Parameter Estimation with a Hidden Markov Model (HMM)

Lect.dr. Cătălina-Lucia COCIANU
Catedra de Informatică Economică, A.S.E. Bucureşti

*The research reported in the paper focused on the estimation of statistical parameters of HMMs. The parameter specification $\hat{\lambda}$ would, in an important statistical sense, allow the HMM of the given structure to account best for the observed data **Y**. We will arrive at the required algorithms (Baum and Baum-Welch, or forward-backward algorithm) by intuitive reasoning and we will derive the Expectation-Maximization algorithm and use it to justify the convergence and optimality of the Baum-Welch algorithm. To derive the Baum-Welch algorithm for a discrete output alphabet, we will use the formulation of HMMs in which various transactions are deterministically related to the observed outputs.*
Keywords: *Hidden Markov Model (HMM), maximum likelihood estimation, Baum algorithm, Baum-Welch algorithm.*

Introducere

Fie $X_1, X_2, \ldots, X_n, \ldots$ o secvenţă de variabile aleatoare cu valori în $\aleph = \{1,2...,c\}$. În general, pentru orice $n \geq 1$, are loc formula Bayes, $\forall\, i_1, i_2, ..., i_n \in \aleph$,

$$P(X_1 = i_1, X_2 = i_2, ..., X_n = i_n) = \prod_{j=1}^{n} P(X_j = i_j \mid X_1 = i_1, X_2 = i_2, ..., X_{j-1} = i_{j-1}).$$

Variabilele aleatoare $X_1, X_2, \ldots, X_n, \ldots$ formează un *lanţ Markov* dacă, pentru orice $j \geq 1$ şi $\forall\, i_1, i_2, ..., i_j \in \aleph$, $P(X_j = i_j \mid X_k = i_k, 1 \leq k \leq j-1) = P(X_j = i_j \mid X_{j-1} = i_{j-1})$.

Lanţul Markov $X_1, X_2, \ldots, X_n, \ldots$ este un proces cu un număr finit de stări, tranziţiile dintre stări fiind specificate prin intermediul funcţiei p. Modelul de lanţ Markov este observabil, în sensul că, la fiecare moment de timp, ieşirea procesului este o mulţime de stări, fiecare stare fiind corespunzătoare unui eveniment observabil.

Conceptul de *model Markov ascuns* (**H**idden **M**arkov **M**odel) este definit astfel încât stările lanţului Markov generează date observabile, în timp ce secvenţa de stări rămâne ascunsă. Astfel, vor fi definite:

- alfabetul de ieşire $\mathsf{Y} = \{0,1,...,b-1\}$;
- spaţiul stărilor $\aleph = \{1,2...,c\}$ şi s_0 stare iniţială unică;
- distribuţia de probabilitate a tranziţiilor dintre stări, $p(s' \mid s)$; procesul determinat de mulţimea stărilor şi probabilitatea tranziţiilor între stări este un lanţ Markov;
- probabilitatea de distribuţie a ieşirilor, asociată fiecărei tranziţii de la o starea *s* la o starea *s'*, $q(y \mid s, s')$, pentru orice $s, s' \in \aleph$; procesul astfel obţinut este un lanţ Markov.

Pentru utilizarea HMM ca model de generare de date este necesară cunoaşterea probabilităţii de tranziţie între stări, a probabilităţii obţinerii fiecărui simbol din alfabetul de ieşire, precum şi a structurii tranziţiilor. Deoarece în cele mai multe situaţii sunt disponibile numai eşantioane de date, problema este de a construi un model astfel încât datele generate de acesta să fie "de acelaşi tip" cu datele observate.

Obţinerea unor estimări "suficient de bune" a structuri interne şi a probabilităţilor care caracterizează un HMM este imposibilă; problema care poate fi rezolvată este estimarea parametrilor modelului pe baza unei structuri a tranziţiilor considerată cunoscută.

