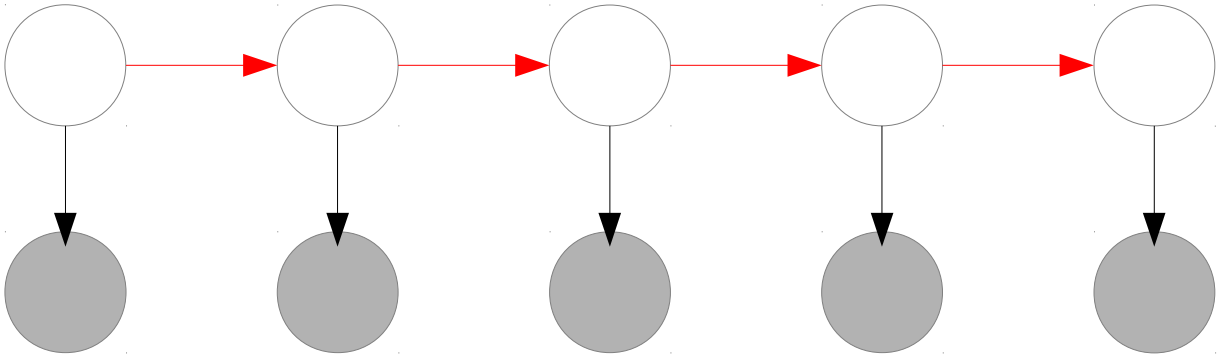
Brown klaszterek

Rejtett Markov Modell (HMM)



Rejtett változók (Y)

Megfigyelt változók (X)

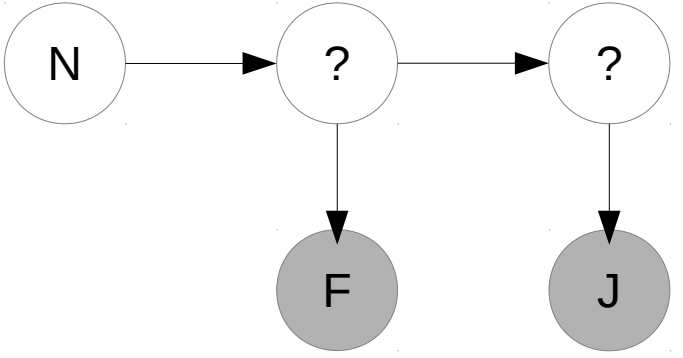
- Valószínűségi modell, amely az állapotátmenet,- és az emissziós valószínűségekből áll
  - Markov feltevés: az adott pillanat rejtett állapota csak a megelőzőtől függ
  - Segítségével a megfigyelt változókat leginkább megmagyarázni képes rejtett változókat határozhatjuk meg

HMM példa

- Tudjuk, hogy 2 napja napos idő volt, és a következő két napban forró, majd jeges teát fogyasztottunk. Milyen időjárás volt a legvalószínűbb az elmúlt 2 nap folyamán?

	Napos	Esős
Napos	0,8	0,2
Esős	0,3	0,7

	Forró	Jeges
Napos	0,1	0,9
Esős	0,6	0,4

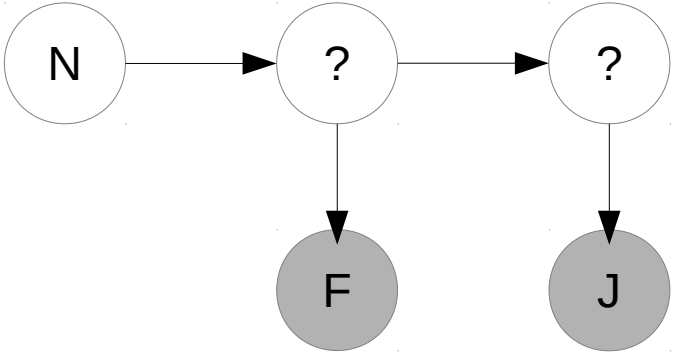


HMM példa

- Tudjuk, hogy 2 napja napos idő volt, és a következő két napban forró, majd jeges teát fogyasztottunk. Milyen időjárás volt a legvalószínűbb az elmúlt 2 nap folyamán?
  - NN→ (0,8*0,1)*(0,8*0,9)=0,0576

