Rejtett Markov Modell

A Rejtett Markov Modell használata beszédfelismerésben

Készítette Feldhoffer Gergely
felhasználva Fodróczi Zoltán előadásanyagát

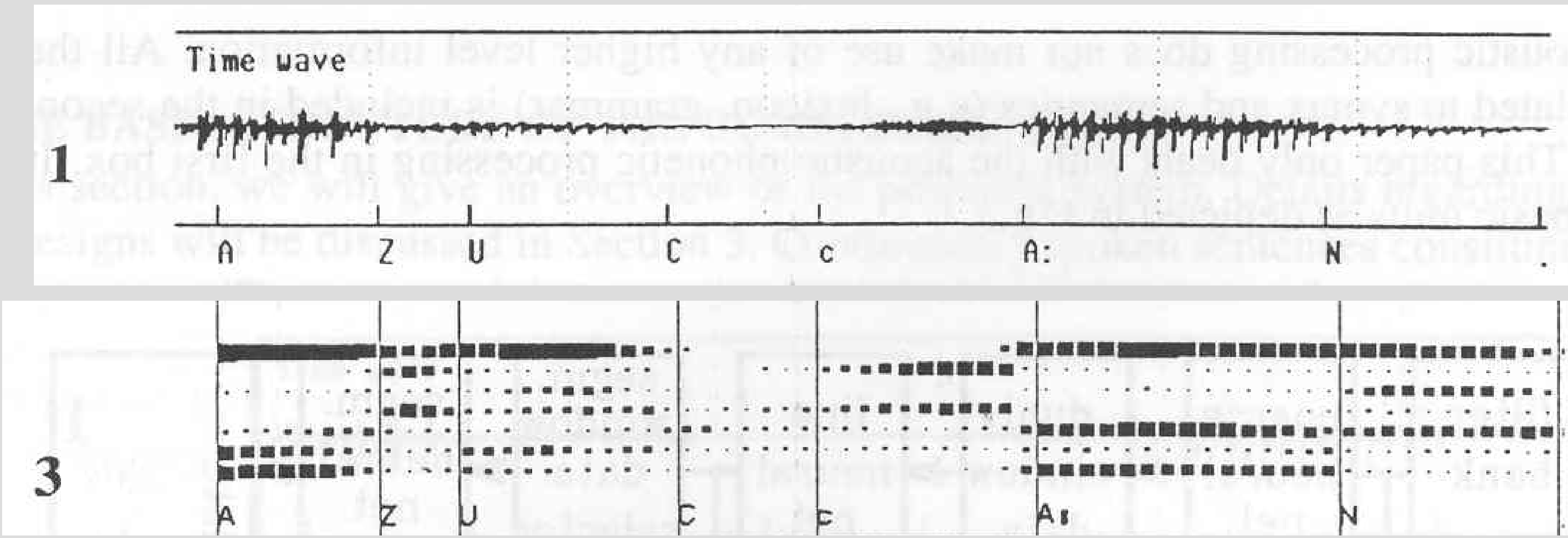
Áttekintés

- „hagyományos” Markov Modell
- Beszédfelismerésbeli szerep
- Rejtett Markov Modell
- Módszerek
- Értékelés

Miről is van szó

- Beszédfelismerő módszerek
  - Izolált, kötött szótáras
    - DTW
    - …
  - Folyamatos
    - Neuronhálós megoldások
    - HMM megoldások
    - …
- Statisztikai alapok

Példa folyamatos feldolgozásra

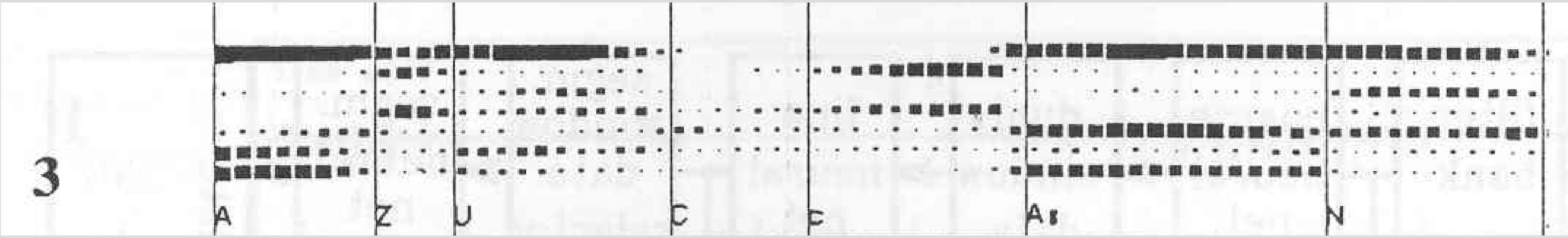


Jellegvektor (Feature vector): olyan jellemzők, amik minden keretre kiszámítva alkalmasak felismerésre, szegmentálásra. A fenti képen a kockák méretét egy jellegvektor adja meg.

Folyamatos feldolgozás

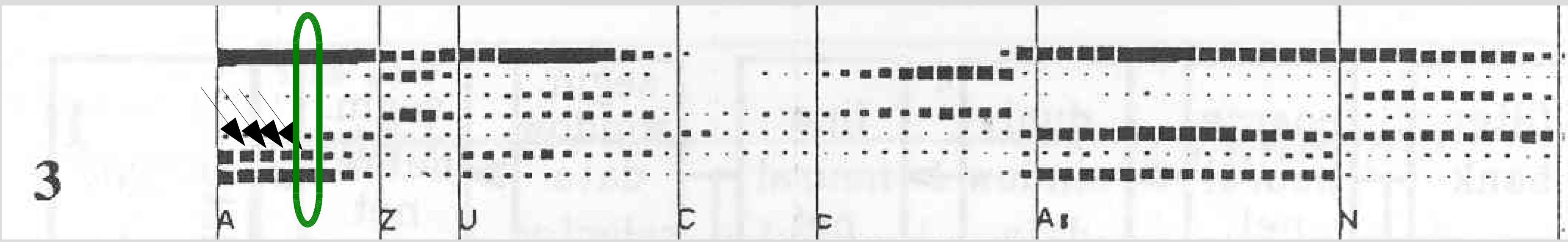
- Célok:
  - szegmentálás
  - fonéma azonosítás
  - szóhatár döntések
- Bemenet: jellegvektorok
- Nehézségek:
  - bizonytalan, zajjal terhelt bemenet
  - sok hasonló fonéma, hasonulások, kiejtésbeli változatosság
  - szóhatáron nincs szünet, ismerni kell a nyelvet