Parametrii unui HMM vor fi estimaţi pe baza principiului verosimilităţii maxime. Pentru aceasta, fie $P_\lambda(\mathbf{Y})$ probabilitatea ca modelul definit prin parametrii λ să producă simbolurile de ieşirile observate $\mathbf{Y} = (y_1, y_2, ..., y_k)$;

estimarea parametrilor λ revine la rezolvarea problemei variaţionale (1) $\hat{\lambda} = \arg\max_{\lambda} P_{\lambda}(\mathbf{Y})$.

Specificarea parametrilor $\hat{\lambda}$ pe baza relaţiei (1) este realizată exclusiv utilizând datele observate $\mathbf{Y} = (y_1, y_2, ..., y_k)$. Pentru ca estimările $\hat{\lambda}$ să fie suficient de bune (dar nu neapărat optime) este necesar ca datele $\mathbf{Y}$ utilizate pentru învăţare modelului să fie reprezentative pentru proces.

Algoritmul Baum

În continuare vom prezenta algoritmul Baum pentru calculul $\hat{\lambda}$ utilizând exclusiv argumente de natură intuitivă şi empirică. Vom considera următoarea reprezentare a modelului ai cărui parametri sunt estimaţi. Între oricare două perechi de stări *s*, *s'* există cel puţin o tranziţie nenulă *t*, $L(t) = s$ şi $R(t) = s'$ şi cel puţin o tranziţie nulă *t'*, $L(t') = s$ şi $R(t') = s'$. Vom nota cu $q(y \mid t) = q(y \mid s, s')$, unde $L(t) = s$ şi $R(t) = s'$.

Fie $\mathbf{Y} = (y_1, y_2, ..., y_k)$ simbolurile de ieşire observate şi $P\{t^i = t\}$ probabilitatea ca, după *i* etape ale procesului, să fie considerată tranziţia *t* (*t* este cea de-a *i*-a tranziţie nenulă aplicată).

Vom defini contoarele, (2) $c(t) = \sum_{i=0}^{k-1} P\{t^i = t\}$, pentru toate tranziţiile *t* şi (3) $c(y, t') = \sum_{i=0}^{k-1} P\{t^i = t'\}\delta(y_{i+1}, y)$, pentru toate tranziţiile nenule *t'* , unde δ este funcţia Kronecker.

Pentru fiecare tranziţie *t*, $c(t)$ este numărul de apariţii ale tranziţiei *t* şi, dacă *t* este nenulă, $c(y,t)$ reprezintă numărul de efectuări ale lui *t* cu generarea simbolului de ieşire *y*. Utilizând contoarele definite prin relaţiile (2) şi (3), rezultă că estimaţiile "naturale" ale parametrilor modelului sunt,

$\hat{q}(y \mid t) = \dfrac{c^*(y,t)}{c^*(t)}$, pentru orice tranziţie nenulă *t* şi $\hat{p}(t) = \dfrac{c^*(t)}{\sum_{t':L(t')=L(t)} c^*(t')}$, pentru orice tranziţie *t*, unde $c^*(y,t) = c(y,t)P(y_1, y_2, ..., y_k)$ şi

$$c^*(t) = c(t)P(y_1, y_2, ..., y_k).$$

Estimarea parametrilor modelului pe baza contoarelor definite prin (2) şi (3) este posibilă dacă pot fi calculate probabilităţile

$$P^*\{t^i = t'\} = P\{t^i = t'\}P(y_1, y_2, ..., y_k).$$

În acest scop vom defini,

- $P^*\{t^i = t\}$- probabilitatea producerii şirului de simboluri de ieşire $\mathbf{Y} = (y_1, y_2, ..., y_k)$ şi astfel încât, după *i* etape ale procesului, să fie efectuată tranziţia *t*, $i = 0,1,...,k-1$;
- $\alpha_i(s) = P\{s^i = s\}$ - probabilitatea generării şirului $y_1, y_2, ..., y_i$ şi starea atinsă la etapa *i* este *s*, $i = 0,1,...,k$;
- $\beta_i(s) = P\{rest \mid s^i = s\}$- probabilitatea generării şirului $y_{i+1}, ..., y_k$, dacă starea atinsă la etapa *i* este *s*, $i = 0,1,...,k$.

