Reconeixement de paraules manuscrites amb HMMs amb mixtures de Bernoulli i de gaussianes als estats

UNIVERSITAT POLITÈCNICA DE VALÈNCIA

Tesi de Màster
realitzada per Adrià Giménez Pastor
dirigida per Alfons Juan i Císcar

13 de novembre de 2008

Pròleg

El camp del reconeixement de text manuscrit *off-line* (RTM) té una llarga història i continua tenint gran interès degut a la seva dificultat. A diferència del reconeixement *on-line*, on l'escriptura és adquirida com una senyal en funció del temps mitjançant una llapissera electrònica o un ratolí, en el reconeixement *off-line* l'adquisició es fa escanejant documents i obtenint com a resultat imatges, normalment en escala de grisos. Tradicionalment s'ha vingut considerant el reconeixement *off-line* com un problema més difícil que el *on-line*. Hi ha que tindre en compte que a diferència del text no manuscrit, on cada caràcter s'escriu sempre igual i els caràcters es troben ben alineats entre ells, en el text manuscrit podem trobar diferents estils d'escriptura, inclinacions (tant a nivell de lletra com de línia), grossors de traç, etc. Inicialment el RTM es va centrar en el reconeixement de caràcters aïllats [19], més tard el reconeixement de paraules aïllades [18, 2] va començar a captar l'atenció, i hui en dia el reconeixement de línies de text o paràgrafs és l'interès principal en RTM [21, 20, 29, 9].

Els models ocults de Markov (HMM) han jugat un paper predominant en l'àmbit del RTM en els últims anys [14, 20, 6, 27]. Importats del reconeixement automàtic de la parla [15], són emprats per a modelar la probabilitat (densitat) d'una seqüència d'observacions, donada la seua transcripció o simplement una etiqueta de classe. Seqüència d'observacions que s'ha obtingut extraguent característiques al llarg de la imatge, mitjançant una finestra de mostreig. Uns dels principals motius del seu èxit és la inclusió de la segmentació en el model, i la facilitat per a utilitzar-los conjuntament amb models basats en estats finits. Tradicionalment en el reconeixement de text s'ha distingit entre tècniques de segmentació explícita i implícita. Les tècniques explícites requereixen d'una segmentació prèvia de les imatges en caràcters o parts de caràcters. Aquestes tècniques han funcionant bé en el reconeixement de text no manuscrit, però al passar a text manuscrit, on és molt comú que les lletres s'ajunten unes amb altres, les tècniques de segmentació implícita i més concretament els HMM, han cobrat protagonisme.

Típicament s'han vingut emprant dos tipus de HMMs: HMMs discrets, que treballen amb seqüències de símbols, resultat de discretitzar prèviament els vectors d'observacions, i HMMs continus amb (mixtures de) gaussianes als estats. Aquestos HMMs treballen amb els vectors de característiques, que solen ser vectors de reals, directament. No obstant, no hi ha en la literatura un conjunt de característiques que s'haja establert com l'estàndard.

En el treball realitzat en el grup PRHLT (`http://prhlt.iti.es`), marc de treball en el que s'ha realitzat aquesta Tesi de Màster, s'han utilitzat tradicionalment vectors de 60 dimensions, cadascun compost de 20 valors de gris i 40 derivades [20]. En [6], sols 9 característiques són calculades: 3 capturant informació global de la finestra de mostreig i 6 amb informació addicional sobre l'escriptura. Mentres que en [27] característiques discretes i continues són combinades. En un treball previ ja vam proposar la idea d'interpretar les columnes, prèvia binarització, directament com els vectors de característiques [4]. Açò ens permetia per una banda no perdre informació al extraure els vectors de característiques. I per una altra banda utilitzar models que tracten més directament amb els objectes a reconèixer. Concretament proposarem substituir les gaussianes per bernoullis multidimensionals, ja que els vectors de característiques són binaris. En aquest treball avancem en eixa direcció expandint el model a mixtures de Bernoulli. A més hem passat de treballar amb HMM simples, que modelaven paraules senceres, a HMM segmentats, que permeten abordar tasques més complexes. Aquest nou pas el fem tenint en ment la idea d'acabar emprant models que aprenguen transformacions geomètriques, tal i com ja vam demostrar amb caràcters aïllats emprant bernoullis amb modelat explícit de transformacions [16].

Fins no fa molts anys, casi tots els sistemes de RTM que es podien trobar en la literatura estaven restringits a dominis molt concrets: reconeixement de xifres en xecs bancaris, d'adreces postals, etc. Actualment açò ha canviat, i s'està treballant amb corpus de text manuscrit amb vocabularis grans i diversos estils d'escriptura: corpus d'escriptura espontània, documents antics, etc. Un dels corpus que més rellevància ha obtingut en aquest àmbit és el corpus IAM. IAM és un corpus de text anglès manuscrit que, entre altres, conté un subcorpus de paraules aïllades i un de frases. Per a realitzar aquest treball hem seleccionat el subcorpus de paraules, fent per tant reconeixement de paraules manuscrites aïllades. Aquesta decisió s'ha pres amb l'objectiu d'estudiar amb deteniment el model de bernoullis enfrontant-lo a un model ben conegut, com ho és el que s'ha vingut gastant al grup PRHLT. Açò ens permet per una banda eliminar tota influència del model de llenguatge. I per una altra reduir el cost computacional que suposa treballar a nivell de frase o línia, permetent una experimentació més amplia. A més la majoria dels resultats obtinguts a nivell de paraula són exportables a nivell de línia o frase. Amb la intenció de que aquest treball puga ser un marc per a provar futurs models basats en HMMs, hem tingut cura de que els experiments puguen ser reproduïts.

El treball queda doncs estructura de la següent manera: en el següent capítol s'introdueixen els conceptes teòrics en els que es fonamenta aquest treball. En el capítol 2 s'explica l'estructura d'un sistema de reconeixement de paraules basat en HMMs, i diferents alternatives d'implementació, tant per al sistema del PRHLT com per a la nova proposta. El capítol 3 presenta el corpus IAM en detall, el subcorpus de paraules i la partició creada per a l'experimentació. Experimentació que s'explica en detall en el capítol 4, que en resum és una comparativa del sistema del PRHLT amb el de mixtures de Bernou-

lli, provant diferents configuracions. Finalment en el capítol 5 trobem les conclusions i el treball futur.

Capítol 1

Fonaments teòrics

1.1 Introducció

En aquest capítol s'introdueixen els conceptes teòrics en els que es basa aquest treball. En particular es parla sobre models ocults de Markov i la formulació per a entrenar i reconèixer. El capítol s'estructura de la següent manera: en la secció 1.2 s'explica el concepte de model ocult de Markov. Aquest concepte es estés a model ocult de Markov segmentat en la secció 1.3, a més de la formulació per a entrenar i reconèixer. Finalment les seccions 1.4 i 1.5 parlen respectivament del cas particular de models ocults de Markov amb mixtures de gaussianes i mixtures de Bernoulli.

1.2 Models ocults de Markov (HMMs)

Un model ocult de Markov (HMM) modela la probabilitat o densitat de probabilitat d'una seqüència finita d'observacions. Els HMMs es basen en la suposició de que existeix una màquina d'estats finits estocàstica coneguda, dins de la qual, cada vegada que es transita a un estat, aquest genera una observació amb una determinada probabilitat. Per tant sabem que la seqüència d'observacions ha sigut generada per la màquina d'estats, però desconeguem la seqüència d'estats concreta que l'ha generada.

Més formalment un HMM queda definit per:

1. Un conjunt finit d'estats $\mathcal{Q} = \{1, \ldots, Q\}$.
2. Els estats especials I (inicial) i F (final), que no emeten observacions.
3. Una funció de probabilitat (densitat de probabilitat) $p(x, \vec{\Theta}')$ definida en el domini de les observacions.
4. Els paràmetres de p $\{\vec{\Theta}'_1, \ldots, \vec{\Theta}'_Q\}$, on $\vec{\Theta}'_q$ son els paràmetres associats a l'estat q.

5. Un conjunt de paràmetres $\{\pi_{qq'} \mid q, q' \in \mathcal{Q} \cup \{I, F\}\}$, on $\pi_{qq'}$ és la probabilitat de transitar de l'estat q a l'estat q'. Per definició $\forall_{q \in \mathcal{Q} \cup \{I,F\}} \pi_{qI} = \pi_{Fq} = \pi_{IF} = 0$.

Aleshores la probabilitat d'observar la seqüència $x_1, x_2, \ldots, x_L$ be donada per l'expressió:

$$P(x_1, x_2, \ldots, x_L) = \sum_{I, s_1, s_2, \ldots, s_L, F} 1 \cdot P(x_1, \ldots, x_L, s_1, \ldots, s_L, F \mid I) \quad (1.1)$$

$$= \sum_{I, s_1, s_2, \ldots, s_L, F} \pi_{Is_1} \left(\prod_{1<l\leq L} \pi_{s_{l-1}s_l} \right) \pi_{s_L F} \prod_l p(x_l, \vec{\Theta'}_{s_l}). \quad (1.2)$$

Es a dir, el sumatori per a tot possible seqüència d'estats, de la probabilitat d'eixa seqüència d'estats, per la probabilitat de que la seqüència d'observacions siga emesa per eixa seqüència d'estats.

Típicament els HMMs solen emprar-se amb classificadors, modelant la probabilitat d'una seqüència d'observacions donada una classe. En aquest cas es tindria un HMM per cada classe. En la Figura 1.1 es pot veure un exemple gràfic d'un HMM.

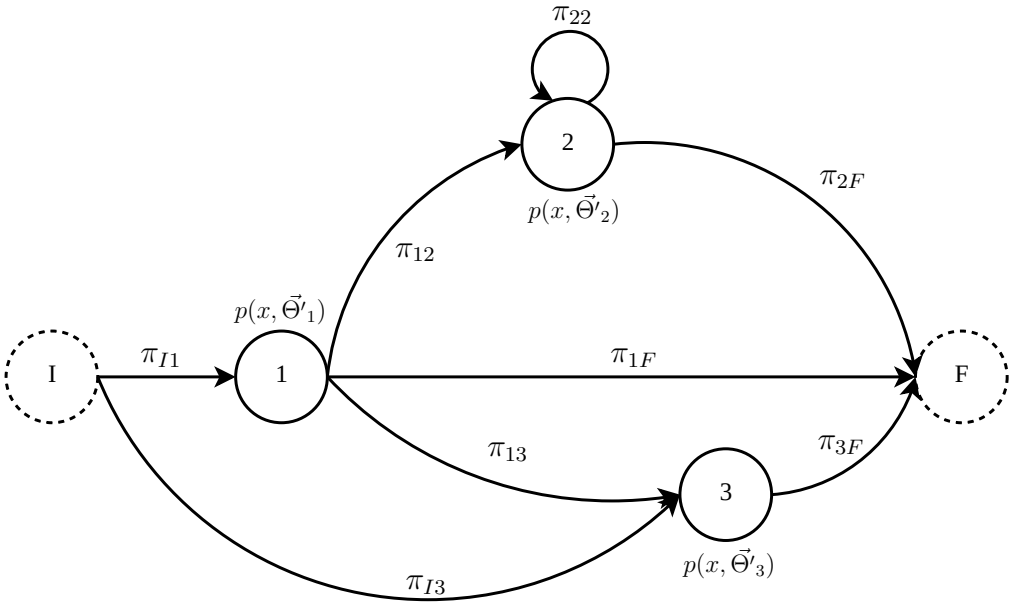


Figura 1.1: Exemple de HMM amb tres estats.