Folyamatos feldolgozás



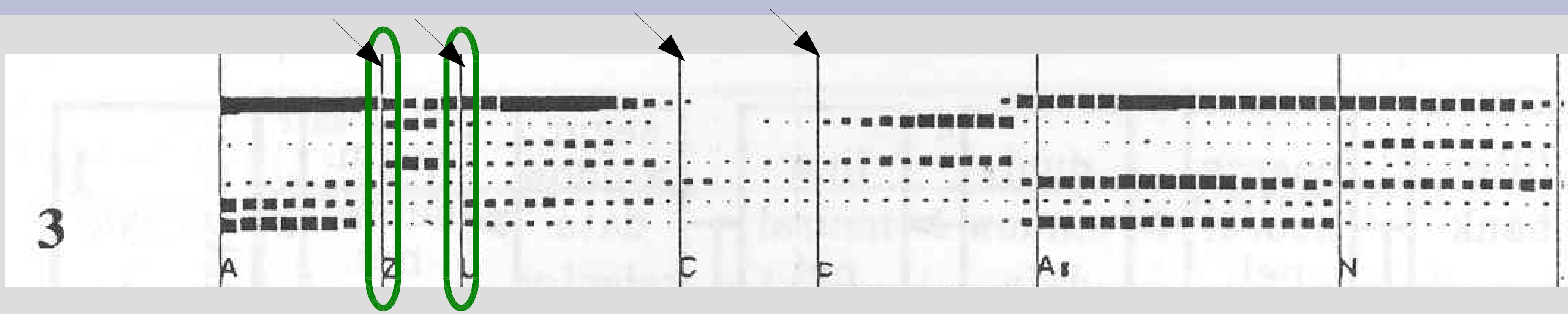
- Jellegvektorok
- Szegmentálás
- Fonéma kiválasztása
- Szóhatárok

Folyamatos feldolgozás



- **Jellegvektorok**
- Szegmentálás
- Fonéma kiválasztása
- Szóhatárok

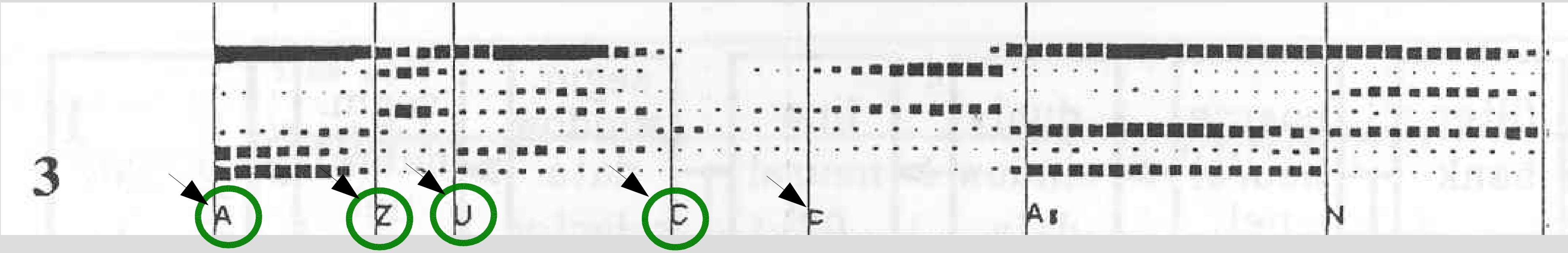
Folyamatos feldolgozás



- Jellegvektorok
- **Szegmentálás**
- Fonéma kiválasztása
- Szóhatárok

Szemre
is látható

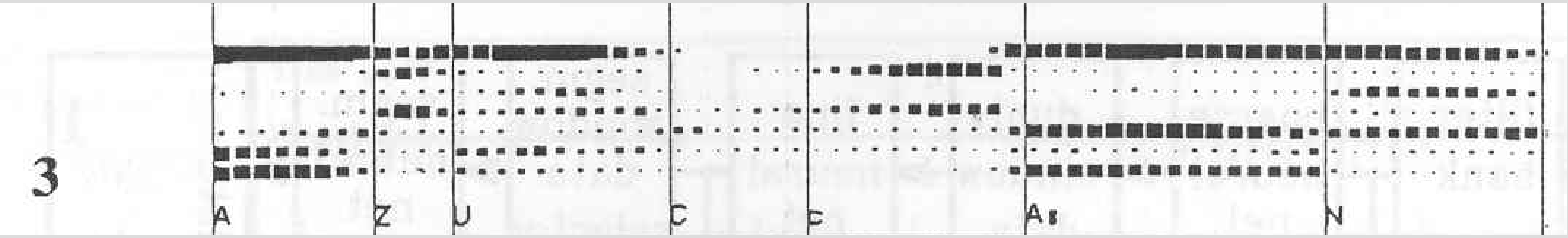
Folyamatos feldolgozás



- Jellegvektorok
- Szegmentálás
- **Fonéma kiválasztása**
- Szóhatárok

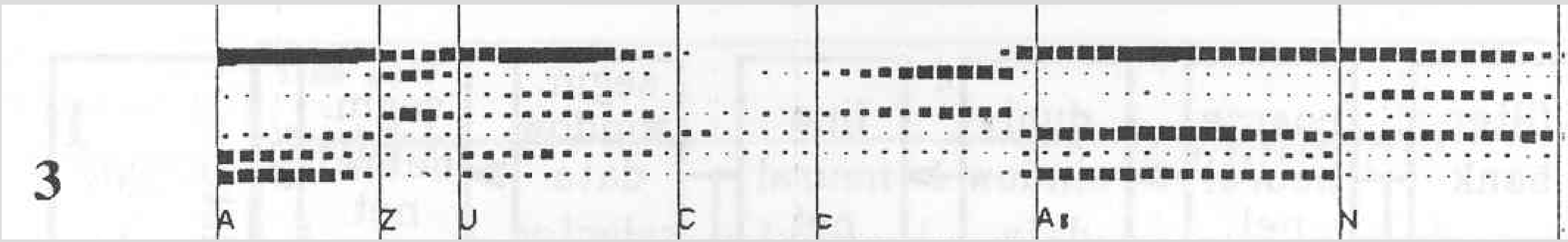
Szemre kevéssé látható

Folyamatos feldolgozás



- Jellegvektorok
- Szegmentálás
- Fonéma kiválasztása
  - Fonéma jelöltek pontozása:
    - jellegvektorok
    - statisztika
  - Statisztika és szótár alapján választás
- Szóhatárok