Deoarece tranziţia *t* poate fi considerată după ce modelul a atins starea $L(t)$ şi, după efectuarea lui *t* procesul continuă începând cu starea $R(t)$, rezultă (4)

$$P^*\{t^i = t\} = \begin{cases} \alpha_i(L(t))p(t)q(y_{i+1} \mid t)\beta_{i+1}(R(t)), & t \text{ nenulă} \\ \alpha_i(L(t))p(t)\beta_i(R(t)), & t \text{ nulă} \end{cases}$$

De asemenea, similar definiţiei recursive (18), obţinem,

$$\alpha_i(s) = \sum_{t \in \overline{N}(s)} \alpha_{i-1}(L(t))p(t)q(y_i \mid t) + \sum_{t \in N(s)} \alpha_i(L(t))p(t) \quad (5)$$

unde *N*(*s*) este mulţimea tuturor tranziţiilor nule *t* cu *R*(*t*)=*s* şi având starea iniţială mai mică decât *s* ($L(t) < s$) şi $\overline{N}(s)$ este mulţimea tuturor tranziţiilor nenule *t* cu *R*(*t*)=*s*.

Cu toate că relaţia (5) include $\alpha_i(s)$ în ambii membri ai egalităţii, poate fi evaluată în condiţiile în care modelul nu include tranziţii nule care să formeze cicluri prin aplicarea unui algoritm de ordonare. Relaţia (5) poartă numele de *recursie înainte (forward)*.

Valorile $\beta_i(s)$ pot fi calculate recursiv prin,

$$\beta_i(s) = \sum_{t \in \overline{M}(s)} p(t) q(y_{i+1} \mid t) \beta_{i+1}(R(t)) + \sum_{t \in M(s)} p(t) \beta_i(R(t)) \quad (6)$$ unde,

$M(s) = \{t : L(t) = s, t \text{ este nulă }, R(t) > s\}$ şi

$\overline{M}(s) = \{t : L(t) = s, t \text{ este nenulă }\}$.

Relaţia (6) este referită drept *recursie înapoi (backward)*.

Algoritmul Baum [6] pentru estimarea parametrilor modelului este construit pe baza relaţiilor (5) şi (6), care permit evaluarea probabilităţilor $P^*\{t^i = t\}$ prin (4).

Pas 1. Pasul înainte: pentru $i = 0,1,...,k$ calculează $\alpha_i(s)$ prin (5), pentru orice stare s a modelului; este utilizată iniţializarea $\alpha_0(s_0) = 1$.

Pas 2. Pasul înapoi: pentru $i = k-1,...,0$ calculează $\beta_i(s)$ prin (6), pentru orice stare s a modelului; sunt folosite iniţializările $\beta_k(s) = 1$, pentru orice $s \in \aleph$.

Pas 3. Evaluarea probabilităţilor de tranziţie: pentru $i = k-1,...,0$ şi orice tranziţie t, calculează $P^*\{t^i = t\}$ utilizând relaţia (4).

Pas 4. Estimarea parametrilor:

Calculează contoarele $c^*(t) = \sum_{i=0}^{k-1} P^*\{t^i = t\}$, pentru toate tranziţiile t şi $c^*(y,t') = \sum_{i=0}^{k-1} P^*\{t^i = t'\}\delta(y_{i+1}, y)$, pentru toate tranziţiile nenule t'.

Calculează estimatorii, (7) $\hat{q}(y \mid t) = \dfrac{c^*(y,t)}{c^*(t)}$, pentru orice tranziţie nenulă t şi (8) $\hat{p}(t) = \dfrac{c^*(t)}{\sum_{t':L(t')=L(t)} c^*(t')}$, pentru orice tranziţie t.

În cadrul algoritmului Baum au fost utilizate valorile $p(t)$ şi $q(y \mid t)$ în aplicarea relaţiilor (4), (5) şi (6), deci tocmai a parametrilor care sunt estimaţi. În scopul evitării acestui inconvenient, algoritmul pentru estimarea parametrilor HMM poate fi definit astfel [6]:

Pas 1. Setează arbitrar parametrii $p(t)$ şi $q(y \mid t)$; alege $n \in \mathbf{N}$, numărul de iteraţii.