	Napos	Esős
Napos	0,8	0,2
Esős	0,3	0,7

	Forró	Jeges
Napos	0,1	0,9
Esős	0,6	0,4



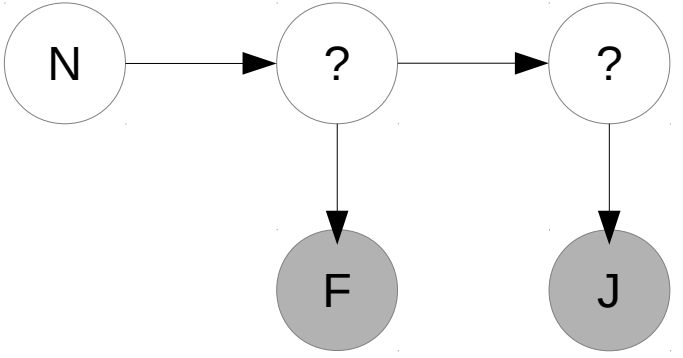
HMM példa

- Tudjuk, hogy 2 napja napos idő volt, és a következő két napban forró, majd jeges teát fogyasztottunk. Milyen időjárás volt a legvalószínűbb az elmúlt 2 nap folyamán?
  - NN→ (0,8*0,1)*(0,8*0,9)=0,0576
  - NE→ (0,8*0,1)*(0,2*0,4)=0,0064
  - EN→ (0,2*0,6)*(0,7*0,9)=0,0324
  - EE→ (0,2*0,6)*(0,7*0,4)=0,0336

Σ=0,18

	Napos	Esős
Napos	0,8	0,2
Esős	0,3	0,7

	Forró	Jeges
Napos	0,1	0,9
Esős	0,6	0,4



HMM feladatok – Tanítás

- Cél: a tanítószekvencia megfigyelését legvalószínűbbé tevő paraméterek meghatározása
  - Ha a rejtett változók ismertek (lennének), akkor egyszerű maximum likelihood módon elvégezhető
    - A rejtett változók azonban nem (legfeljebb részlegesen) ismertek
    - A szekvencia hosszában (l) és a lehetséges rejtett állapotok számában (H) exponenciálisan sok (H^l) lehetséges rejtett állapot szekvencia

HMM feladatok – Tanítás

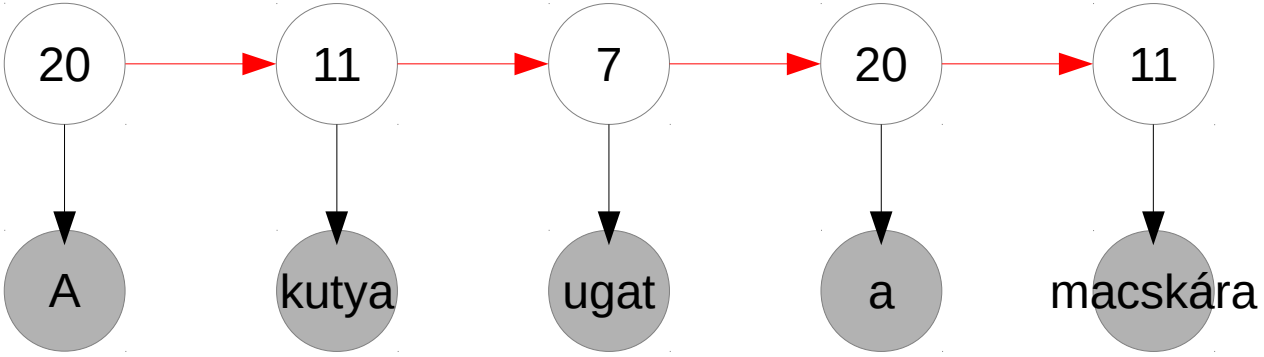
- Cél: a tanítószekvencia megfigyelését legvalószínűbbé tevő paraméterek meghatározása
  - Ha a rejtett változók ismertek (lennének), akkor egyszerű maximum likelihood módon elvégezhető
    - A rejtett változók azonban nem (legfeljebb részlegesen) ismertek
      - Expectation Maximization (EM) algoritmus
    - A szekvencia hosszában (l) és a lehetséges rejtett állapotok számában (H) exponenciálisan sok (H^l) lehetséges rejtett állapot szekvencia
      - Dinamikus programozással kiküszöbölhető a H^l szekvencia explicit kiszámítása