1.3 HMM segmentats

Són comuns el problemes on es modela una probabilitat condicionada $p(x_1, \ldots, x_L \mid c)$ a una classe, i aquesta es pot representar com una seqüència de símbols $y_1, \ldots, y_T$ d'un

alfabet Σ donat. Quan el nombre de classes és molt gran, tindre un HMM per a cada classe esdevé inviable. És en aquestos casos on es solen emprar HMM segmentats. En un HMM segmentat es té un HMM per cada símbol de Σ, i per a cada classe es construeix un HMM virtual concatenant els HMMs dels símbols que formen la classe. Un HMM es concatena amb el següent HMM fusionant l'estat final del primer HMM, amb l'estat inicial del segon HMM.

Més formalment un HMM segmentat queda definit per:

1. Un conjunt finit de símbols $\Sigma = \{1, \ldots, C\}$.
2. Un conjunt de HMMs $\{\mathcal{H}_1, \ldots, \mathcal{H}_C\}$.

Aleshores la probabilitat d'observar la seqüència $x_1, \ldots, x_L$ donada la classe $y_1, \ldots, y_T$, on $T \leq L$, ve donada per la següent expressió:

$$P(x_1, \ldots, x_L \mid y_1, \ldots, y_T) = \sum_{S_0, s_1, \ldots, s_{j(t)}, S_t, s_{i(t+1)}, \ldots, s_L, S_T} 1 \cdot \prod_t P(x_{i(t)}, \ldots, x_{j(t)}, s_{i(t)}, \ldots, s_{j(t)}, S_t \mid \mathcal{H}_{y_t}, S_{t-1}), \tag{1.3}$$

on s_i és l'estat on s'ha emés la i-èsima observació, $i(t)$ i $j(t)$ són respectivament les posicions de la primera i última observacions emeses per $\mathcal{H}_{y_t}$ ($i(1) = 1$ i $j(T) = L$), i S_t és tant l'estat I_{t+1} com l'estat F_t ($S_0 = I_1$).

En la Figura 1.2 es pot veure un exemple de HMM virtual d'un HMM segmentat.

1.3.1 Entrenament mitjançant *Baum-Welch*

Un dels algorismes més utilitzats per a l'entrenament de HMM (segmentats i no segmentats) és l'algorisme *Baum-Welch* [1]. Aquest algorisme és un cas particular, per a entrenament de HMM, de l'algorisme *Expectation-Maximization* (EM) [3]. L'algorisme EM es sol emprar per a entrenar mitjançant màxima versemblança amb dades ocultes. En els HMM les dades ocultes es corresponen amb les seqüències d'estats. És un algorisme iteratiu, on cada iteració té dos passos. El pas E (*Expectation*), on es calcula el valor esperat de les dades ocultes, i el pas M (*Maximization*), on es calculen per màxima versemblança, fent ús dels valors del pas E, els paràmetres del model. Per tant, l'algorisme *Baum-Welch* és iteratiu i cada iteració té dos passos. En el primer pas, que es correspon amb el pas E, es calculen les probabilitats *forward* i *backward* per a cada estat del HMM virtual, i per a cada instant de la seqüència d'observacions. En el segon pas, es correspon amb el pas M, s'actualitzen els paràmetres del model fent ús de les probabilitats *forward* i *backward*.

Més formalment, siguen un conjunt de N mostres, parelles d'observacions i símbols, $\mathcal{S} = \{(x_1, y_1), \ldots, (x_N, y_N)\}$. En la k-èsima iteració les probabilitats de transició entre estats (les d'emissió no es mostren en aquest punt) es calculen com:

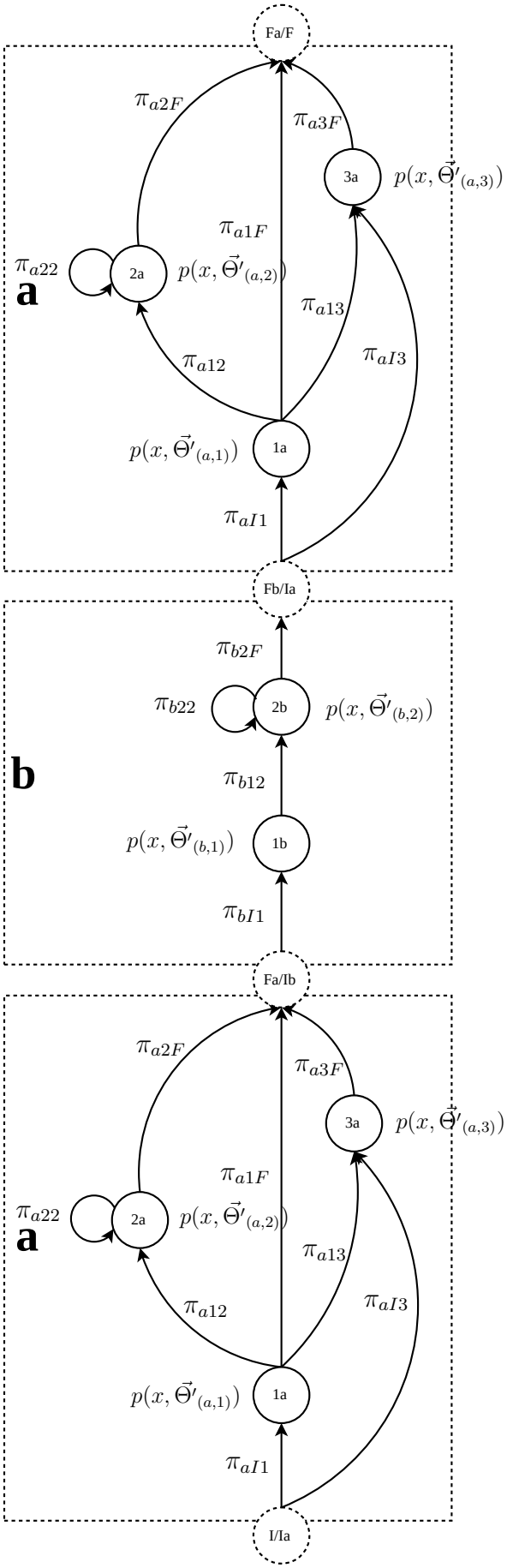


Figura 1.2: Exemple d'un HMM virtual de la seqüència aba per a un HMM segmentat amb $\Sigma = \{a, b\}$.

- Si $q \in \mathcal{Q}_c$

$$\pi_{cq'q}^{(k+1)} = \frac{\sum_n \frac{1}{\alpha_{nL(TF)}^{(k)}} \sum_{t:y_{nt}=c} \sum_l \alpha_{nl-1(tq')}^{(k)} \pi_{cq'q}^{(k)} p(x_{nl}, \vec{\Theta'}_{(c,q)}^{(k)}) \beta_{nl(tq)}^{(k)}}{\sum_n \frac{1}{\alpha_{nL(TF)}^{(k)}} \sum_{t:y_{nt}=c} \sum_l \alpha_{nl-1(tq')}^{(k)} \beta_{nl-1(tq')}^{(k)}} \tag{1.4}$$

- Si $q = F_c$

$$\pi_{cq'F_c}^{(k+1)} = \frac{\sum_n \frac{1}{\alpha_{nL(TF)}^{(k)}} \sum_{t:y_{nt}=c} \sum_l \alpha_{nl(tq')}^{(k)} \pi_{cq'F_c}^{(k)} \beta_{nl(tF)}^{(k)}}{\sum_n \frac{1}{\alpha_{nL(TF)}^{(k)}} \sum_{t:y_{nt}=c} \sum_l \alpha_{nl(tq')}^{(k)} \beta_{nl(tq')}^{(k)}} \tag{1.5}$$

On els valors de les α's (*forward*) i les β's (*backward*) s'han calculat prèviament. $\alpha_{nl(tq)}$ és la probabilitat d'emetre fins a l'observació l-èsima de la mostra n, i acabar en l'estat q del símbol t-èsim, i es calcula com:

- Si $q \in \mathcal{Q}_{y_{nt}}$

$$\alpha_{nl(tq)}^{(k)} = \sum_{q' \in \mathcal{Q}_{y_{nt}} \cup \{I_{y_{nt}}\}} \alpha_{nl-1(tq')}^{(k)} \pi_{y_{nt}q'q}^{(k)} p(x_{nl}, \vec{\Theta'}_{(y_{nt},q)}^{(k)}) \tag{1.6}$$

- Si $q = F_{y_{nt}}$

$$\alpha_{nl(tF)}^{(k)} = \alpha_{nl(t+1I)}^{(k)} = \sum_{q \in \mathcal{Q}_{y_{nt}}} \alpha_{nl(tq)}^{(k)} \pi_{y_{nt}qF_{y_{nt}}}^{(k)} \tag{1.7}$$

- Casos base

$$\alpha_{nl-1(1q)}^{(k)} = \begin{cases} 0 & \text{si} \quad (q \in \mathcal{Q}_{y_{n1}} \wedge l = 1) \vee (q = I_{y_{n1}} \wedge l > 1) \\ 1 & \text{si} \quad q = I_{y_{n1}} \wedge l = 1 \end{cases} \tag{1.8}$$

$\beta_{nl(tq)}$ és la probabilitat d'emetre les observacions de la mostra n-èsima, des de la posició $l+1$ fins l'última, sabent que es parteix de l'estat q del t-èsim símbol, i es calcula com:

- Si $q \in \mathcal{Q}_{y_{nt}} \cup \{I_{y_{nt}}\}$

$$\beta_{nl(tq)}^{(k)} = \pi_{y_{nt}qF_{y_{nt}}}^{(k)} \beta_{nl(tF)}^{(k)} + \sum_{q' \in \mathcal{Q}_{y_{nt}}} \pi_{y_{nt}qq'}^{(k)} p(x_{nl+1}, \vec{\Theta'}_{(y_{nt},q')}^{(k)}) \beta_{nl+1(tq')}^{(k)} \tag{1.9}$$

- Si $q = F_{y_{nt}}$

$$\beta_{nl(tF)}^{(k)} = \beta_{nl(t+1I)}^{(k)} \tag{1.10}$$

- Casos base

$$\beta_{nl(TF)}^{(k)} = \begin{cases} 1 & \text{si} \quad l = L \\ 0 & \text{si} \quad l < L \end{cases} \tag{1.11}$$

$$\forall_{q \in \mathcal{Q}_{y_{nT}} \cup \{F_{y_{nT}}\}} \ \beta_{nL+1(Tq)}^{(k)} = 0 \tag{1.12}$$

1.3.2 Reconeixement mitjançant *Viterbi*

No es pot parlar d'un algorisme de reconeixement per a HMMs segmentats, perquè el reconeixement ve dictat per el model que hàgem definit. Els HMMs segmentats són peces que s'empren en models més complexos per a realitzar diverses tasques de reconeixement. Però pensant en el treball que ens ocupa i que ens trobem en el capítol de teoria, pareix convenient veure com es realitzaria el reconeixement suposant que el model és un classificador basat en un HMM segmentat.