Pentru $l = 1,2,...,n$, repetă

Pas 2. Aplică algoritmul Baum pentru calculul $\hat{p}(t)$ şi $\hat{q}(y \mid t)$ utilizând ultimele valori calculate pentru $p(t)$ şi $q(y \mid t)$.

Observaţii.

1. Algoritmul Baum determină obţinerea unor estimatori $\hat{\lambda}$ care nu îndeplinesc neapărat (1); estimatorii calculaţi de algoritmul Baum sunt maxime locale: fie $P_\lambda(\mathbf{Y})$ probabilitatea ca modelul definit prin parametrii λ să producă simbolurile de ieşirile observate $\mathbf{Y} = (y_1, y_2, ..., y_k)$, parametrii λ' parametrii calculaţi prin (7) şi (8) şi λ parametrii utilizaţi în aplicarea relaţiilor (4), (5) şi (6). Obţinem că $P_{\lambda'}(\mathbf{Y}) > P_\lambda(\mathbf{Y})$.

2. În multe situaţii practice, nu poate fi cunoscut sau estimat numărul de paşi, *n*, necesar obţinerii estimatorilor $\hat{p}(t)$ şi $\hat{q}(y \mid t)$. O variantă utilizată în implementarea algoritmului prezentat mai sus, este `repetă pasul 2 cât timp modificările asupra` $\hat{p}(t)$ `şi` $\hat{q}(y \mid t)$ `sunt superioare unei valori date,` ε.

Normalizarea în algoritmul Baum

În multe situaţii practice în care este aplicat algoritmului Baum, calculul recursiv al valorilor $\alpha_i(s)$ şi $\beta_i(s)$ prin relaţiile (5) şi respectiv (6) presupune un număr mare de etape *k*. În consecinţă, precizia cu care sunt calculate $\alpha_i(s)$ şi $\beta_i(s)$ este mică pe măsură ce *i* creşte (în cazul $\alpha_i(s)$), respectiv descreşte (în cazul $\beta_i(s)$). În scopul evitării pierderii de precizie prin variaţia lui *i*, pot fi aplicate normalizări în procesul recursiv de calcul al valorilor $\alpha_i(s)$ şi $\beta_i(s)$.

Fie $Q_1, Q_2, ..., Q_k$ numere pozitive şi $\alpha_i^*(s) = \dfrac{\alpha_i(s)}{\prod_{j=0}^{i} Q_j}$. Relaţia (5) implică,

$$\alpha'_i(s) = \alpha_i^*(s) Q_i = \sum_{t \in \overline{N}(s)} \alpha_{i-1}^*(L(t)) p(t) q(y_i \mid t) + \sum_{t \in N(s)} \alpha'_i(L(t)) p(t) \qquad (9)$$

Dacă vom considera

(10) $Q_0 = 1$ şi $\forall 1 \le i \le k,\quad Q_i = \sum_s \alpha'_i(s)$, rezultă că $\forall 1 \le i \le k,\quad \sum_s \alpha^*_i(s) = 1$ şi pasul înainte din algoritmul Baum este descris astfel [6].

1. Calculează valorile $\alpha'_i(s)$ conform ordonării stărilor, utilizând valorile normalizate $\alpha^*_{i-1}(s)$.
2. Calculează $Q_i = \sum_s \alpha'_i(s)$
3. Aplică normalizarea $\alpha^*_i(s) = \dfrac{\alpha'_i(s)}{Q_i}$ necesară calculului valorilor $\alpha'_{i+1}(s)$.