HMM feladatok – Inferencia

- Cél: a modell paraméterei alapján a megfigyeléseket legjobban magyarázó rejtett állapotsorozat meghatározása
  - A szekvencia hosszában (l) és a lehetséges rejtett állapotok számában (H) exponenciálisan sok (H^l) lehetséges magyarázó rejtett állapot szekvencia
    - A tanítás során használthoz hasonló dinamikus programozási megoldás

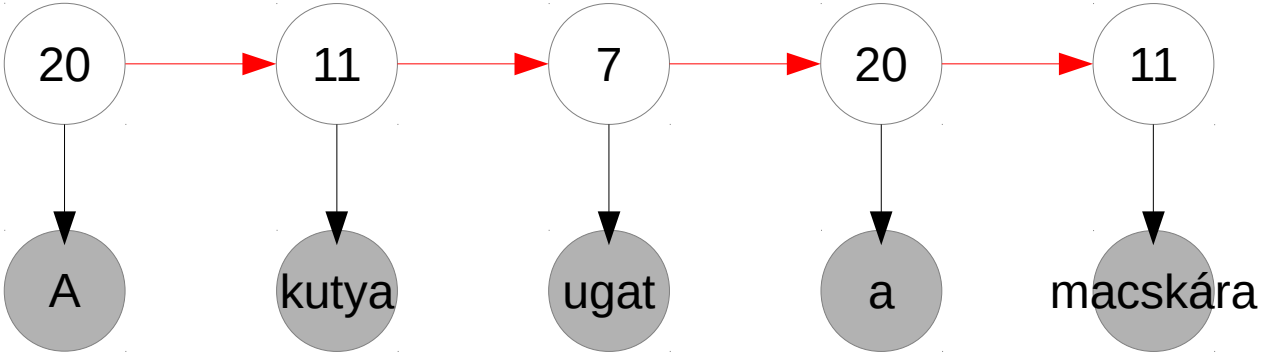
Brown klaszterezés

- Tegyük fel, hogy minden egyes korpuszban megfigyelt szót egy **rejtett** szóosztályok “generálják”
  - Pl. a {macska, kutya, egér, ...} szavakat egy adott (állatokhoz kötődő dolgokat összefogó) klaszter generálja



Brown klaszterezés

- Tegyük fel, hogy minden egyes korpuszban megfigyelt szót egy **rejtett** szóosztályok “generálják”
  - Pl. a {macska, kutya, egér, ...} szavakat egy adott (állatokhoz kötődő dolgokat összefogó) klaszter generálja



Lényegében egy HMM-el van dolgunk!

Brown klaszterek kiértékelése

- Egy adott B: szó→ klaszter hozzárendelés minőségét a megfigyeléseink log-likelihoodjával jellemezhetjük
  - A szavak klaszterezése áttételesen kihat az emissziós és tranzíciós paraméterek értékeire

$$\text{Minőség}(B) = \Sigma_i \log e(w_i \mid B(w_i)) * t(B(w_i \mid w_{i-1})) = \ldots =$$

$$= \Sigma_b \Sigma_{b'} p(b, b') [\log p(b, b') - \log p(b) - \log(b')] - H(w)$$

$$= MI(B) - H(w)$$

Brown klaszterek kiértékelése

- Egy adott B: szó→ klaszter hozzárendelés minőségét a megfigyeléseink log-likelihoodjával jellemezhetjük
  - A szavak klaszterezése áttételesen kihat az emissziós és tranzíciós paraméterek értékeire

$$\text{Minőség}(B) = \Sigma_i \log e(w_i \mid B(w_i)) * t(B(w_i \mid w_{i-1})) = \ldots =$$

$$= \Sigma_b \Sigma_{b'} p(b, b') [\log p(b, b') - \log p(b) - \log(b')] - H(w)$$

$$= MI(B) - H(w)$$

vagyis a klaszterek kölcsönös információtartalmának és a szavak entrópiájának (B-től nem függő) összege