Calcular la probabilitat d'una seqüència d'observacions en un HMM segmentat resulta costós degut al sumatori de tots els camins ocults, per aquest motiu el que es sol fer és simplificar-ho calculant la probabilitat del camí més probable. Realment els dos casos es poden resoldre amb programació dinàmica amb igual cost asintòtic, però en el la pràctica el cost del càlcul del camí màxim és substancialment menor. Formalment, per a una seqüència d'observacions $x_1, \ldots, x_L$ tenim que:

$$\begin{aligned} y^* &= \arg\max_y \; p(y)p(x_1, \ldots, x_L \mid y_1, \ldots, y_T) \\ &= \arg\max_y \; p(y) \sum_{S_0, s_1, \ldots, s_L, S_T} 1 \cdot \prod_t P(x_{i(t)}, \ldots, x_{j(t)}, s_{i(t)}, \ldots, s_{j(t)}, S_t \mid \mathcal{H}_{y_t}, S_{t-1}) \\ &\approx \arg\max_y \; p(y) \max_{S_0, s_1, \ldots, s_L, S_T} \prod_t P(x_{i(t)}, \ldots, x_{j(t)}, s_{i(t)}, \ldots, s_{j(t)}, S_t \mid \mathcal{H}_{y_t}, S_{t-1}) \end{aligned} \tag{1.13}$$

El càlcul del camí màxim és pot realitzar eficientment amb programació dinàmica utilitzant l'algorisme de *Viterbi* [25]. Per tant:

$$\max_{S_0, s_1, \ldots, s_L, S_T} \prod_t P(x_{i(t)}, \ldots, x_{j(t)}, s_{i(t)}, \ldots, s_{j(t)}, S_t \mid \mathcal{H}_{y_t}, S_{t-1}) = \gamma_{L(TF)}, \tag{1.14}$$

on γ es defineix com:

- Si $q \in \mathcal{Q}_{y_{nt}}$

$$\gamma_{l(tq)} = \max_{q' \in \mathcal{Q}_{y_t} \cup \{I_{y_t}\}} \gamma_{l-1(tq')} \pi_{y_t q' q} p(x_l, \vec{\Theta'}_{(y_t, q)}) \tag{1.15}$$

- Si $q = F_{y_t}$

$$\gamma_{l(tF)} = \gamma_{l(t+1I)} = \max_{q \in \mathcal{Q}_{y_t}} \gamma_{l(tq)} \pi_{y_t q F_{y_t}} \tag{1.16}$$

- Casos base

$$\gamma_{l-1(1q)} = \begin{cases} 0 & \text{si} \quad (q \in \mathcal{Q}_{y_1} \wedge l = 1) \vee (q = I_{y_1} \wedge l > 1) \\ 1 & \text{si} \quad q = I_{y_1} \wedge l = 1 \end{cases} \tag{1.17}$$

1.4 HMM segmentat amb mixtures de gaussianes

Les observacions en un HMM segmentat de mixtures de gaussianes de dimensió D pertanyen al domini $\mathbb{R}^D$. Les funcions de densitat de probabilitat són mixtures de gaussianes de dimensió D. La densitat de probabilitat de x en l'estat q del HMM c es defineix com:

$$p(x, \vec{\Theta'}_{(c,q)}) = \sum_{i=1}^{I} p_{(cq)i}\ p(x, \vec{\Theta'}_{(c,q,i)}), \tag{1.18}$$

on I és el nombre de components de la mixtura en l'estat indicat, $p_{(cq)i}$ és la probabilitat a priori de la component i $p(x, \vec{\Theta'}_{(c,q,i)})$ és la densitat de probabilitat d'emissió en eixa component, que es defineix com:

$$p(x, \vec{\Theta'}_{(c,q,i)}) = (2\pi)^{-\frac{D}{2}} |\Sigma_{(cq)i}|^{-\frac{1}{2}} e^{-\frac{1}{2}(x-\mu_{(cq)i})^t \Sigma^{-1}_{(cq)i}(x-\mu_{(cq)i})}, \tag{1.19}$$

on $\mu_{(cq)i}$ i $\Sigma_{(cq)i}$ són respectivament, la mitja i matriu de covariances en eixa component.

Fent ús de l'algorisme de *Baum-Welch* (veure 1.3.1) els coeficients de les mixtures al final de la iteració k s'actualitzen de la següent manera:

$$p^{(k+1)}_{(cq)i} = \frac{\sum_n \frac{1}{\alpha_{nL(TF)}} \sum_{t:y_{nt}=c} \sum_l \alpha^{(k)}_{nl(tq)} z^{(k)}_{nl(tq)i} \beta^{(k)}_{nl(tq)}}{\sum_n \frac{1}{\alpha_{nL(TF)}} \sum_{t:y_{nt}=c} \sum_l \alpha^{(k)}_{nl(tq)} \beta^{(k)}_{nl(tq)}}, \tag{1.20}$$

mentres que els paràmetres de les gaussianes ho fan així:

$$\mu^{(k+1)}_{(cq)i} = \frac{\sum_n \frac{1}{\alpha_{nL(TF)}} \sum_{t:y_{nt}=c} \sum_l \alpha^{(k)}_{nl(tq)} z^{(k)}_{nl(tq)i} x_{nl} \beta^{(k)}_{nl(tq)}}{\sum_n \frac{1}{\alpha_{nL(TF)}} \sum_{t:y_{nt}=c} \sum_l \alpha^{(k)}_{nl(tq)} z^{(k)}_{nl(tq)i} \beta^{(k)}_{nl(tq)}}, \tag{1.21}$$

$$\Sigma^{(k+1)}_{(cq)i} = \frac{\sum_n \frac{1}{\alpha_{nL(TF)}} \sum_{t:y_{nt}=c} \sum_l \alpha^{(k)}_{nl(tq)} z^{(k)}_{nl(tq)i} (x_{nl} - \mu^{(k)}_{(cq)i})(x_{nl} - \mu^{(k)}_{(cq)i})^t \beta^{(k)}_{nl(tq)}}{\sum_n \frac{1}{\alpha_{nL(TF)}} \sum_{t:y_{nt}=c} \sum_l \alpha^{(k)}_{nl(tq)} z^{(k)}_{nl(tq)i} \beta^{(k)}_{nl(tq)}} \tag{1.22}$$

on les z s'han calculat prèviament de la següent manera:

$$z^{(k)}_{nl(tq)i} = \frac{p^{(k)}_{(cq)i}\ p(x_{nl}, \vec{\Theta'}^{(k)}_{(y_{nt},q,i)})}{p(x_{nl}, \vec{\Theta'}^{(k)}_{(y_{nt},q)})}. \tag{1.23}$$

1.5 HMM segmentat amb mixtures de Bernoulli

En un treball previ ja introduirem HMMs amb bernoullis als estats [4]. En aquesta secció ho estenem a HMMs segmentats amb mixtures de Bernoulli. En un HMM segmentat

de mixtures de Bernoulli de dimensió D, les observacions pertanyen al domini $\{0,1\}^D$, i les funcions de probabilitat són mixtures de Bernoulli. La probabilitat d'emetre l'observació x en l'estat q del HMM c, es defineix de forma anàloga a (1.18), on $p(x, \vec{\Theta'}_{(c,q,i)})$ és:

$$p(x, \vec{\Theta'}_{(c,q,i)}) = \prod_{d=1}^{D} p_{(cq)id}^{x_d} (1 - p_{(cq)id})^{1-x_d}, \tag{1.24}$$

on $p_{(cq)id}$ és la probabilitat de emetre en eixa component un 1 en el bit d-èsim. Aquesta última expressió es correspon amb la d'una bernoulli multidimensional de dimensió D. Remarcar que una bernoulli multidimensional de dimensió D no es mes que el productori de la probabilitat de D variables independents entre si, per tant no pot modelar ningun tipus de correlació entre bits.

Fent ús de l'algorisme de *Baum-Welch* (veure 1.3.1) els paràmetres de les bernoullis al final de la iteració k s'actualitzen de la següent manera:

$$p_{(cq)id}^{(k+1)} = \frac{\sum_n \frac{1}{\alpha_{nL(TF)}} \sum_{t:y_{nt}=c} \sum_l \alpha_{nl(tq)}^{(k)} z_{nl(tq)i}^{(k)} x_{nld} \beta_{nl(tq)}^{(k)}}{\sum_n \frac{1}{\alpha_{nL(TF)}} \sum_{t:y_{nt}=c} \sum_l \alpha_{nl(tq)}^{(k)} z_{nl(tq)i}^{(k)} \beta_{nl(tq)}^{(k)}}, \tag{1.25}$$

on les z i els coeficients de les mixtures s'han calculat com en la secció anterior 1.4.

Degut a que les bernoullis treballen amb dades discretes, pot ocórrer que la probabilitat d'un determinat bit de ser 0 siga 1, o viceversa, apareixent el problema de dades no vistes durant l'entrenament. Per aquest motiu una mostra no vista sempre té probabilitat 0, cosa que és poc realista. Per açò al final de cada iteració de *Baum-Welch* les probabilitats de les bernoullis i dels coeficients, són suavitzades mitjançant interpolació lineal amb la probabilitat uniforme:

$$p_{(cq)id} = (1 - \lambda) p_{(cq)id} + \lambda 0.5, \tag{1.26}$$

$$p_{(cq)i} = (1 - \lambda) p_{(cq)i} + \lambda \frac{1}{I}, \tag{1.27}$$

on λ és normalment 10^{-6}.

Capítol 2

Reconeixement de paraules manuscrites

2.1 Introducció

En aquest capítol s'aborda el desenvolupament d'un sistema de reconeixement de paraules manuscrites. El sistema agafa com a entrada una imatge digital en escala de grisos. Es suposa que aquesta imatge conté una paraula manuscrita, que per simplificar, es suposarà d'un vocabulari tancat. Es pressuposa que les paraules s'han escrit d'esquerra a dreta. El sistema que es desenvolupe ha d'estar preparat per a treballar amb tot tipus d'estils d'escriptura (diferents escriptors, grossor traç, inclinacions, grandària text, etc.) i amb vocabularis de gran tamany.

Els HMMs segmentats han sigut emprats amb èxit en el desenvolupament de sistemes de reconeixement de paraules manuscrites, i en general en el reconeixement de text manuscrit [6, 13, 24, 11, 21, 7]. Aquestos sistemes basats en HMMs assumeixen que la imatge es pot interpretar o transformar en una seqüència d'observacions, recorrent-la d'esquerra a dreta. Aquesta seqüència és després classificada amb un classificador basat en un HMM segmentat. L'ús d'un HMM segmentat, front a un classificador de HMMs, s'explica per la grandària del vocabulari. Amb un HMM segmentat cada símbol representa un caràcter, tenint tants HMMs com caràcters, amb un classificador de HMMs es tenen tants HMMs com paraules, amb el perill de no tindre prou mostres per a tants paràmetres. En general aquestos sistemes són molt sensibles a les diferències en l'estil d'escriptura, per aquest motiu compten amb un mòdul de preprocés amb l'objectiu de normalitzar el màxim possible les imatges amb text. En la Figura 2.1 es pot veure un esquema bàsic del sistema, dividit en un mòdul de preprocés i un altre de reconeixement, que inclou l'extracció de característiques i la classificació.

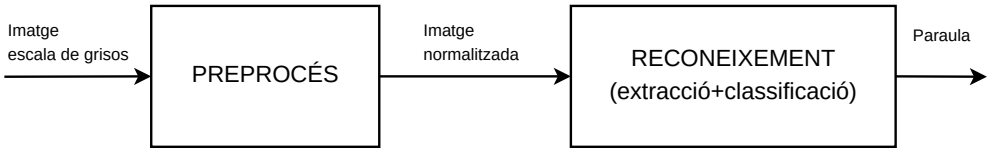


Figura 2.1: Esquema bàsic d'un sistema de reconeixement de paraules manuscrites.