În scopul normalizării la pasul înapoi al algoritmului Baum sunt folosiţi aceiaşi factori $Q_1, Q_2, ..., Q_k$, $\beta^*_i(s) = \dfrac{\beta_i(s)}{\prod_{j=i}^{k} Q_j}$. Rezultă,

$$\beta'_i(s) = \beta^*_i(s) Q_i = \sum_{t \in \overline{M}(s)} p(t) q(y_{i+1} \mid t) \beta^*_{i+1}(R(t)) + \sum_{t \in M(s)} p(t) \beta'_i(R(t)) \qquad (11)$$

Pasul înapoi al algoritmului Baum este modificat conform relaţiei (11) similar pasului înainte prezentat mai sus.

Observaţii

1. Deoarece $\forall 1 \le i \le k,\quad \sum_s \alpha^*_i(s) = 1$, din relaţia (10) rezultă,

$$P(y_1, y_2, ..., y_k) = \sum_s \alpha_k(s) = \sum_s \left[\alpha^*_k(s) \prod_{i=1}^{k} Q_i \right] = \prod_{i=1}^{k} Q_i \; 2.$$

Utilizând definiţiile valorilor $\alpha^*_i(s)$ şi $\beta^*_i(s)$, obţinem,

$$\alpha_i(s) \beta_{i+1}(s) = \alpha^*_i(s) \beta^*_{i+1}(s) \prod_{j=1}^{k} Q_j = \alpha^*_i(s) \beta^*_{i+1}(s) P(y_1, y_2, ..., y_k)$$

3. Din $P^*\{t^i = t'\} = P\{t^i = t'\} P(y_1, y_2, ..., y_k)$, rezultă că probabilitatea $P\{t^i = t\}$ poate fi calculată similar relaţiei (4) prin, $P\{t^i = t\} = \begin{cases} \alpha^*_i(L(t)) p(t) q(y_{i+1} \mid t) \beta^*_{i+1}(R(t)), & t \text{ nenulă} \\ \alpha^*_i(L(t)) p(t) \beta^*_i(R(t)) Q_i, & t \text{ nulă} \end{cases}$

şi este utilizată în evaluarea contoarelor $c(t)$ şi $c(y,t)$ conform (2) şi (3), evitând astfel calcularea explicită a valorii $\prod_{i=1}^{k} Q_i$.

Teorema EM. Algoritmul de maximizare a mediei

În continuarea este prezentată teorema de maximizare a mediei, utilizată pentru construcţia şi demonstrarea convergenţei şi optimalităţii algoritmului Baum-Welch [6]. Teorema EM este bazată pe inegalitatea Jensen: pentru orice distribuţii discrete de probabilitate, $p(x)$ şi $q(x)$,

(12) $\sum_x p(x) \log_b p(x) \ge \sum_x p(x) \log_b q(x)$.

Fie y data observabilă, $P_{\theta'}(y)$ distribuţia de probabilitate a lui y în cadrul unui model caracterizat de parametrii θ' (θ' reprezintă totalitatea parametrilor ale căror valori intervin în specificarea distribuţiei $P_{\theta'}$) şi $P_\theta(y)$ distribuţia de probabilitate corespunzătoare setului de parametri θ. Problema este de a determina condiţiile în care apariţia datei y este mai probabilă în modelul caracterizat de θ comparativ cu modelul specificat prin θ' (în acest caz, θ este o îmbunătăţire a lui θ').

Fie t o variabilă aleatoare cu valori determinate în cadrul aceluiaşi proces care a determinat generarea datei y; t este deci guvernată de valorile parametrilor θ', respectiv θ (de exemplu, în cazul unui HMM, y corespunde unei secvenţe de ieşire observate, iar t poate reprezenta o mulţime de stări sau o secvenţă de tranziţii).