Folyamatos feldolgozás



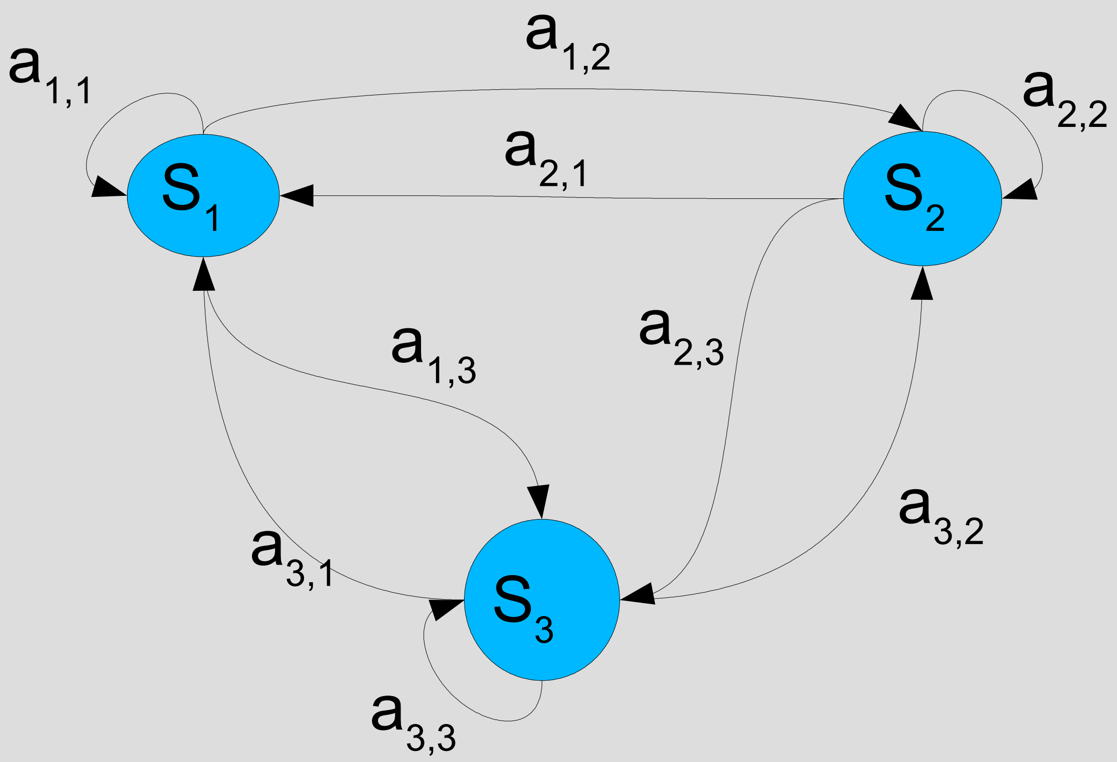
- Jellegvektorok
- Szegmentálás
- Fonéma kiválasztása
  - Fonéma jelöltek pontozása:
    - jellegvektorok
    - statisztika
  - Statisztika és szótár alapján választás
- Szóhatárok

Neuronháló

HMM

Markov Modell

- Állapotok, állapotátmenetek



Markov Modell példa: időjárás

- Állapotok: S={S1:Napos, S2:Borult, S3:Esős}
- Állapotátmenetek:

$$A=\{a_{i,j}\}=\begin{vmatrix} 0,4 & 0,3 & 0,3 \\ 0,2 & 0,6 & 0,2 \\ 0,1 & 0,1 & 0,8 \end{vmatrix}$$

- Kérdés: adott állapot-sorozat valószínűsége
- Pl: Mi az esélye, hogy egy Napos állapot után {Napos, Esős} sorozat következzen?

Markov Modell példa: időjárás

- Állapotok: S={S1:Napos, S2:Borult, S3:Esős}
- Állapotátmenetek:

$$A=\{a_{i,j}\}=\begin{vmatrix} 0,4 & 0,3 & 0,3 \\ 0,2 & 0,6 & 0,2 \\ 0,1 & 0,1 & 0,8 \end{vmatrix}$$

- Kérdés: adott állapot-sorozat valószínűsége
- Pl: Mi az esélye, hogy egy Napos állapot után {Napos, Esős} sorozat következzen?
- $P(\{S1,S1,S3\}|S1)=1*a_{1,1}*a_{1,3}=1*0.8*0.3=0.24$

Markov Modell Beszédfelismerésben

- Az állapotokat a fonémáknak feleltetjük meg
- Az állapotátmenetek ismertek:
- Nagy adatbázisokból kinyerhető az A mátrix
- Állapotok?

Markov Modell Beszédfelismerésben

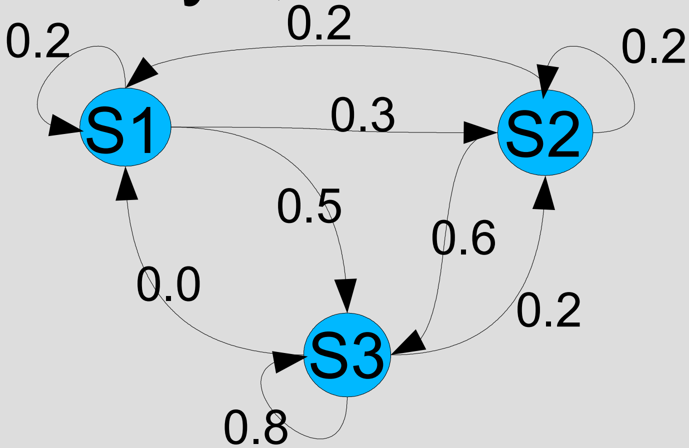
- Az állapotokat a fonémáknak feleltetjük meg
- Az állapotátmenetek ismertek:
- Nagy adatbázisokból kinyerhető az A mátrix
- Állapotok?
- Sajnos éppen az állapotok bizonytalanok
- Megfigyelésekre hagyatkozhatunk csak
- Viszont vannak megfigyelt sorozataink!