Aquest capítol es divideix en dos seccions. En la primera secció 2.2 s'explica en detall el mòdul de preprocés. En la segona secció 2.3 s'expliquen dos propostes per al mòdul de reconeixement. La primera és ben coneguda [21], es basa en HMMs amb mixtures de gaussianes, mentres que la segon és una proposta original que es basa en HMMs amb mixtures de Bernoulli.

2.2 Preprocés

No existeix en la literatura una tècnica, o conjunt de tècniques, que hagen sigut demostrades com les més adients per a un sistema de reconeixement de paraules. En el que sí que hi ha un cert consens és en el tipus de normalitzacions que hi ha que fer a les imatges: correcció del *slant*, normalització de la grandària, etc. El mòdul de preprocés que utilitzem en aquest treball es basa en la seua totalitat en les tècniques presentades en la tesis del Dr. Moisés Pator i Gadea [7], en particular en les de normalització a nivell de text. El mòdul consta de tres submòduls connectats formant una canonada. El primer submòdul és el de normalització de l'escala de grisos, el segon el de correcció del *slant*, i finalment el submòdul de normalització de la grandària del text. En la Figura 2.2 podem veure un esquema del mòdul.

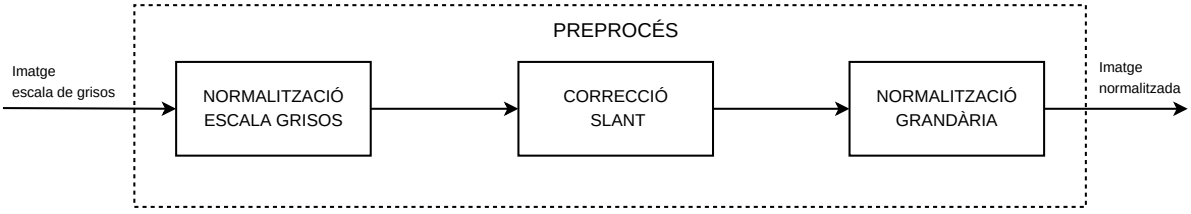


Figura 2.2: Esquema del mòdul de preprocés.

En les següents subseccions s'explica en més detall que fa cadascun dels submòduls, i les possibles tècniques a emprar. Per descomptat un submòdul sempre pot ser eliminat de la canonada.

2.2.1 Normalització de l'escala de grisos

Idealment els traços del text deurien de ser negres i el fons blanc. Al tractar amb imatges en escala de grisos ens podem trobar amb dos fenòmens que poden afectar negativament als següents mòduls de preprocés i reconeixement. Per una banda pot ser que degut a l'adquisició de les imatges, o de com s'ha escrit el text, hi hagen imatges on el text és molt més negre que en altres. Per una altra banda pot aparèixer soroll en el fons, o inclús, un fons que més que blanc és gris clar. Per aquestos motius en aquest mòdul busquem normalitzar l'escala de grisos, per a que els valors de grisos per al fons i per al text siguen semblants en totes les imatges. Estudiem dos possibilitats:

Normalització del contrast: força el píxel més clar a ser blanc, el més obscur a ser negre, i els intermedis són linealment escalats. Nom és fa falta trobar un píxel negre i un altre blanc en la mateixa imatge per a que aquesta normalització no tinga efecte. Per aquest motiu, es sol ficar empíricament un llindar mínim (màxim) per davall (damunt) del qual es força a ser negre (blanc).

Binarització *Otsu*: binaritza la imatge fent ús d'un llindar global. Tots els píxels per davall es forcen a ser negre, i per damunt a ser blanc. Per a cada imatge el llindar es calcula mitjançant el mètode d'*Otsu*, entenent el llindar de binarització com la divisió dels nivells de grisos en dos classes. Aquest mètode busca el llindar que minimitza la variança de grisos dins de la classe, i maximitza la variança de grisos entre classes. Per tant siga L el nombre de nivells de grisos, h l'histograma normalitzat d'una imatge, i siguen:

$$p_1(T) = \sum_{g=0}^{T} h_g, \qquad p_2(T) = \sum_{g=T+1}^{L-1} h_g, \tag{2.1}$$

$$\mu_1(T) = \frac{1}{p_1(T)} \sum_{g=0}^{T} g h_g, \qquad \mu_2(T) = \frac{1}{p_2(T)} \sum_{g=T+1}^{L-1} g h_g, \tag{2.2}$$

aleshores el llindar es calcula com:

$$\hat{T} = \arg\max_{T} p_1(T) p_2(T) (\mu_1(T) - \mu_2(T))^2 \tag{2.3}$$

2.2.2 Correcció del *slant*

El *slant* és l'angle, en sentit horari, de les components verticals dels caràcters respecte a l'eix vertical. La correcció del *slant* pretén que aquest angle siga cero, es a dir, que el text

no estiga inclinat. El procés consta de dos fases, primer es calcula l'angle de *slant* de la paraula. I després es corregeix aplicant una operació de *shear*, que consisteix en traslladar un píxel sobre l'eix x depenent de la seua altura i angle de *slant*. En la Figura 2.3 es pot veure un exemple de correcció del *slant*.

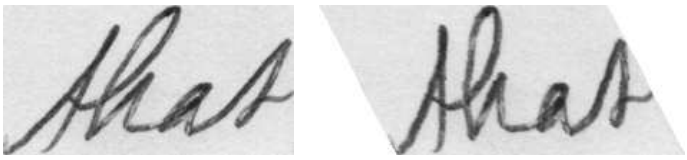


Figura 2.3: A l'esquerra una imatge amb la paraula manuscrita *that*, a la dreta la mateixa imatge amb el *slant* corregit.

El mètode per a calcular el *slant* consisteix en suposar un conjunt de possibles angles de *slant*. Per a cadascun d'aquestos angles es corregeix la imatge original i es calcula la projecció vertical de la imatge corregida, açò és, per a cada columna el sumatori dels valors de gris dels píxels que la composen. Una vegada fet açò l'angle de *slant* és aquell que dels possibles angles la seua projecció maximitza una determinada funció objectiu. Més formalment,

$$\hat{\alpha} = \underset{\alpha \in [45:135]}{\arg\max}\, f(v_\alpha), \tag{2.4}$$

on v_α és la projecció vertical de la imatge original corregida suposant un angle de *slant* α. Aquest mètode es basa en què la projecció vertical d'una imatge sense *slant*, presenta una major variabilitat entre pics i valls respecte al de la mateixa imatge amb *slant*. Contemplem tres possibles funcions objectiu: mètode IDIAP, longitud del contorn i desviació típica.

Mètode IDIAP

Aquesta funció premia l'abundància de traç continu recte respecte a la vertical. Per a poder-se calcular es necessita calcular prèviament, una projecció vertical normalitzada (C_α) per a cada projecció v_α, de la següent manera:

$$C_\alpha(m) = \frac{v_\alpha(m)}{\triangle y_\alpha(m)}, \tag{2.5}$$

on $\triangle y_\alpha(m)$ és la distància entre el primer i últim píxels negres, de la columna m de la imatge corregida per a un slant α. Per tant $C_\alpha(m)$ denota, amb un valor en el rang $[0, 1]$, la quantitat de traç continu en la columna m.

A partir de C_α la funció objectiu es calcula de la següent manera:

$$f(v_\alpha) = \sum_{\{m:C\alpha(m)=1\}} v_\alpha(m)^2. \tag{2.6}$$

Longitud del contorn

Aquesta funció es basa en la idea de que quant més llarg siga la longitud del contorn de la projecció vertical, més gran serà la variabilitat de la projecció.

$$f(v_\alpha) = \sum_{m=1}^{cols-1} |v_\alpha(m) - v_\alpha(m+1)|. \tag{2.7}$$

Desviació típica

S'utilitza com a funció objectiu la desviació típica

$$f(v_\alpha) = \sqrt{\sum_{1\leq m\leq cols} \frac{(\mu - v_\alpha(m))^2}{cols}}, \tag{2.8}$$

on μ es el valor mig dels valors de la projecció vertical.

2.2.3 Normalització de la grandària

En escriptura es distingeixen tres zones (almenys amb caràcters llatins): zona central o cos del text, ascendents i descendents. La majoria de lletres s'escriuen dins de la zona central. Algunes lletres com la *l* o la *t* (la *g* o la *q*), tenen parts que sobreïxen per dalt (per baix) de la zona central, aquestes parts s'anomenen ascendents (descendents).

Amb açò en ment, aquesta normalització té dos objectius. Per una banda alinear les zones verticalment, de tal manera que les línies base i superior del cos siguen efectivament línies rectes, i estiguen en totes les imatges fixades a una mateixa posició relativa a l'altura de la imatge. I també que els ascendents comencen en el primer píxel de la imatge i els descendents acaben en l'últim píxel.

Per una altra banda, depenent de quin tipus d'extracció de característiques fem després, es interessarà donar-li més o menys pes a les diferents zones. Per això es busca reescalar les grandàries de les tres zones en funció dels nostres interessos. Normalment, donat que la majoria de la informació es troba en la zona central, i que dels ascendents i descendents casi importa més la seua presencia que no la seua forma, el que es fa és reduir les zones d'ascendents i de descendents.

La normalització de la grandària comença amb la determinació de les distintes zones del text. Açò no es fa globalment sinó que es divideix la imatge en segments, i per a cada

segment es determinen les tres zones. La segmentació es basa en la idea de que caràcters que s'hagen escrit molt junts tindran el mateix estil d'escriptura, per aquest motiu la imatge es segmenta pels espais en blanc grans. Açò es fa calculant la projecció vertical i obtenint la grandària mitja d'espai. Una vegada segmentat, cada segment es binaritza i es suavitza amb l'algorisme RLSA [26], primer s'aplica horitzontalment i després verticalment. Per a cada columna del segment ens quedem amb el primer i últim píxels del tram de píxels negres continu més llarg. Aquestos píxels determinaran el contorn superior i inferior del segment. Basant-se amb la mitja i la desviació típica s'eliminen els punts del contorn que siguen anòmals. Per al contorn superior (inferior) es calcula un histograma horitzontal, la fila amb més valor es considera el límit superior (inferior).

Una vegada determinades les tres zones cal escalar-les. Les zones ascendents i descendents s'escalaran a una altura proporcional a la de la zona central. Respecte a l'altura de la zona central, hi han diverses alternatives per a calcular-la:

Màxima: l'altura, de la zona central, màxima de tots els segments de la imatge.

Segon màxim: la segona altura més gran.

Mitja: l'altura mitja de tots els segments de la imatge.

Mitja ponderada: la mitja de les altures de la zona central de cada segment, ponderades per les longituds dels seus respectius segments. Siga $\bar{A}$ l'altura mitja, A_i l'altura de la zona central del segment i i l_i la seua longitud, aleshores:

$$\bar{A} = \sum_{i}^{N} \frac{l_i}{\sum_j^N l_j} A_i \tag{2.9}$$

Mitja ponderada+5: La mitja ponderada més 5.

Moda amb context: la moda de les altures, es a dir, l'altura, de zona central de segment, més freqüent. Per a evitar salts abruptes la moda es calcula sobre un histograma d'altures suavitzat que té en compte el context:

$$\bar{H(i)} = \sum_{j=i-K}^{i+K} H(j) \tag{2.10}$$

En la Figura 2.4 es pot veure un exemple de normalització de la grandària.