Deoarece $\sum_t P_{\theta'}(t \mid y) = 1$, rezultă,

$$\log_b P_\theta(y) - \log_b P_{\theta'}(y) = \sum_t P_{\theta'}(t \mid y)\log_b P_\theta(y) - \sum_t P_{\theta'}(t \mid y)\log_b P_{\theta'}(y).$$

În continuare, obținem,

$$\log_b P_\theta(y) - \log_b P_{\theta'}(y) = \sum_t P_{\theta'}(t \mid y)\log_b\left(P_\theta(y)\frac{P_\theta(t,y)}{P_\theta(t,y)}\right) - \sum_t P_{\theta'}(t \mid y)\log_b\left(P_{\theta'}(y)\frac{P_{\theta'}(t,y)}{P_{\theta'}(t,y)}\right) =$$
$$= \sum_t P_{\theta'}(t \mid y)\log_b \frac{P_\theta(t,y)}{P_\theta(t \mid y)} - \sum_t P_{\theta'}(t \mid y)\log_b \frac{P_{\theta'}(t,y)}{P_{\theta'}(t \mid y)} =$$
$$= \sum_t P_{\theta'}(t \mid y)\log_b P_\theta(t,y) - \sum_t P_{\theta'}(t \mid y)\log_b P_{\theta'}(t,y) +$$
$$+ \sum_t P_{\theta'}(t \mid y)\log_b P_\theta(t \mid y) - \sum_t P_{\theta'}(t \mid y)\log_b P_\theta(t \mid y) \geq$$
$$\geq \sum_t P_{\theta'}(t \mid y)\log_b P_\theta(t,y) - \sum_t P_{\theta'}(t \mid y)\log_b P_{\theta'}(t,y)$$

inegalitatea rezultând din (12). Obținem următoarea teoremă.

Teorema EM (maximizarea mediei)

Dacă

(13) $\sum_t P_{\theta'}(t \mid y)\log_b P_\theta(t,y) > \sum_t P_{\theta'}(t \mid y)\log_b P_{\theta'}(t,y)$,

rezultă (14) $P_\theta(y) > P_{\theta'}(y)$

Teorema de maximizare a mediei are următoarea interpretare. Dacă inițial parametrii modelului sunt setați pe θ' şi sunt selectați parametrii θ astfel încât relația (13) să fie îndeplinită, atunci data y este observată cu o probabilitate mai mare în modelul caracterizat de parametrii θ comparativ cu modelul specificat prin θ'.

În scopul aplicării optime a teoremei EM, este necesară determinarea unei instanțe θ care să maximizeze membrul stâng al inegalității (13). Teorema este numită *maximizarea mediei* pentru că, media variabilei aleatoare $\log_b P_{\theta'}(t,y)$ în raport cu distribuția inițială $P_{\theta'}(t \mid y)$ este maximizată prin considerarea ei ca funcție de argument θ,

$$\sum_t P_{\theta'}(t \mid y)\log_b P_\theta(t,y) > \sum_t P_{\theta'}(t \mid y)\log_b P_{\theta'}(t,y).$$

Pe baza teoremei EM este obținut algoritmul de maximizare a mediei, astfel [6].

Pas1. Selectează θ', valorile inițiale ale parametrilor modelului.

Repetă

Pas2. Calculează θ care maximizează membrul stâng al inegalității (13)

Pas3. $\theta' \leftarrow \theta$

cât timp *C*.

Condiția *C* este prestabilită şi asigură încheierea calculului; specificarea condiției *C* poate fi realizată utilizând următoarea observație. Pe măsură ce algoritmul calculează noi valori ale parametrilor θ', probabilitatea $P_{\theta'}(y)$ creşte până la o anumită valoare, datorită faptului că $P_{\theta'}(y) \leq 1$. Rezultă *C*: la ultima iterație $P_{\theta'}(y)$ şi-a modificat valoarea cu cel puțin ε, unde ε este un parametru fixat.

Observație. Aplicarea algoritmului de maximizare a mediei depinde esențial de alegerea variabilei aleatoare auxiliare t, care trebuie să permită calcularea mediei maxime în membrul stâng al relației (13). În cazul HMM, determinarea unei astfel de variabile este posibilă.

Algoritmul Baum-Welch

Algoritmul Baum-Welch este derivat pe baza teoremei de maximizare a mediei şi este aplicat unui alfabet discret de simboluri de ieşire Y [6]. În continuare vom utiliza definiția unui HMM în care tranzițiile sunt caracterizate în mod determinist de simbolurile de ieşire observate; fiecărei tranziții nenule *t* îi este asociat simbolul de ieşire $Y(t) = y$.

