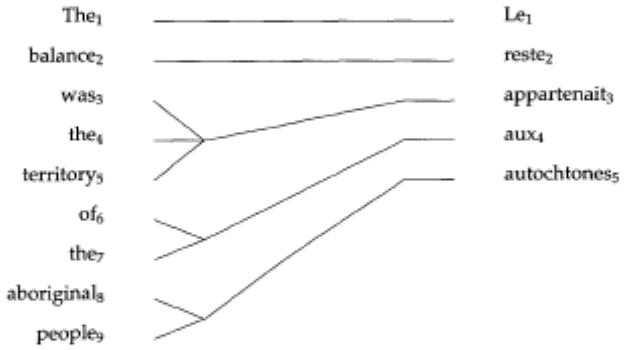
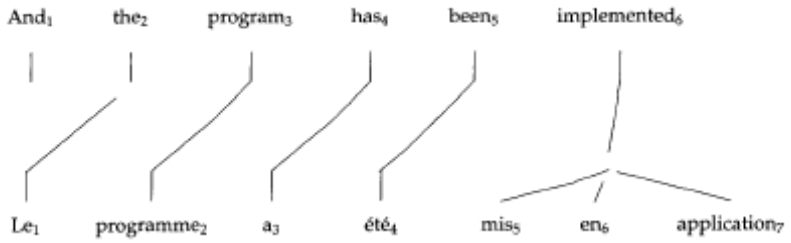


Figure 2
An alignment with independent French words.

Figure 3
A general alignment.

Representasjon



Denne alignmenten:

<2, 3, 4, 5, 6, 6, 6>

<a1,a2,a3,a4,a5,a6>

Generelt:

- Lengden av engelsk streng: k
- Lengden av fransk streng: m
- En alignment er en vektor av m tall, hvert mellom 0 og k. (Hvorfor 0?)
- (k+1)^m mange forskjellige

OBS:

- I prinsippet kan flere franske ord stamme fra samme engelske
- Men hvert fransk ord stammer enten fra et engelsk eller fra ingenting

IBM-modellene

IBM-modell 1 og 2

$$\Pr(\mathbf{f} \mid \mathbf{e}) = \sum_{\mathbf{a}} \Pr(\mathbf{f}, \mathbf{a} \mid \mathbf{e})$$

$$\Pr(\mathbf{f}, \mathbf{a} \mid \mathbf{e}) = \Pr(m \mid \mathbf{e}) \prod_{j=1}^{m} \Pr(a_j \mid a_1^{j-1}, f_1^{j-1}, m, \mathbf{e}) \Pr(f_j \mid a_1^{j}, f_1^{j-1}, m, \mathbf{e})$$

Betyr:

- Velg lengde av den franske strengen gitt den engelske.
- På plass 1 i den franske strengen, velg hvilken plass i den engelske vi skal se på gitt den engelske og lengden av den franske.
- Velg deretter det franske ordet på grunnlag av denne etablerte forbindelsen og de samme opplysningene

…..

- Se på neste plass i den franske strengen. Velg plass i den engelske strengen på grunnlag av den engelske, lengden av den franske og alle fobindleser og ord som er valgt så langt.
- Ta også dette med i betraktning og velg hvilket fransk ord som skal stå her.

Så langt ikke en tilnærming.

IBM-modell 1

$$\Pr(\mathbf{f},\mathbf{a}\mid\mathbf{e}) = \Pr(m\mid\mathbf{e})\prod_{j=1}^{m}\Pr(a_j \mid a_1^{j-1}, f_1^{j-1}, m, \mathbf{e})\Pr(f_j \mid a_1^{j}, f_1^{j-1}, m, \mathbf{e})$$

Nå skal vi gjøre approksimasjoner.

$\Pr(m\mid\mathbf{e})$ er uavhengig av m og **e**.

$\Pr(a_j \mid a_1^{j-1}, f_1^{j-1}, m, \mathbf{e}) = (k+1)^{-1}$, dvs den avhenger bare av lengden k av **e**.

$\Pr(f_j \mid a_1^{j}, f_1^{j-1}, m, \mathbf{e}) = t(f_j \mid e_{a_j})$, oversettelsessannsynligheten av f_j, gitt e_{a_j}, dvs den avhenger bare av det ordet den er forbundet med, jfr tagging.

Hadde vi hatt et word-aligned korpus kunne vi estimert denne ved opptelling:

$$t(f_j \mid e_{a_j}) = \frac{C(f_j, e_{a_j})}{\sum_f C(f, e_{a_j})}$$

$$\Pr(\mathbf{f},\mathbf{a}\mid\mathbf{e}) = \varepsilon\prod_{j=1}^{m}(k+1)^{-1}t(f_j \mid e_{a_j})$$

$$\Pr(\mathbf{f},\mathbf{a}\mid\mathbf{e}) = \frac{\varepsilon}{(k+1)^m}\prod_{j=1}^{m}t(f_j \mid e_{a_j})$$

Vi har

$$\Pr(\mathbf{f},\mathbf{a}\mid\mathbf{e}) = \frac{\varepsilon}{(k+1)^m}\prod_{j=1}^{m} t(f_j \mid e_{a_j})$$