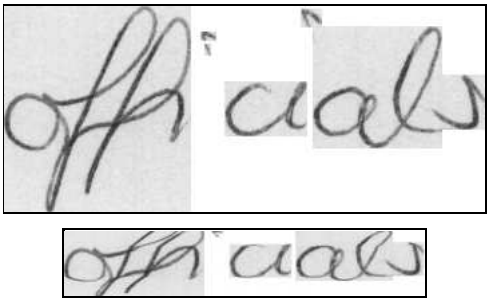


Figura 2.4: Dalt una imatge amb la paraula manuscrita *officials*, baix la mateixa imatge amb grandària normalitzada.

2.3 Reconeixement: extracció de característiques i classificació amb un HMM segmentat

Aquest mòdul es divideix en dos submòduls: extracció de característiques i classificador. Com ja s'ha comentat abans, l'extracció de característiques consisteix en recórrer la imatge d'esquerra a dreta extraient vectors de característiques, que composaran la seqüència d'observacions. Mentres que el mòdul de classificació consisteix en un classificador amb un HMM segmentat, on cada símbol es un caràcter del vocabulari. La classificació es realitza mitjançant *Viterbi*, tal i com s'ha vist en (1.13). El motiu de que no s'haja separat aquest mòdul en dos, es deu a l'alta interdependència entre els dos submòduls, ja que el tipus de HMM emprat vindrà determinat per la naturalesa i dimensió dels vectors de característiques. En la Figura 2.5 es mostra un esquema del mòdul.

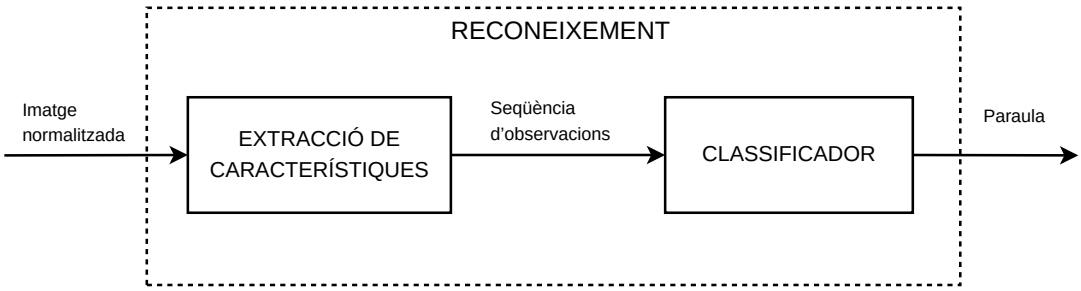


Figura 2.5: Esquema del mòdul de reconeixement.

La topologia d'un HMM fa referència al conjunt de possibles transicions entre estats permeses. Donat que l'escriptura s'escriu d'esquerra a dreta, és habitual en tasques de reconeixement d'escriptura, i així ho fem nosaltres, treballar amb HMMs amb topologies

lineals. En una topologia lineal s'estableix un ordre en els estats $\mathcal{Q}$, de manera que cada estat sols pot transitar a ell mateix o al següent estat. L'estat inicial sols pot transitar al primer estat, i l'estat final es considera l'estat que segueix a l'últim estat. En la Figura 2.6 es pot veure un exemple de topologia lineal.

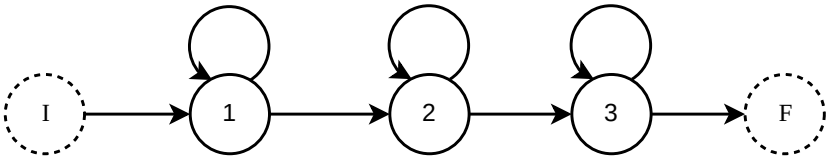


Figura 2.6: Exemple de topologia lineal en un HMM.

Considerarem dos alternatives per aquest mòdul. Per una banda el que anomenarem com mòdul PRHLT, que és el mòdul que ha vingut utilitzant en els últims temps el grup PRHLT (`http://prhlt.iti.es`). Aquest mòdul treballa amb vectors reals i mixtures de gaussianes als estats. És una proposta original del Dr. Alejandro Toselli [21]. Per una altra banda, una proposta original on es treballa amb vectors binaris i mixtures de Bernoulli als estats, a la que ens referim com mòdul Bernoulli HMM.

2.3.1 PRHLT

En aquesta aproximació els vectors de característiques es defineixen en el domini $\mathbb{R}^{60}$. On les 20 primeres característiques són valors de gris i les altres 40 són derivades horitzontals i verticals.

El procés d'extracció comença amb la divisió de la imatge original en $20 \times N$ cel·les de mostreig, respectant les proporcions originals. Per a cada columna de cel·les s'extrauen 20 valors de griso, 20 derivades horitzontals i 20 derivades verticals de valors de gris. Sobre cada cel·la de la columna es fixa un finestra de mostreig de 5 cel·les de la que s'extrauran 1 valor de gris, 1 derivada horitzontal i 1 vertical. El valor de gris s'obté aplicant un filtre gaussià sobre la finestra i normalitzant el valor resultant en el rang $[0, 1]$. Per a la derivada horitzontal es calcula primer el promig de gris per a cada columna de píxels de la finestra de mostreig. Després per regressió lineal es determina la recta $ax + b$, amb la distància a cadascun dels seus punts, que minimitza la funció objectiu J:

$$J = \sum_{i=1}^{n} w_i (g_i - (ax_i + b))^2, \tag{2.11}$$

on n és el nombre de columnes de píxels, x_i és la coordenada de la columna, g_i és el valor promig de gris en la columna i i w_i és el valor de la funció filtre gaussià en la posició x_i:

$$w_i = exp\left(-\frac{1}{2}\frac{(x_i - n/2)^2}{(n/4)^2}\right). \tag{2.12}$$

El coeficient a de la recta calculada és el valor de la derivada. La derivada vertical es calcula de forma anàloga. En la Figura 2.7 es pot veure un exemple de l'extracció de característiques. Com visualment no es distingeixen be les derivades hem afegit un exemple amb vectors de 180 dimensions.

Figura 2.7: Exemple d'extracció de característiques del mòdul PRHLT. Dalt la imatge original a escala $\times 0.5$. En el mig el vector d'observacions de dimensió 180 a escala $\times 0.5$. Baix el vector d'observacions de dimensió 60 a escala $\times 1$.

En els HMMs s'empren mixtures de gaussianes de dimensió 60, les quals són funcions de densitat de probabilitat. Degut al càlcul computacional i a que no s'ha demostrat una millora significativa, s'utilitzen gaussianes amb matrius de covariança diagonals. El nombre de components per a cada mixtura és objecte d'estudi en el capítol 4.

2.3.2 Bernoulli HMM

En aquesta proposta el vector d'observacions resultant és la imatge original escalada i binaritzada. Precisament un dels interessos principals d'aquesta proposta és treballar directament sobre la imatge binaritzada.

La imatge s'escala a una altura de 30 píxels, mantenint les proporcions originals. El motiu d'una altura de 30 píxels és purament empíric, basat en resultats de treballs anteriors [4]. La posterior binarització es realitza amb el mètode d'*Otsu*, ja explicat en la secció 2.2.3. La imatge binària resultant s'interpreta com una seqüència de vectors d'observacions definits en el domini $\{0,1\}^{30}$. En la Figura 2.8 es pot veure un exemple gràfic.

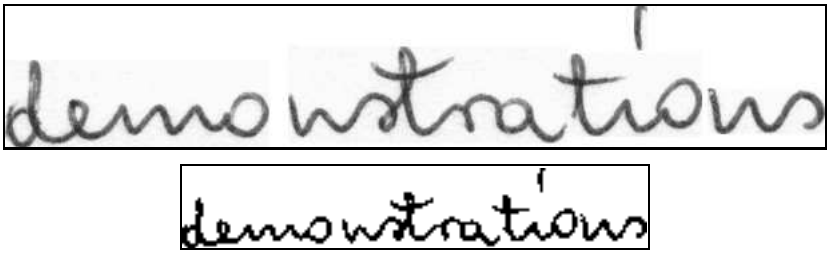


Figura 2.8: Exemple d'extracció de característiques del mòdul Bernoulli HMM. Dalt la imatge original a escala $\times 0.5$. Baix el vector d'observacions de dimensió 30 a escala $\times 1$.

En els HMMs s'empren mixtures de Bernoulli de dimensió 30, les quals són funcions de probabilitat. El nombre de components per estat s'estudia en el capítol 4.

Capítol 3

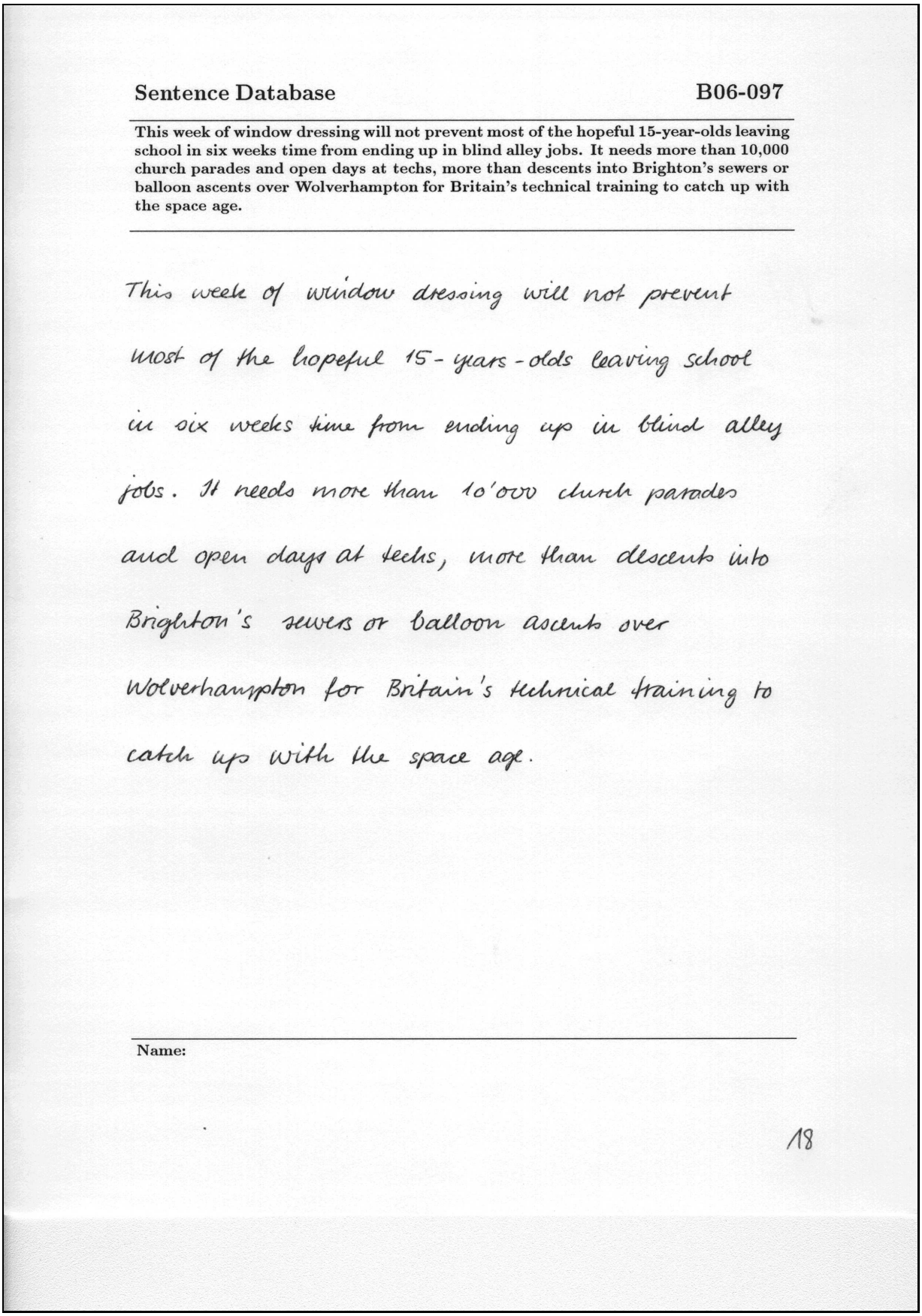
IAM database

3.1 Introducció

IAM database és un corpus de text anglès manuscrit no restringit [22, 23, 12, 30]. Aquest corpus ha sigut desenvolupat pel grup FKI del *Institut für Informatik und angewandte Mathematik* (IAM). La versió del corpus que hem utilitzat és la 3.0 que es pot baixar en (`http://www.iam.unibe.ch/fki/databases/iam-handwriting-database`).