Fie $\mathbf{y} = y_1 y_2 ... y_n$ secvența de simboluri observată şi $\mathbf{t} = t_{i_1 j_1} t_{i_2 j_3} ... t_{i_k j_k}$, $k \geq n$ şirul de tranziții care a generat $\mathbf{y}$; pentru orice $1 \leq l \leq k$, $t_{i_l j_l}$ reprezintă a $j_l - a$ tranziție, în care starea inițială este i_l. Deoarece $\mathbf{t}$ este un drum, $\forall 1 \leq l < k$, $R(t_{i_l j_l}) = i_{l+1}$. Parametrii modelului, θ, reprezintă totalitatea probabilităților de tranziție, $p_{ij} = P(t_{ij})$.

Problema este de a calcula $\max_{\theta} \sum_{\mathbf{t}} P_{\theta'}(\mathbf{t} \mid \mathbf{y}) \ln P_{\theta}(\mathbf{t},\mathbf{y})$. Este utilizată în continuare metoda multiplicatorilor Lagrange.

$$\frac{\partial}{\partial p_{ij}}\left[\sum_{\mathbf{t}} P_{\theta'}(\mathbf{t} \mid \mathbf{y}) \ln P_{\theta}(\mathbf{t},\mathbf{y}) - \sum_{m} \lambda_m \sum_{n} p_{mn}\right] = 0$$

Obținem $\sum_{\mathbf{t}} P_{\theta'}(\mathbf{t} \mid \mathbf{y}) \dfrac{\frac{\partial}{\partial p_{ij}} P_{\theta}(\mathbf{t},\mathbf{y})}{P_{\theta}(\mathbf{t},\mathbf{y})} - \lambda_i = 0$ (15)

Deoarece drumul $\mathbf{t}$ determină secvența $\mathbf{y}$, rezultă că $P_{\theta}(\mathbf{t},\mathbf{y}) = 0$ dacă $\mathbf{y}$ este incompatibil cu $\mathbf{t}$ sau $P_{\theta}(\mathbf{t},\mathbf{y}) = P_{\theta}(\mathbf{t}) = \prod_{l=1}^{k} P(t_{i_l j_l}) = \prod_{l=1}^{k} p_{i_l j_l}$.

Fie $c_{ij}(\mathbf{t})$ numărul de tranziții t_{ij} din drumul $\mathbf{t}$. Pe baza relației $\frac{\partial}{\partial p_{ij}} P_{\theta}(\mathbf{t},\mathbf{y}) = \frac{\partial}{\partial p_{ij}} \prod_{l=1}^{k} p_{i_l j_l}$ obținem $\dfrac{\frac{\partial}{\partial p_{ij}} P_{\theta}(\mathbf{t},\mathbf{y})}{P_{\theta}(\mathbf{t},\mathbf{y})} = \dfrac{c_{ij}(\mathbf{t})}{p_{ij}}$.

Utilizând relația (15), rezultă $\sum_{\mathbf{t}} P_{\theta'}(\mathbf{t} \mid \mathbf{y}) \dfrac{c_{ij}(\mathbf{t})}{p_{ij}} = \lambda_i$, deci

$$p_{ij} = \frac{1}{\lambda_i P_{\theta'}(\mathbf{y})} \sum_{\mathbf{t}} P_{\theta'}(\mathbf{t},\mathbf{y}) c_{ij}(\mathbf{t}) = \frac{1}{K_i} \sum_{\mathbf{t}} P_{\theta'}(\mathbf{t},\mathbf{y}) c_{ij}(\mathbf{t}),$$

unde K_i este constanta de normalizare $\left(\sum_{j} p_{ij} = 1\right)$. Deoarece $\mathbf{t} = t_{i_1 j_1} t_{i_2 j_3} \ldots t_{i_k j_k}$, $c_{ij}(\mathbf{t})$ poate fi evaluată astfel $c_{ij}(\mathbf{t}) = \sum_{l=1}^{k} \delta(t_{ij}, t_{i_l j_l})$.