A Rejtett Markov Modell (HMM)

- Minden állapothoz hozzárendelhető a lehetséges megfigyelések valószínűsége
- $e_{si}(x)$: x megfigyelés valószínűsége az i állapotban
- Ismerjük az állapotátmeneti valószínűségeket: Az A mátrix
- Keressük az állapotok sorozatát megfigyelések sorozata alapján

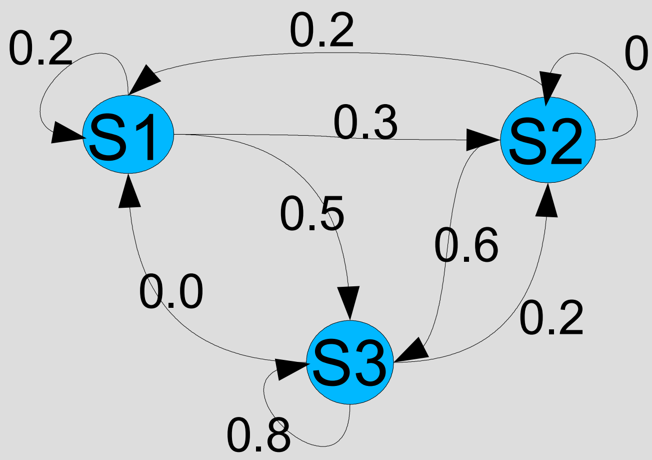
A Rejtett Markov Modell (HMM) Példa

- Egy oktató többféle állapotban lehet: S1:jókedvű, S2:közömbös, S3:ideges
- A házifeladat feladásában befolyásolja ez az állapot, de nem határozza meg teljesen
- A házifeladat lehet O1:könnyű, O2:félórás, O3:többórás
- e_{S1}={0.7,0.2,0.1}
- e_{S2}={0.3,0.4,0.3}
- e_{S3}={0.0,0.1,0.9}
- Tudjuk:milyen házifeladatot adott egy hétig



A Rejtett Markov Modell (HMM) Példa

- Megfigyelések: {O1, O3, O1, O2, O3}
- Keressük a legvalószínűbb állapot-sorozatot
- Megoldás: Viterbi algoritmus
  - halmozott valószínűségek
  - utolsó állapot
  - visszakövetkeztetés



A Rejtett Markov Modell (HMM) Példa: Viterbi

	Könnyű	Többórás	Könnyű	Félórás	Többórás
Jókedvű	0,7/3				
Közömbös	0,3/3				
Ideges	0,0/3				

Az első oszlopba kerülnek azok a valószínűségek, amik kezdetben érvényesek. Feltételezve egyenletes eloszlást, a megfigyelések valószínűségeit elosztjuk az állapotok számával (a későbbiekben ennél okosabban is lehet csinálni, például külön kezelni a kezdőállapotok valószínűségeit)

A Rejtett Markov Modell (HMM)
Példa: Viterbi

	Könnyű	Többórás	Könnyű	Félórás	Többórás
Jókedvű	0.233	?			
Közömbös	0.100				
Ideges	0.000				

A következő oszlopba már az állapotátmenet valószínűsége-ket is bele kell számolni.

$V_{2,l}=e_l(O3)*max_k\{V_{1,k}*a_{k,l}\}$

$V_{2,S1}=e_{S1}(O3)*max\{ V_{1,1}*a_{1,1} , V_{1,2}*a_{2,1} ,V_{1,3}*a_{3,1}\}$

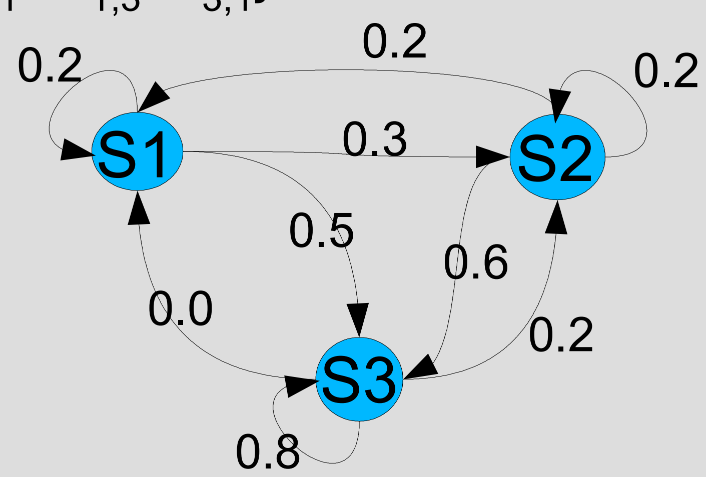
$V_{2,S1}=0,1*max\{ 0.233*0.2=\boldsymbol{0.0466}, , 0.1*0.3=\boldsymbol{0.03}, 0.0*0.5=\boldsymbol{0} \}$