El corpus està composat per un total de 1539 formularis amb text manuscrit. Cada formulari té una capçalera amb un identificador i el text imprès a escriure, i després un gran espai en blanc entre dos línies on es troba el text manuscrit. Un total de 657 de escriptors van participar en l'adquisició del corpus. Les dos úniques restriccions que van tindre van ser: que empraren plantilles per a escriure les línies rectes, amb una separació aproximada entre línies de $1.5cm$, i que si una paraula no es cabia en l'actual línia que canviaren de línia, abans que tindre que escriure-la atapida contra el marge. Per el demés es va donar total llibertat a l'hora d'escriure tant d'estil com d'instrument emprat, de fet, se'ls va demanar que escrigueren amb el seu estil d'escriptura habitual. Els formularis van ser escanejats amb una resolució de $300dpi$ en imatges en escala de grisos. Respecte als textos que apareixen en els formularis, tots tenen al voltant de 6 frases i almenys 50 paraules. Aquests fragments de text van ser extrets del corpus *Lancaster - Oslo/Bergen* (LOB) [8]. El corpus LOB és una recopilació de textos reals del 1961, escrits en anglès britànic. LOB té al voltant del milió de paraules, i es poden trobar fins a 15 temàtiques diferents: reportatge, religió, humor, etc. Per a cada formulari el corpus IAM proporciona un identificador d'escriptor i una referència al text original en el corpus LOB. En les Figures 3.1 i 3.2 es poden veure alguns exemples de formularis.

A més dels formularis, el corpus IAM conté tres subcorpus obtinguts a partir dels formularis mitjançant tècniques de segmentació automàtiques. Aquestos subcorpus són: un subcorpus de línies, un de frases i un altre de paraules. En aquest treball hem utilitzat el subcorpus de paraules, que s'explica en més detall en la secció 3.2. En l'última secció 3.3



Sentence Database **B06-097**

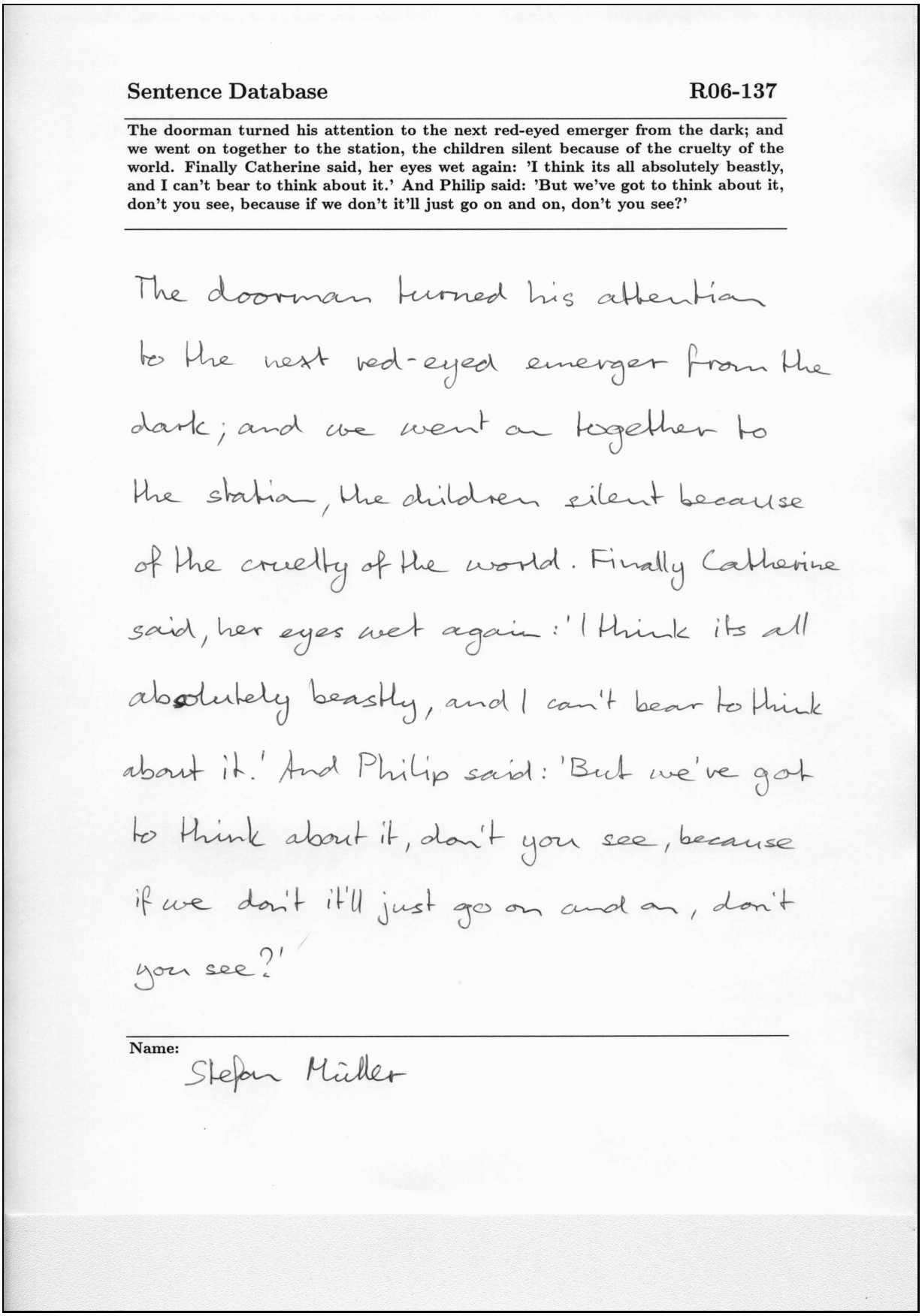
This week of window dressing will not prevent most of the hopeful 15-year-olds leaving school in six weeks time from ending up in blind alley jobs. It needs more than 10,000 church parades and open days at techs, more than descents into Brighton's sewers or balloon ascents over Wolverhampton for Britain's technical training to catch up with the space age.

This week of window dressing will not prevent most of the hopeful 15-years-olds leaving school in six weeks time from ending up in blind alley jobs. It needs more than 10'000 church parades and open days at techs, more than descents into Brighton's sewers or balloon ascents over Wolverhampton for Britain's technical training to catch up with the space age.

Name:

18

Figura 3.1: Exemple de formulari del corpus IAM. Formulari *B06-097*.



Sentence Database **R06-137**

The doorman turned his attention to the next red-eyed emerger from the dark; and we went on together to the station, the children silent because of the cruelty of the world. Finally Catherine said, her eyes wet again: 'I think its all absolutely beastly, and I can't bear to think about it.' And Philip said: 'But we've got to think about it, don't you see, because if we don't it'll just go on and on, don't you see?'

The doorman turned his attention to the next red-eyed emerger from the dark; and we went on together to the station, the children silent because of the cruelty of the world. Finally Catherine said, her eyes wet again: 'I think its all absolutely beastly, and I can't bear to think about it.' And Philip said: 'But we've got to think about it, don't you see, because if we don't it'll just go on and on, don't you see?'

Name: Stefan Müller

Figura 3.2: Exemple de formulari del corpus IAM. Formulari *R06-137*.

s'explica en detall com s'ha extret la partició, del subcorpus de paraules, amb la que s'han obtingut els resultats del capítol 4.

3.2 *Handwritten word dataset*

El subcorpus de paraules és fruit de segmentar el subcorpus de línies, que a la vegada és resultat de segmentar els formularis. Per a extraure les línies van corregir el *skew* dels formularis i van aïllar la secció de text manuscrit. Després per a segmentar en línies es van basar en l'histograma de transicions blanc-negre horitzontal, tallant pels mínims. El problema de decidir a quina línia pertanyen els traços que intersequen amb una línia de tall, el van resoldre emprant el centre de gravetat de les components connexes implicades. La segmentació en paraules es va dur a terme amb tècniques semblants a les citades en [17, 10]. El problema de les paraules segmentades ho resolen creant un graf de components connexes. Cada component està connectada amb la següent pels seus respectius centres de gravetats. A cada aresta s'assigna la distància entre les dos components. Finalment creen dos classes: distàncies entre paraules i distàncies dins de paraules. I amb l'algorisme d'Otsu calculen la distància llindar. Una explicació més detallada de la segmentació en paraules es pot trobar en [12].

El subcorpus de paraules resultant està compost d'aproximadament 115000 mostres, i un vocabulari prop de les 13500 paraules. A més de la transcripció, per a cada mostra es proporciona una etiqueta gramatical i un indicador de si la paraula està ben segmentada o no, encara que hem pogut comprovar que açò no és sempre cert. En la Taula 3.1 es mostren algunes dades del subcorpus, i en la Figura 3.3 es pot veure una mostra del subcorpus.

No existeixen molts resultats publicats amb aquesta tasca. Concretament sols tenim constància dels treballs del Dr. Simon Günter [6, 5]. En aquestos treballs aplica també HMMs amb gaussianes, però emprant vectors de característiques molt diferents dels emprats pel mòdul PRHLT 2.3.1. Aconsegueix reportar errors al voltant del 20%.

Taula 3.1: Algunes dades sobre el subcorpus de paraules de IAM3.0. De dalt a baix, d'esquerra a dreta: nombre d'escriptors, de mostres, de mostres ben segmentades, de paraules distintes, de caràcters distints, amplària mitja, altura mitja, *aspect ratio* mig, nombre de paraules amb almenys 10 mostres i amb almenys 5 mostres.

Nº Escriptors	657	**Amplària m.**	156 ± 116
Nº Mostres	115320	**Altura m.**	70 ± 33
Nº Mostres *ok*	96456	***Aspect Ratio m.***	0.72 ± 0.71
Nº Paraules	13542	**Nº Paraules (≥ 10)**	1289
Nº Caràcters	78	**Nº Paraules (≥ 5)**	2573

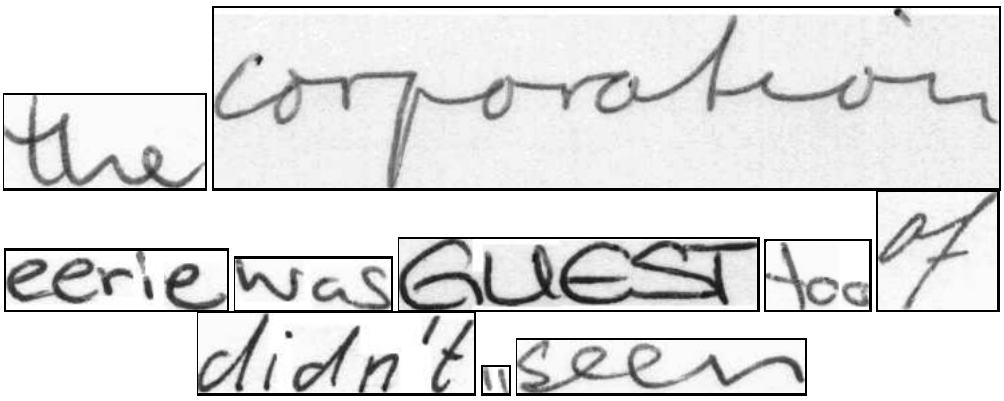


Figura 3.3: Mostres del subcorpus de paraules IAM.