Pentru toate tranzițiile nenule t_{ij}, are loc şirul de egalități,

$$\sum_{\mathbf{t}} P_{\theta'}(\mathbf{t},\mathbf{y}) c_{ij}(\mathbf{t}) = \sum_{l=1}^{k} \sum_{\mathbf{t}} P_{\theta'}(\mathbf{t},\mathbf{y}) c_{ij}(t_{ij} t_{i_l j_l}) =$$

$$= \sum_{l=1}^{k} P_{\theta'}(y_1, y_2, \ldots, y_{l-1}, s_{l-1} = i) p'_{ij}\, P_{\theta'}(y_l, y_{l+1}, \ldots, y_n \mid s_l = R(t_{ij})), \text{ deci}$$

$$\sum_{\mathbf{t}} P_{\theta'}(\mathbf{t},\mathbf{y}) c_{ij}(\mathbf{t}) = \sum_{l=1}^{k} \alpha_{l-1}^{\theta'}(i) p'_{ij}\, \beta_l^{\theta'}(R(t_{ij})), \quad (16)$$

unde, $\alpha_l^{\theta'}(s) = P_{\theta'}(y_1, y_2, \ldots, y_l, s_l = s)$ şi $\beta_l^{\theta'}(s) = P_{\theta'}(y_{l+1}, y_{l+2}, \ldots, y_k \mid s_l = s)$.

Pentru toate tranzițiile nule t_{ij}, obținem,

$$\sum_{\mathbf{t}} P_{\theta'}(\mathbf{t},\mathbf{y}) c_{ij}(\mathbf{t}) = \sum_{l=1}^{k} \sum_{\mathbf{t}} P_{\theta'}(\mathbf{t},\mathbf{y}) c_{ij}(t_{ij} t_{i_l j_l}) =$$

$$= \sum_{l=1}^{k} P_{\theta'}(y_1, y_2, \ldots, y_{l-1}, s_{l-1} = i) p'_{ij}\, P_{\theta'}(y_l, y_{l+1}, \ldots, y_n \mid s_{l-1} = R(t_{ij}))$$

Şi $\sum_{\mathbf{t}} P_{\theta'}(\mathbf{t},\mathbf{y}) c_{ij}(\mathbf{t}) = \sum_{l=1}^{k} \alpha_{l-1}^{\theta'}(i) p'_{ij}\, \beta_{l-1}^{\theta'}(R(t_{ij}))$ (17)

În relațiile (16) şi (17), fiecare membru drept este contribuția adusă de algoritmul Baum-Welch în calculul valorilor p_{ij} la următorul pas. Valorile $\alpha_l^{\theta'}(s)$ şi $\beta_l^{\theta'}(s)$ sunt obținute iterativ pe baza relațiilor (5) şi (6).

Bibliografie

[1] Bishop C., "Neural Networks for Pattern Recognition", Oxford University Press, 1996
[2] Boulard H., Morgan N., "Connectionist Speech Recognition: A Hybrid Approach", Kluwer Academic Press, 1994
[3] Devroye L., Gyorfi L., Lugosi G., "A Probabilistic Theory of Pattern Recognition", Springer Verlag, 1996
[4] Gold B., Morgan N., "Speech Audio Signal Processing", John Wiley & Sons, 2000
[5] Husmeier D., "Learning Non-stationary Conditional Probability Distributions", Neural Networks, Vol. 13, 2000
[6] Jelinek F., "Statistical Methods for Speech Recognition", MIT Press, 1997
[7] Morgan D.P., "Neural Networks and Speech Processing", Kluwer Academic Publishers, 1991
[8] Juang B.-H., Rabiner L., "Issue in Using Hidden Markov Models for Speech Recognition", Advances in Signal Proces-sing, 1992
[9] Stewart W., "Introduction to the Numerical Solutions of Markov Chains", Princeton University Press, 1994
[10] Young S.J., "Competitive Training in Hidden Markov Models", Proceedings of IEEE Int. Conf. on Acoustic, Speech and Signal Processing", 1990