Da vil

$$\Pr(\mathbf{f}\mid\mathbf{e}) = \sum_{\mathbf{a}} \Pr(\mathbf{f},\mathbf{a}\mid\mathbf{e}) = \sum_{\mathbf{a}} \frac{\varepsilon}{(k+1)^m}\prod_{j=1}^{m} t(f_j \mid e_{a_j})$$

$$\Pr(\mathbf{f}\mid\mathbf{e}) = \sum_{a_1=0}^{k}\cdots\sum_{am=0}^{k} \frac{\varepsilon}{(k+1)^m}\prod_{j=1}^{m} t(f_j \mid e_{a_j})$$

$$\Pr(\mathbf{f}\mid\mathbf{e}) = \frac{\varepsilon}{(k+1)^m}\prod_{j=1}^{m}\sum_{i=0}^{k} t(f_j \mid e_i)$$

IBM-modell 2

$$\Pr(\mathbf{f},\mathbf{a}\mid\mathbf{e}) = \Pr(m\mid\mathbf{e})\prod_{j=1}^{m}\Pr(a_j \mid a_1^{j-1}, f_1^{j-1}, m, \mathbf{e})\Pr(f_j \mid a_1^{j}, f_1^{j-1}, m, \mathbf{e})$$

En endring

$$\Pr(a_j \mid a_1^{j-1}, f_1^{j-1}, m, \mathbf{e}) = a(a_j \mid j, m, k),$$

Nå avhenger denne av lenden av de to strengene, og hvor vi befinner oss i den franske strengen.
Dette er også noe vi kunnet telt opp fra et word aligned korpus.

IBM-modell 3

Knight SMT-handbook:
15. Translation Modeling

……

Here's the story:

1. For each word ei in an English sentence (i = 1 ... l), we choose a fertility phi-i. The choice of fertility is dependent solely on the English word in question. It is not dependent on the other English words in the English sentence, or on their fertilities.

2. For each word ei, we generate phi-i French words. The choice of French word is dependent solely on the English word that generates it. It is not dependent on the English context around the English word. It is not dependent on other French words that have been generated from this or any other English word.

3. All those French words are permuted. Each French word is assigned an absolute target "position slot." For example, one word may be assigned position 3, and another word may be assigned position 2 -- the latter word would then precede the former in the final French sentence. The choice of position for a French word is dependent solely on the absolute position of the English word that generates it.

Is that a funny story, or what?

IBM-modell 3

(Knight: seksjon 17)

Now we are ready to see the real generative Model 3:

1. For each English word ei indexed by i = 1, 2, ..., l,
   choose fertility phi-i with probability n(phi-i | ei).

2. Choose the number phi-0 of "spurious" French words to be generated from e0 = NULL,
   using probability p1 and the sum of fertilities from step 1.

3. Let m be the sum of fertilities for all words, including NULL.

4. For each i = 0, 1, 2, ..., l, and each k = 1, 2, ..., phi-i,
   choose a French word tau-ik with probability t(tau-ik | ei).

5. For each i = 1, 2, ..., l, and each k = 1, 2, ..., phi-i,
   choose target French position pi-ik with probability d(pi-ik | i,l,m).

6. For each k = 1, 2, ..., phi-0, choose a position pi-0k from the phi-0 - k + 1
   remaining vacant positions in 1, 2, ... m, for a total probability of 1/phi-0!.

7. Output the French sentence with words tau-ik in positions pi-ik (0 <= i <= l, 1 <= k <= phi-i).

If you want to think about this in terms of string rewriting, consider the following sequence:

Mary did not slap the green witch (input)
Mary not slap slap slap the green witch (choose fertilities)
Mary not slap slap slap NULL the green witch (choose number of spurious words)
Mary no daba una botefada a la verde bruja (choose translations)
Mary no daba una botefada a la bruja verde (choose target positions)

(Et litt annet forhold mellom valg av ord og plassering enn tidligere.)

IBM-modell 4

IBM-modell 4 opererer med to typer plassering.
Tar utgangspunkt ikke i enkeltord, men i ”cept” (nesten ”concept”)
En absolutt plassering av cept (som før)
En relativ plassering innen cept

IBM-modell 5

IBM-modellene 3 og 4 er ikke ordentlige sannsynlighetsmodeller (i motsetning til 2 og 3).
Modell 5 retter på dette.

Parameterlæring

Vi har store parallellkorpus, men de er ikke aligned.

Hadde vi visst hvilke ord som er oversettelser av hvilke, kunne vi laget alignment.
Hadde vi visst alignment, kunne vi bestemt hvilke ord som er oversettelser av hvilke.
Høna og egget!

Her kommer EM-algoritmen inn.
Vi kan lære begge samtidig:
Fra en sannsynlighet for alignment kan vi lage sannsynligheter for ordoversettelser.
Fra en sannsynlighet for ordoversettelser, kan vi lage sannsynligheter for alignments.

Merk at:
Fra et aligned korpus, vil vi kunne snakke om at et ord er oversettelsen av et annet ord. Men modellen vil si noe om sannsynligheten for at ordet er en oversettelse av det andre.

Tilsvarende med alignments. Vi vil ikke ha en riktig alignment, men fordele sannsynligheten mellom ulike alignements.

Outline