$V_{2,S1}=0.00466$

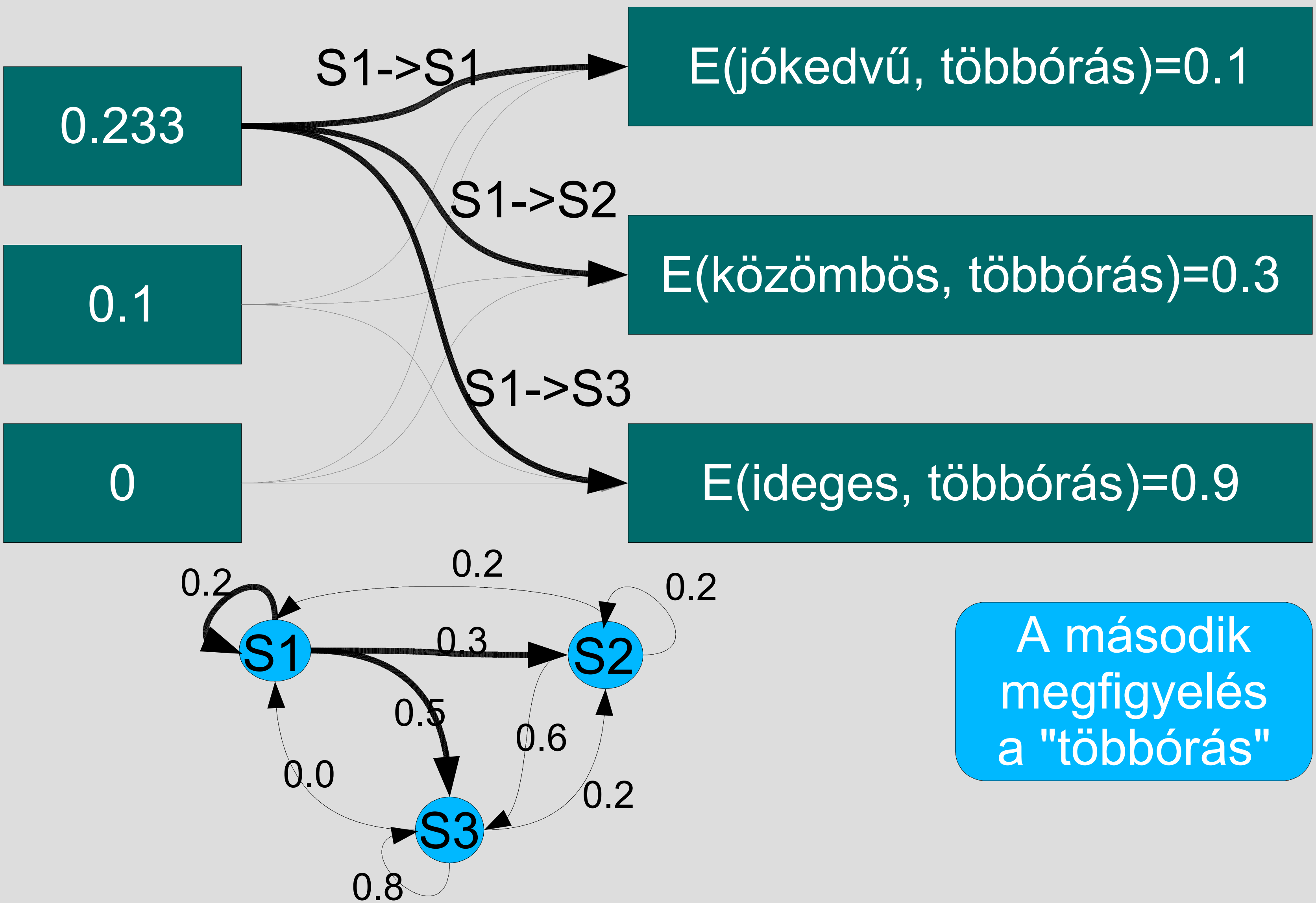
$e_{S1}=\{0.7,0.2,0.1\}$

$e_{S2}=\{0.3,0.4,0.3\}$

$e_{S3}=\{0.0,0.1,0.9\}$

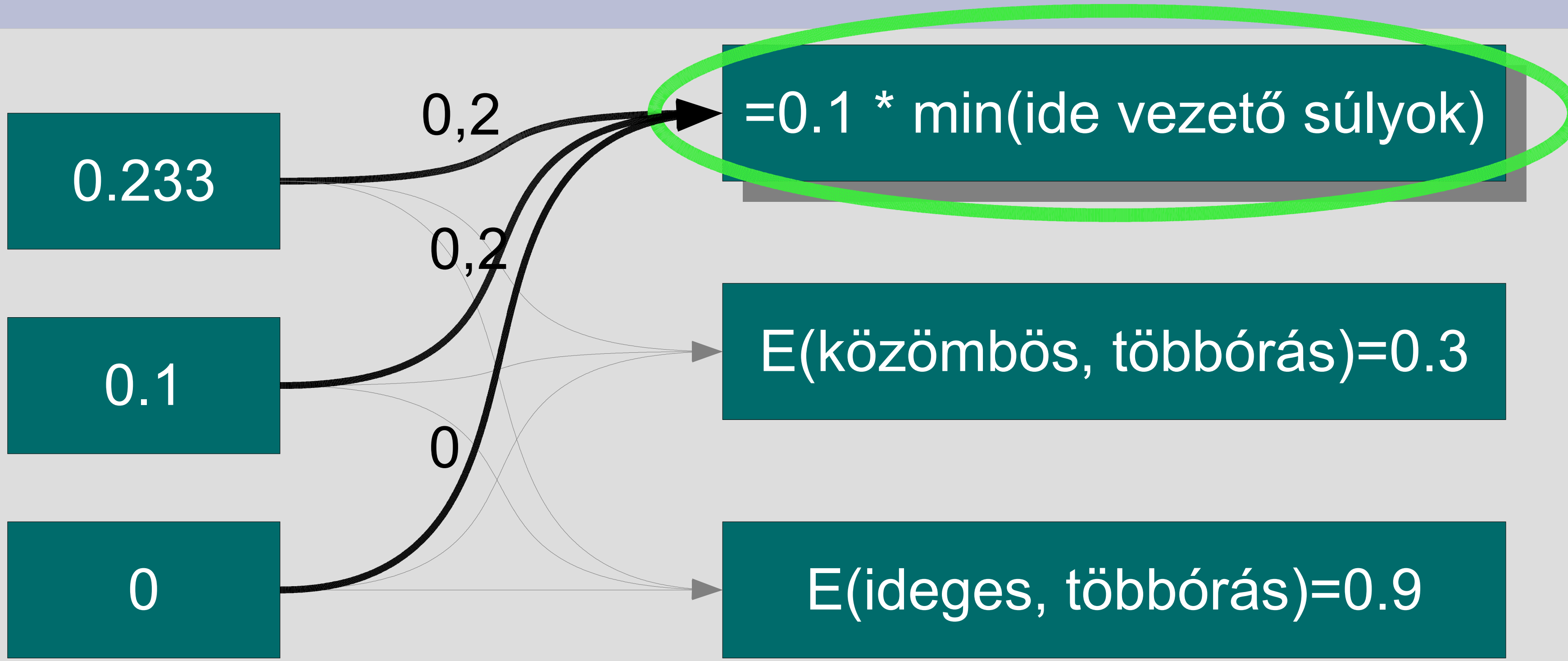


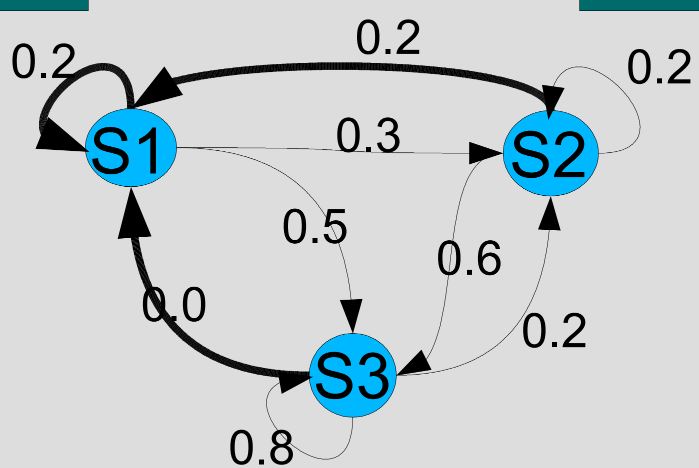
A Rejtett Markov Modell (HMM)
Példa: Viterbi