3.3 Partició per a l'experimentació

Per a l'experimentació realitzada en el capítol 4, hem utilitzat una partició d'un subconjunt del subcorpus de paraules. No ha sigut possible reproduir la partició utilitzada en [6, 5], per aqueix motiu hem tingut que crear una partició pròpia. Per a facilitar una possible reproducció dels experiments detallem a continuació com s'ha creat la partició. Primer es seleccionaren totes les mostres etiquetades com a ben segmentades. Després s'eliminaren totes les paraules amb menys de deu mostres. Les mostres *r06-022-03-05* i *a01-117-05-02* també foren eliminades per estar els fitxers d'imatge incorrectes. Aquest subconjunt es dividí en un conjunt d'entrenament i de test. La divisió es va fer per escriptor, de manera que totes les mostres d'un escriptor aparegueren, o bé en entrenament, o bé en test, pero no en els dos conjunts. La divisió va ser aleatòria intentant que el conjunt d'entrenament continguera el 80% de les mostres. Els identificadors dels escriptors de test són: *000*, *003*, *006*, *019*, *028*, *040*, *047*, *048*, *060*, *069*, *071*, *080*, *081*, *087*, *088*, *092*, *096*, *098*, *109*, *112*, *113*, *126*, *129*, *141*, *147*, *155*, *156*, *157*, *158*, *163*, *166*, *168*, *172*, *182*, *183*, *185*, *186*, *191*, *193*, *202*, *207*, *214*, *224*, *230*, *233*, *238*, *241*, *243*, *245*, *249*, *250*, *251*, *252*, *254*, *256*, *257*, *265*, *266*, *267*, *268*, *273*, *279*, *297*, *302*, *308*, *319*, *320*, *334*, *342*, *363*, *370*, *385*, *391*, *400*, *411*, *413*, *427*, *428*, *431*, *440*, *453*, *463*, *469*, *473*, *485*, *487*, *488*, *492*, *497*, *510*, *514*, *519*, *524*, *526*, *531*, *535*, *544*, *553*, *554*, *558*, *559*, *563*, *564*, *568*, *570*, *571*, *574*, *575*, *576*, *580*, *588*, *590*, *593*, *603*, *604*, *607*, *617*, *618*, *619*, *622*, *633*, *636*, *650*, *664*.

En la Taula 3.2 es poden veure algunes dades de la partició.

Taula 3.2: Algunes dades sobre la partició per a l'experimentació. De dalt a baix, d'esquerra a dreta: nombre d'escriptors, d'escriptors de *train*, de mostres, de mostres de *train*, de paraules distintes, de caràcters distints, amplària mitja, altura mitja i *aspect ratio* mig.

Nº Escriptors	657	**Nº Caràcters**	71
Nº Escriptors *train*	533	**Amplària m.**	123 ± 87
Nº Mostres	73837	**Altura m.**	65 ± 31
Nº Mostres *train*	59157	***Aspect Ratio m.***	0.81 ± 0.76
Nº Paraules	1117		

Capítol 4

Experimentació

4.1 Introducció

L'experimentació que presentem en aquest capítol l'hem realitzada amb tres objectius en ment. Primer de tot obtenir resultats amb la tasca de paraules amb els dos models presentats en la secció 2.3. Segon, fer una exploració de paràmetres, mètodes de preprocés i estratègies d'entrenament, que poden donar-mos una idea del comportament per a altres tasques, sobretot les relacionades amb el corpus IAM. I finalment, fer una comparació dels dos models presentats en 2.3.

L'experimentació s'ha realitzat sobre la partició, del subcorpus de paraules, presentada en la secció 3.3. Amb HMM lineals pot ocórrer, tant en entrenament com en test, que ens trobem amb seqüències d'observacions que són massa curtes per a ser generades per un HMM. Aquest problema l'hem solucionat modificant l'extracció de característiques per a què s'exigisca una llargària mínima de seqüència. En les mostres d'entrenament, la llargària mínima per a cada seqüència és el nombre d'estats del model virtual associat a la paraula de la mostra. Per a les mostres de test, la llargària mínima és el nombre d'estats del model virtual (associat a una paraula del vocabulari) amb menys estats.

Respecte al *software* emprat, per al preprocés, i per a l'extracció de característiques del mòdul PRHLT (2.3.1), s'han emprat ferramentes internes del grup PRHLT. Per als HMM amb mixtures de gaussianes als estats hem emprat el *software* HTK [28]. I finalment per als HMMs amb mixtures de Bernoulli hem utilitzat un *software* propi.

Hem organitzat l'experimentació de la següent manera. Primer, sobre un sistema base sense preprocés, hem estudiat l'estratègia d'inicialització 4.2. Una vegada fixada l'estratègia d'inicialització, hem fet una exploració de possibles configuracions del mòdul de preprocés 4.3. Seguidament, sobre el millor sistema, hem estudiat el nombre d'estats i components per estat dels HMMs 4.4. Finalment el *Grammar Scale Factor* i un estudi del nombre d'iteracions ha sigut realitzat en 4.5 i 4.6. Tots aquests estudis s'han fet de forma paral·lela per als dos mòduls de reconeixement presentats.

4.2 Inicialització

El sistema base amb el que comencem no té preprocés, i tots els HMMs tenen 8 estats i una única component per estat. Estudiem dos alternatives per a la inicialització.

La primera alternativa consisteix en inicialitzar tots els estats amb els mateixos paràmetres. Estos paràmetres, mitges i matrius de covariança en gaussianes o probabilitats d'emissió en bernoullis, s'han calculat prèviament a partir de totes les observacions de la partició d'entrenament, per màxima versemblança. Les probabilitats de transició s'inicialitzen a 0.4 les de transitar al següent estat, i a 0.6 les de transitar al mateix estat.

La segona alternativa consisteix en utilitzar el model entrenat (4 iteracions) que hem inicialitzat com abans, i realitzar segmentació forçada sobre les mostres d'entrenament. Com per a cada mostra d'entrenament es coneix el model virtual que l'accepta, encara que el model no siga massa bo es poden obtenir segmentacions prou bones. Una vegada tenim les mostres segmentades en caràcters, cada HMM s'inicialitza primer uniformement i després s'entrena a la *Viterbi*. Cada HMM s'ha entrenat fins convergència o un màxim de 20 iteracions. Quan l'increment relatiu de la logversemblança no supera el 0.0001 es considera que ha convergit.

En els dos casos l'entrenament ha consistit en 4 iteracions de *Baum-Welch*. En la Taula 4.1 es poden veure els resultats per a les dos inicialitzacions. S'observa que la segona alternativa obté millors resultats però no hi ha una gran diferència.

Taula 4.1: Error (en percentatge) de classificació, sense preprocés, 8 estats i 1 component per estat amb les dos alternatives d'inicialització proposades. Tots els estats amb els mateixos paràmetres (A1). Segmentació de les mostres i reinicialització a la *Viterbi* (A2).

	PRHLT	**Bern. HMM**
A1	81.92	75.99
A2	$\mathbf{81.91}$	$\mathbf{73.56}$

4.3 Preprocés

En la secció 2.2 proposàvem un preprocés que es dividia en tres submòduls, i per a cada submòduls es plantejaven diverses alternatives. En aquesta secció estudiem les diverses alternatives incloent l'absència de tots o part dels submòduls. Idealment es deurien de provar totes les possibles combinacions, però com el nombre d'experiments seria massa gran, hem relaxat l'experimentació estudiant cada submòdul per separat. Primer hem realitzat un estudi de la correcció del *slant*, després del submòdul de normalització de l'escala de grisos i finalment del de normalització de la grandària.

4.3.1 Correcció del *slant*

Partint del sistema base de 4.2, i utilitzant la inicialització per segmentació i reinicialització a la *Viterbi*, hem provat a afegir el mòdul de correcció del *slant* amb els tres mètodes vists en 2.2.2: IDIAP, longitud contorn i desviació típica. En la Taula 4.2 es mostren els resultats per als tres mètodes i sense mòdul. S'aprecia que la correcció del *slant* té un impacte significatiu en l'error, especialment amb les bernoullis. Entre els tres mètodes de correcció la diferència és discreta però pareix funcionar millor la desviació típica.

Taula 4.2: Error (en percentatge) de classificació, per als millors sistemes de la Taula 4.1 amb quatre possibles mètodes de correcció del *slant*: sense correcció, mètode IDIAP, per longitud del contorn i màxima desviació típica.

	PRHLT	**Bern. HMM**
-	81.91	73.56
IDIAP	79.87	62.73
L. Contorn	79.58	62.30
D. Típica	**79.04**	**61.78**

4.3.2 Normalització de l'escala de grisos

En 2.2.1 proposàvem dos alternatives per al submòdul de normalització de l'escala de grisos: normalització del contrast i binarització *Otsu*. En aquest apartat estudiem les dos alternatives, contrastant-les amb la no normalització. Els experiments s'han realitzat sobre el sistema de l'apartat anterior que millors resultats ha obtingut, açò és, el submòdul de correcció del *slant* mitjançant desviació típica. La normalització del contrast ha consistit realment en dos normalitzacions concatenades. En la primera el 5% dels píxels més obscurs s'han fet negres, i el 70% dels píxels més clars s'han fet blancs. En la segona s'han establert llindars inferior i superior, concretament 150 per a negre i 200 per a blanc. Tots aquestos valors s'han extret de forma empírica, i es solen emprar normalment sobre tasques relacionades amb el corpus IAM. En la taula 4.3 es poden veure els resultats. En el cas del mòdul PRHLT hi ha una millora de casi 3 punts, i el millor resultat s'obté amb normalització del contrast. En el cas del mòdul Bernoulli HMM la millora no aplega al punt, i s'obté amb binarització, de fet amb normalització de contrast el resultat empitjora.

Taula 4.3: Error (en percentatge) de classificació, per als millors sistemes de la Taula 4.2 amb tres possibles mètodes de normalització de l'escala de grisos: sense normalització, binarització *Otsu* i normalització del contrast.

	PRHLT	**Bern. HMM**
-	79.04	61.78
Bi. Otsu	76.38	**60.90**
N. Contrast	**76.34**	62.43

4.3.3 Normalització de la grandària

Sobre els millors sistemes de l'apartat anterior, estudiem el mòdul de normalització de la grandària. L'estudi s'ha dividit en dos parts, primer s'han provat els diferents mètodes per al càlcul de la grandària de la zona central, citats en 2.2.3. Les grandàries per a la zona d'ascendents i descendents ha sigut la que tenia la ferramenta per defecte, que és 30% de la zona central per a ascendents i 15% per a descendents. En la Taula 4.4 es poden veure els resultats.