Egyszerű algoritmus

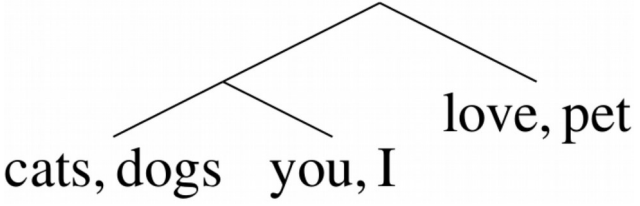
- Mind a $|V|$ szóalak kerüljön egy külön klaszterbe
- $|V|$-c összevonást végrehajtva alakítsunk ki c klasztert
  - Egy összevonás során mohó módon egyesítsünk két klasztert úgy, hogy Minőség(B) a legjobban nőjön
- Naív implementációja $O(|V|^5)$
  - Még a hatékony verziója is $O(|V|^3)$, ami a gyakorlatban túl lassú

Hatékony algoritmus

- Vegyük a *c* leggyakoribb szót, tekintsünk mindegyikre önálló klaszterként
  - Egy lépésben *c* klaszterből válasszuk ki azt, amelyik a Minőség(B) mutatót a legnagyobb mértékben növeli
  - Az összevonást kompenzálandó, a gyakoriság szerint következő szót vegyük föl egy új klaszterként, majd vonjunk össze újra

Hatékony algoritmus

- Vegyük a *c* leggyakoribb szót, tekintsünk mindegyikre önálló klaszterként
  - Egy lépésben *c* klaszterből válasszuk ki azt, amelyik a Minőség(B) mutatót a legnagyobb mértékben növeli
  - Az összevonást kompenzálandó, a gyakoriság szerint következő szót vegyük föl egy új klaszterként, majd vonjunk össze újra
- $|V|-c$ iteráció után a teljes szótárt földolgozzuk
  - Összességében $O(|V|c^2+n)$ művelet, ahol *n* a korpusz mérete
  - Hierarchiát fogunk kapni



Példa Brown klaszterek

- Magyar Twitterről Percy Liang implementációjával kinyerve

cluster path 0110110010

490 words, 14,612 tokens freq alpha suffix

Words in frequency order

1	ülföld	1,196
2	megnövekedett	930
3	áció	461
4	ép	424
5	ülföldifoci	337
6	ácsony	328
7	ékelyföld	327
8	átokközt	314
9	ánsok	307

cluster path 110010111010

37 words, 13,219 tokens freq alpha suffix

Words in frequency order

1	akit	3,231
2	amire	1,889
3	amiért	1,650
4	amiket	1,622
5	akivel	1,030
6	akiket	932
7	amiről	799
8	akikkel	309
9	akire	289

Példa Brown klaszterek

- Magyar Twitterről Percy Liang implementációjával kinyerve

cluster path 0110110010

490 words, 14,612 tokens freq alpha suffix

Words in frequency order

1	ülföld	1,196
2	megnövekedett	930
3	áció	461
4	ép	424
5	ülföldifoci	337
6	ácsony	328
7	ékelyföld	327

cluster path 110010111010

37 words, 13,219 tokens freq alpha suffix

Words in frequency order

1	akit	3,231
2	amire	1,889
3	amiért	1,650
4	amiket	1,622
5	akivel	1,030
6	akiket	932
7	amiről	799

^01101011110 (522)	#szjg
^011010111110 (400)	#adal #wtf
^011010111111 (400)	

Tune Your Brown Clustering, Please (Derczynski et al., 2015)

- Avoid default values of c
- Getting a big corpus is more helpful than generating a high number of clusters, though watch out: very small numbers of clusters can be bad with larger corpora
- If you care more about path information (maybe you're dealing with tweets or NER), make c as big as you can; if you care more about how words are clustered together (maybe you're doing text normalisation), avoid making c too big
- Try random search for c, weighted away from very low and high values of c
- To save time, start your parameter search using some of our pre-generated clusterings (download link below)

Tune Your Brown Clustering, Please (Derczynski et al., 2015)

- **Avoid default values of c**
- A big corpus is helps more than a large *c*
  - Caveat: big corpus & small *c* can be a bad combination
- If path information matters → increase *c* as much as possible
- Try random search for *c*, weighted away from extreme values

Összefoglalás

- Brown klaszterekkel koherens szemantikus csoportokba tudjuk rendezni a szóalakokat
  - A ritka szóvektorokhoz hasonló eredményt ad azzal a különbséggel, hogy egy szó egy fix klaszterbe tartozik
- A kialakuló hierarchia alapján beszélhetünk a klaszterek részleges hasonlóságáról (az átfedő prefixek mentén)