A második megfigyelés a "többórás"

A Rejtett Markov Modell (HMM)
Példa: Viterbi





A súly az élek súlya és a kiindulópont szorzata

A Rejtett Markov Modell (HMM)
Példa: Viterbi

	Könnyű	Többórás	Könnyű	Félórás	Többórás
Jókedvű	0.233	0.00466			
Közömbös	0.100				
Ideges	0.000				

A következő oszlopba már az állapotátmenet valószínűsége-ket is bele kell számolni.

$V_{2,l}=e_l(O3)*\max_k\{V_{1,k}*a_{k,l}\}$

$V_{2,S1}=e_{S1}(O3)*\max\{ V_{1,1}*a_{1,1} , V_{1,2}*a_{2,1} ,V_{1,3}*a_{3,1}\}$

$V_{2,S1}=0,1*\max\{$ **0.233*0.2=0.0466**,

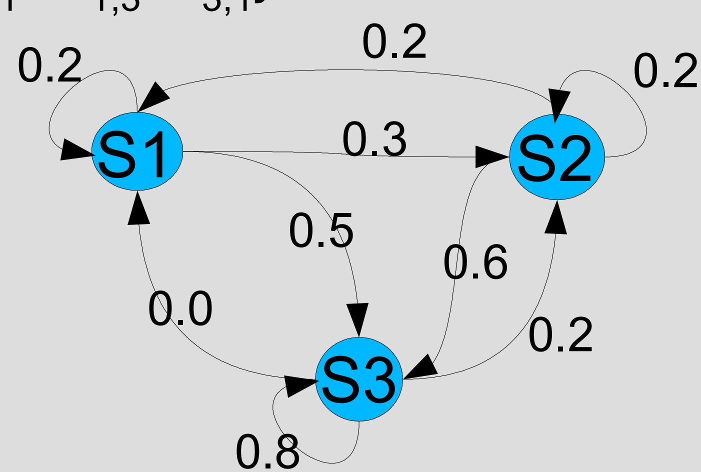
, 0.1*0.3=0.03, 0.0*0.5=0 $\}$

$V_{2,S1}=0.00466$

$e_{S1}=\{0.7,0.2,0.1\}$

$e_{S2}=\{0.3,0.4,0.3\}$

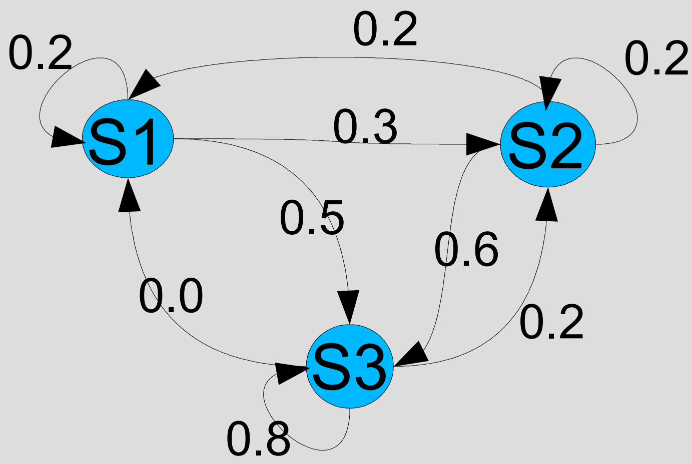
$e_{S3}=\{0.0,0.1,0.9\}$



A Rejtett Markov Modell (HMM)
Példa: Viterbi

	Könnyű	Többórás	Könnyű	Félórás	Többórás
Jókedvű	0.233	0.00466			
Közömbös	0.100	0.0207			
Ideges	0.000	0.105			

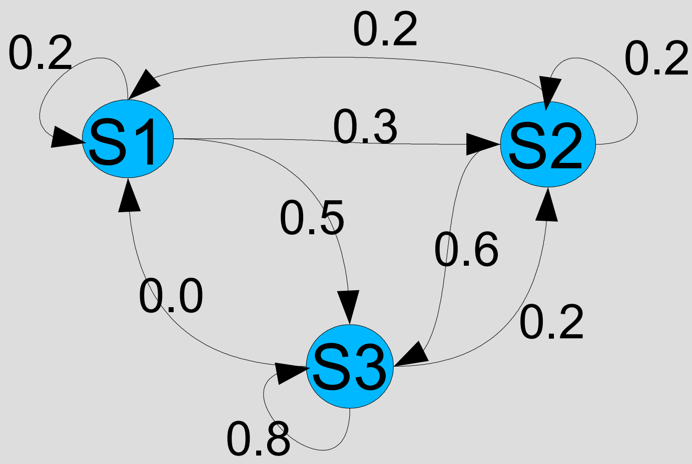
Minden értékhez megjegyezzük, hogy a maximális valószínűség melyik állapot felől következett



A Rejtett Markov Modell (HMM) Példa: Viterbi

	Könnyű	Többórás	Könnyű	Félórás	Többórás
Jókedvű	0.233	0.00466	0.00294	0.000252	0.0000108
Közömbös	0.100	0.0207	0.0063	0.000504	0.0000324
Ideges	0.000	0.105	0.000	0.000378	0.0002721

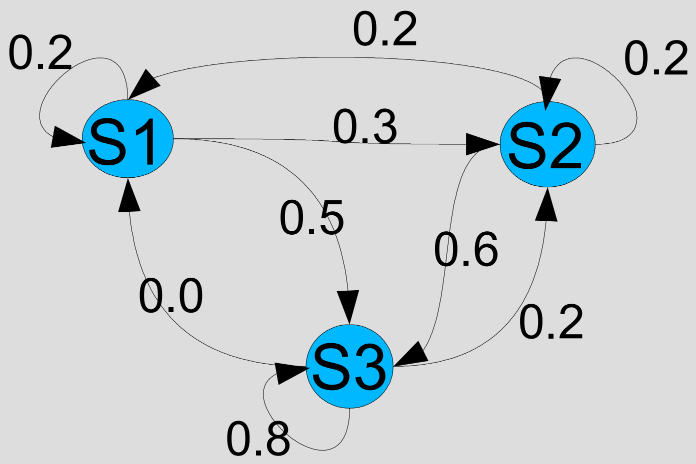
Végigszámoljuk az egész táblát



A Rejtett Markov Modell (HMM)
Példa: Viterbi

	Könnyű	Többórás	Könnyű	Félórás	Többórás
Jókedvű	0.233	0.00466	0.00294	0.000252	0.0000108
Közömbös	0.100	0.0207	0.0063	0.000504	0.0000324
Ideges	0.000	0.105	0.000	0.000378	0.0002721

Kiválasztjuk a maximumot az utolsó állapotban.
A halmozott valószínűségek közül a maximálisat választva egy egész sorozatot választunk

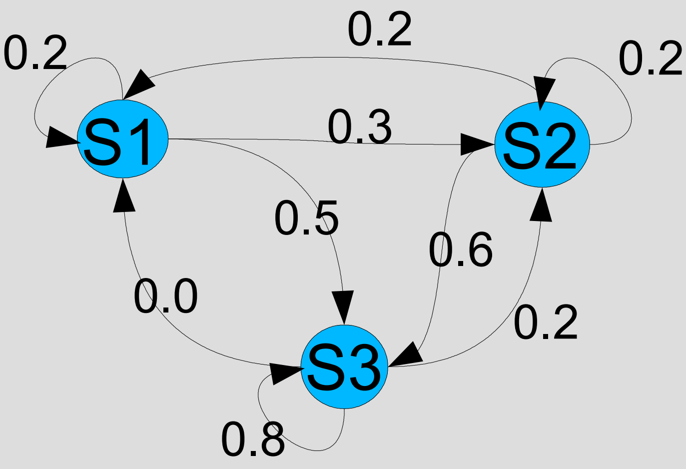


A Rejtett Markov Modell (HMM)
Példa: Viterbi

	Könnyű	Többórás	Könnyű	Félórás	Többórás
Jókedvű	0.233	0.00466	0.00294	0.000252	0.0000108
Közömbös	0.100	0.0207	0.0063	0.000504	0.0000324
Ideges	0.000	0.105	0.000	0.000378	0.0002721

Így a maximális választások útján visszafelé haladva a teljes állapot-sorozat visszakövetkeztethető.

Az eredmény: {S1,S3,S2,S3,S3}



A Rejtett Markov Modell (HMM)
Példa: Viterbi

	Könnyű	Többórás	Könnyű	Félórás	Többórás
Jókedvű	0.233	0.00466	0.00294	0.000252	0.0000108
Közömbös	0.100	0.0207	0.0063	0.000504	0.0000324
Ideges	0.000	0.105	0.000	0.000378	0.0002721

Megjegyzés:
Látható, hogy vannak olyan értékek, amik helyben maximálisak, mégsem képezik részét a végeredményben kapott útnak. A naiv, egyértelműen hozzárendelő elv szerint is, a jelzett valószínűség szerint is az S2 lett volna a nyerő, de a teljes folyamatban ez már nem igaz.

Viterbi algoritmus C++ nyelven

```
for (int i=1;i<o.size();i++) {
	for (int j=0;j<a.size();j++) {
		maxv=0;
		for (int k=0;k<3;k++)
		{
			tmp=v[k][i-1]*a[k][j];
			if (maxv<tmp) {maxv=tmp; max=k;}
		}
		v[j][i]=maxv*e[j][o[i]];
	}
}
```

Ahol `a` az A mátrix, `o` a megfigyelések sorozata, `e` az állapothoz tartozó megfigyelés-valószínűségek mátrixa, `v` a Viterbi táblázat.

Összefoglalás

- A HMM jól leírja a beszédfolyamatokat
- Megfigyelés bármilyen jellegvektor lehet
- Állapotoknak a fonémákat nevezzük ki
  - Az optimális állapotsorozat feladatára a Viterbi algoritmus a megoldás
- A módszer a dinamikus idővetemítéshez hason-lóan egy valószínűséghalmozás, és egy visszafe-lé haladó optimális út keresés
- Dinamikus programozás
- Kitekintés: Viterbi eredetileg jelátvitel hibajavítá-sára találta ki az algoritmust

Más HMM algoritmusok

- Az előrefelé következtetés
- A kérdés: egy bizonyos megfigyeléssorozat-nak mi a valószínűsége
- A megoldás: HMM-Forward algoritmus
- Az előző táblázathoz hasonló, de maximum-kiválasztás helyett összegzés: minden lehe-tőséget összegzünk, nem legjobbat kere-sünk
- $f_{S,i}=e_S(O)\Sigma_k(f_{k,i-1}*a_{k,S})$

Forward algoritmus

	Könnyű	Többórás	Könnyű	Félórás	Többórás
Jókedvű	0.233	?			
Közömbös	0.100				
Ideges	0.000				

Az első oszlop ugyanúgy megy, mint a Viterbinél, majd

$f_{S,i}=e_S(O)\Sigma_k(f_{k,i-1}*a_{k,S})$

A következő oszlop mindegyikére ki kell számolni a lehetséges útvonalakhoz tartozó valószínűségek összegét:

(0.233*0.2+0.1*0.2+0.00*0.5)=0,066

Majd ezt megszorozni a megfigyelés valószínűségével:

0,1*0,066=0,0066

Forward algoritmus

	Könnyű	Többórás	Könnyű	Félórás	Többórás
Jókedvű	0.233	0.0066	0.004713	0.000658	0.000043
Közömbös	0.100	0.027	0.01176	0.001506	0.000206
Ideges	0.000	0.159	0.000	0.000941	0.001787

Kiszámítjuk az összegeket.
A végeredmény az utolsó oszlop összege: 0,002036
Ez tehát az a valószínűség, ami egyenletes eloszlás szerinti kezdőállapot mellett a fenti megfigyelés-sorozathoz tartozik

Forward algoritmus C++ nyelven

```
for (int i=1;i<o.size();i++) {
    for (int j=0;j<a.size();j++) {
        sum=0;
        for (int k=0;k<3;k++)
            sum +=f[k][i-1]*a[k][j];
        f[j][i]=sum*e[j][o[i]];
    }
}
```

Ahol az `a` a rákövetkezési mátrix, az `e` a megfigyelések valószínű-sége, `o` a megfigyelések sorozata, `f` a Forward mátrixa.

Paraméterek becslése

- Ahhoz, hogy egy HMM sikeresen működjön beszédfelismerési, vagy egyéb rendszerekben, az A és E mátrixok pontos becslése szükséges.
- Ezeket az értékeket adatbázisok tartalmából ki lehet számítani megközelítőleg
- A megoldás a statisztika, azon belül is a maximum likelihood módszer.

Paraméterek naív becslése

- Az A mátrix elemeinek becslése:
- Legyen $T_{i,j}$ az adatbázisban található $i \rightarrow j$ átmenetek száma.
- $a_{i,j}=T_{i,j}/(\Sigma_k T_{i,k})$
- Így igaz lesz, hogy $\Sigma_k a_{i,k}=1$ minden i-re

Paraméterek naív becslése

- Az E mátrix elemeinek becslése:
- Legyen $O_{i,j}$ az adatbázisban található j megfigyelések száma, amik i állapotban fordultak elő.
- $e_{i,j}=O_{i,j}/(\Sigma_k O_{i,k})$
- Így igaz lesz, hogy $\Sigma_k e_{i,k}=1$ minden i-re

Paraméterek becslése

- A naív becslés sajnos nem optimális
- Az optimális paraméterek megadására analitikus megoldás nem ismert
- Közelítő módszerrel pontosabb:
- Baum-Welch algoritmus

Baum – Welch algoritmus

- Jelölje V a Viterbi algoritmus táblázatát, és F a Forward algoritmusét.
- V(i,j) illetve F(i,j) az i-edik megfigyelés j állapothoz tartozó valószínűsége
- Jelentse Mod=(A,E) egy modell minden paraméterét, ezt szeretnénk optimalizálni
- Végül jelentse O a megfigyelések sorozatát
- A BW algoritmus már 3 dimenziós táblával dolgozik
- Jelentse $BW(t,i,j) = P(S_t=S_i \; \& \; S_{t+1}=S_j | O,Mod)$

Baum – Welch algoritmus

- $BW(t,i,j) = P(S_t=S_i \ \& \ S_{t+1}=S_j | \ O,Mod)$
- $BW(t,i,j) =$
  $P(S_t=S_i \ \& \ S_{t+1}=S_j \ \& \ O \ | \ Mod) \ / \ P(O|Mod)$
- A levezetést nem fejtjük ki, ami fontos:
- Ez a valószínűség kifejezhető a Viterbi és a Forward algoritmusok táblázataival
- Ennek alapján egy iteratív módszerrel egyre jobb Mod modell adható

Baum – Welch algoritmus

$$A(i,j) = \frac{\text{Si} \blacktriangleright \text{Sj állapotátmenetek várható értéke}}{\text{Si-ből induló állapotátmenetek várható értéke}}$$

$$E(i,j) = \frac{\text{Si állapotban Oj-t megfigyelések várható értéke}}{\text{Si állapot előfordulásának várható értéke}}$$

- Vegyük észre: Ha az A és E mátrixok változnak, más lesz a várható érték – ettől lesz az algoritmus iteratív
- Fixpont tétel: Ez egy kontrakció, tehát ha egy Mod paraméterállapota a fenti iteráció egy lépése ugyanezt a Mod paraméterállapotot adja, akkor az a keresett szélsőérték.

Baum – Welch algoritmus Összefoglalás

- A naív becslés egy torzítatlan egylépéses iteráció a BW algoritmushoz képest
- A tanítás iteratív volta miatt a neuronhálózatos módszerekhez hasonlóan a tanítás sebessége a tanító adatbázis méretétől és jellegétől függ
- Statisztikai szemszög: Gauss eloszlások keverékét szeretnénk szeparált (ismert) Gauss eloszlások összegeként előállítani

HMM: Értékelés

- Az állapot-sorozatok elemzésével a fonémák kinyerhetőek a beszédjelből
- Ez jóval erősebb eszköz a dinamikus idővetemítésnél, ami csak egész kifejezések felismerésére használható
- Tanítható
- Részben beszélőfüggetlen, a beszélőn főleg az E mátrix múlik
- A megfigyelések halmaza tetszőlegesen választható (pl spektrális információ és energia, egyéb akusztikus jegyek)

Problémák

- Szegmentálás: ekvidisztáns időkeretek, vagy szegmentálási algoritmus?
- Egyforma keretek esetében sok múlik a keretek méretén. Ha a keret túl kicsi, az $a_{i,i}$ értékek túl nagyok lesznek.
- Szegmentálási algoritmus esetében az éles használatkor az adatbázistól eltérő körülmények között (más zaj, stb) elromolhat az A mátrix.

Lehetőségek

- Hierarchikus HMM: fonémák szintje, szavak szintje, nyelvtani szint
- Adaptív HMM: a modell folyamatosan változik használat közben, fenntartva a modell lehetőségekhez mért optimalitását
- Másodrendű, sokadrendű HMM: az állapotátmenetek nem csak a megelőző, de a megelőző n mintától függnek.

HTK (HMM Tool Kit)

- A HTK egy C/C++ könyvtár, ami hangfeldolgozó eljárásokat, és sok beszédfelismerésben hasznos algoritmust tartalmaz
- Többek között:
  - HMM
  - LPC, és egyéb előfeldolgozók
  - Nyelvi modell
  - Fájlformátumok
  - VQ

HTK (HMM Tool Kit)

- Fájlformátumok:
  - Saját formátumok
  - Hangfájlok (WAV, AIFF, AU8 ..)
  - Tesztanyagok cimkézései (TIMIT, SCRIBE ..)
- HMM
  - Adatszerkezetek modellekhez, akár többedrendűekhez
  - Többféle tanítási módszer
  - Viterbi/Forward és egyéb kiértékelések
  - Adaptív HMM

Egy HMM rendszer felépítése