Taula 4.4: Error (en percentatge) de classificació, per als millors sistemes de la Taula 4.3 amb 6 possibles mètodes per al càlcul de la grandària de la zona central: altura màxima, segon màxim, mitja, mitja ponderada, mitja ponderada més cinc i moda amb context. Amb 30% grandària ascendents i 15% descendents. - és sense normalitzar grandària.

	PRHLT	**Bern. HMM**
-	76.34	60.90
Max.	66.17	47.64
2 Max.	66.16	47.70
Mitja	66.14	47.55
Mit. Ponderada	**62.96**	**46.64**
Mit. Pon. + 5	66.12	47.68
Moda Context	**62.96**	47.10

Els resultats mostren una gran millora al normalitzar la grandària. Els dos mètodes que millor han resultat són la mitja ponderada i la moda amb context, un poc millor la mitja. Després s'ha fet un estudi sobre la grandària dels ascendents i descendents, provant per a diversos percentatges i fixant l'altura igual per a descendents i ascendents. El mètode elegit per al càlcul de l'altura central ha sigut la mitja ponderada. El resultats estan recollits en la Taula 4.5.

Taula 4.5: Error (en percentatge) de classificació, per als millors sistemes de la Taula 4.4 i diverses grandàries (en percentatge de la grandària de la zona central) per a ascendents i descendents. '-' és 30% grandària ascendents i 15% descendents.

	PRHLT	**Bern. HMM**
-	$\mathbf{62.96}$	$\mathbf{46.64}$
5%	67.72	52.55
10%	65.15	48.79
20%	63.78	46.86
30%	65.35	48.00
40%	67.72	49.54

Com era d'esperar els valors per defecte de la ferramenta obtenen els millors resultats, s'entén que hi ha hagut un estudi previ per a seleccionar-los. De totes maneres podem observar en els altres resultats, que al voltant del 20% és com millor va, i que si augmenta o disminueix la grandària empitjoren poc a poc.

4.4 Nombre d'estats i nombre de components

Partint dels sistemes amb el millor preprocés i inicialització, estudiem el nombre d'estats i de components per estat. Per al nombre de components començarem amb models d'una component per estat, i en cada experiment doblarem el nombre de components fins aplegar a 64 components per estat. La inicialització dels models d'una component es realitza com s'ha vist abans. Mentres que, el de $2x$ components s'inicialitza a partir del model entrenat de x components de la següent manera: per a cada estat del model es selecciona la component amb major probabilitat. La component seleccionada s'elimina i s'afegeixen dos noves components que són el resultat de distorsionar la component eliminada. La probabilitat d'aquestes noves components és la mitat del de la component eliminada. En el cas de gaussianes la distorsió és un ± 0.2 de la desviació típica a les mitges, en el cas de les bernoullis un ± 0.05 a les probabilitats d'emissió. Aquest procés es repeteix fins obtenir el nombre de components desitjat.

Per al nombre d'estats estudiarem dos estratègies, que s'expliquen en les següents seccions: nombre d'estats fixe i variable. En els dos casos s'han fet conjuntament els dos estudis, enfrontant el nombre d'estats contra el nombre de components per estat.

4.4.1 Nombre d'estats fixe

Aquesta estratègia consisteix en què tots els HMMs tenen el mateix nombre d'estats. Hem provat per a $\{4, 6, 8, 10, 12\}$ estats. Els resultats per al mòdul del PRHLT i per al

Bernoulli HMM, es poden veure respectivament en les Figures 4.1 i 4.2.

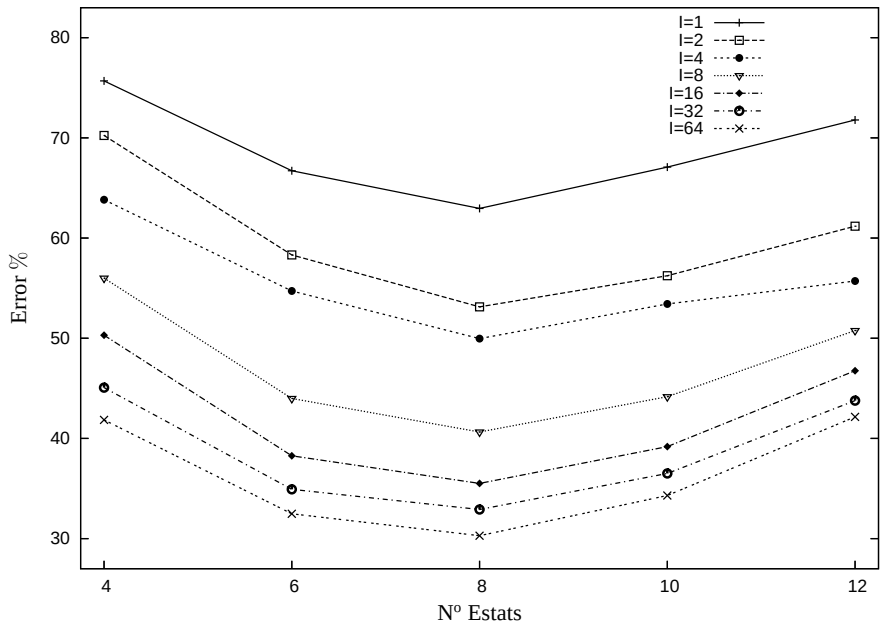


Figura 4.1: Error (en percentatge) de classificació front a nombre d'estats per HMM, per a diferent nombre de components per estat I. Resultats amb el millor sistema amb el mòdul PRHLT de la Taula 4.4.

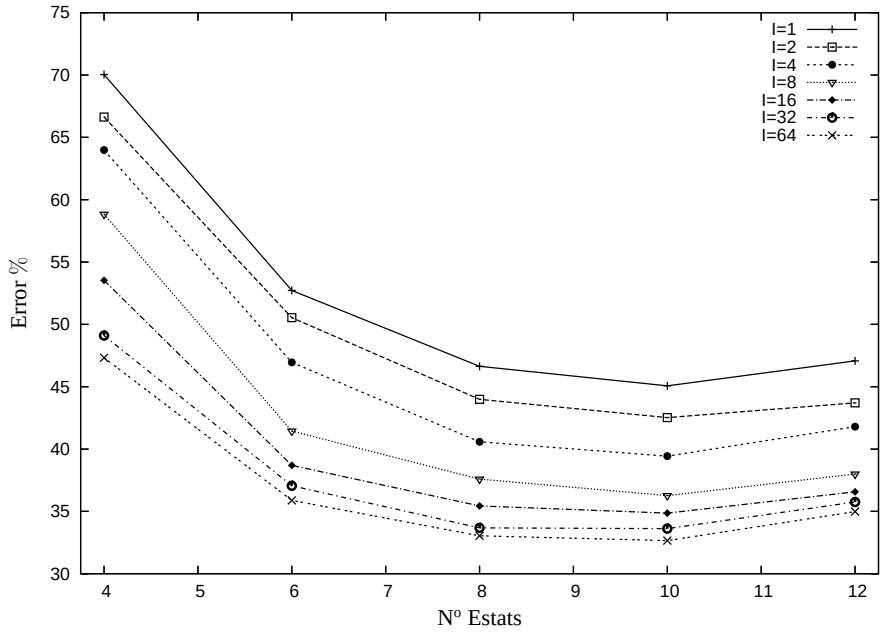


Figura 4.2: Error (en percentatge) de classificació front a nombre d'estats per HMM, per a diferent nombre de components per estat I. Resultats amb el millor sistema amb el mòdul Bernoulli HMM de la Taula 4.4.

El millor resultat per al mòdul PRHLT és un 30.3% amb 8 estats i 64 components. Per al mòdul Bernoulli HMM el millor resultat és un 32.7% amb 8 estats i 64 components. Queda patent que l'error disminueix directament proporcional amb el nombre de components. Si bé és cert que amb el PRHLT hi ha un marge de millora que el Bernoulli HMM pareix no tindre ja. El nombre d'estats que millor funciona està al voltant de 8-10. El millor funcionament de Bernoulli HMM amb nombre d'estats més elevats tal volta siga degut a seqüències més llargues. Resulta curiós com amb una component Bernoulli HMM obté errors per davall del 50% i PRHLT no aplega al 65%, i que al final siga aquest el que millor resultat obté.

4.4.2 Nombre d'estats variable

En aquesta estratègia no tots els HMMs tenen el mateix nombre d'estats. El nombre d'estats dels HMMs es calculen a partir de la longitud de seqüència d'observació mitja associada a cada HMM, i d'un factor de càrrega f que fixem per a tot el model. El factor de càrrega indica cada estat quants vectors emet en terme mig, per exemple, un factor de càrrega de $f = 0.5$, vol dir que en terme mig cada estat emet dos vectors. Per tant, per a cada HMM el nombre d'estats es calcula com el factor de càrrega per la longitud mitja de les observacions associades a aqueix HMM.

Amb un HMM segmentat apareix un problema, i és que els HMMs emeten símbols (caràcters), mentres que les mostres són seqüències de símbols (paraules). Resulta doncs impossible calcular la llargària de seqüència mitja d'un determinat caràcter. Per resoldre aquest problema el que fem és crear un model inicial on tots els HMMs tenen 8 estats, menys els d'alguns signes de puntuació que hem fixat a 4 estats. Amb aquest model entrenat realitzem una segmentació forçada sobre el conjunt d'entrenament, segmentant-lo en caràcters. Amb els segments calculem la llargària de seqüència mitja per a cada HMM.

Hem provat diferents factors de càrrega. Per al mòdul PRHLT els resultats es poden veure en la Figura 4.3, per al Bernoulli HMM es poden veure en la Figura 4.4.

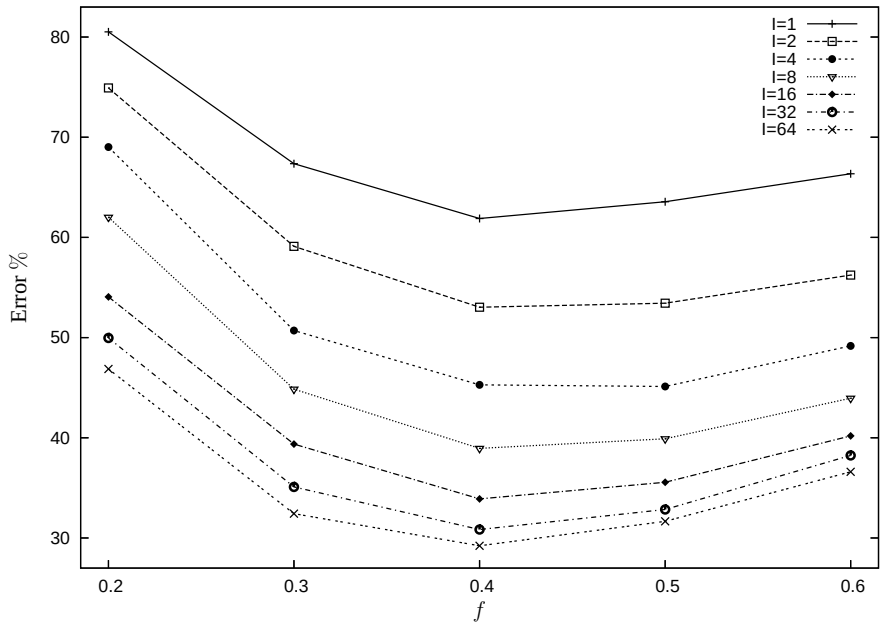


Figura 4.3: Error (en percentatge) de classificació front a factor de càrrega f, per a diferent nombre de components per estat I. Resultats amb el millor sistema amb el mòdul PRHLT de la Taula 4.4.

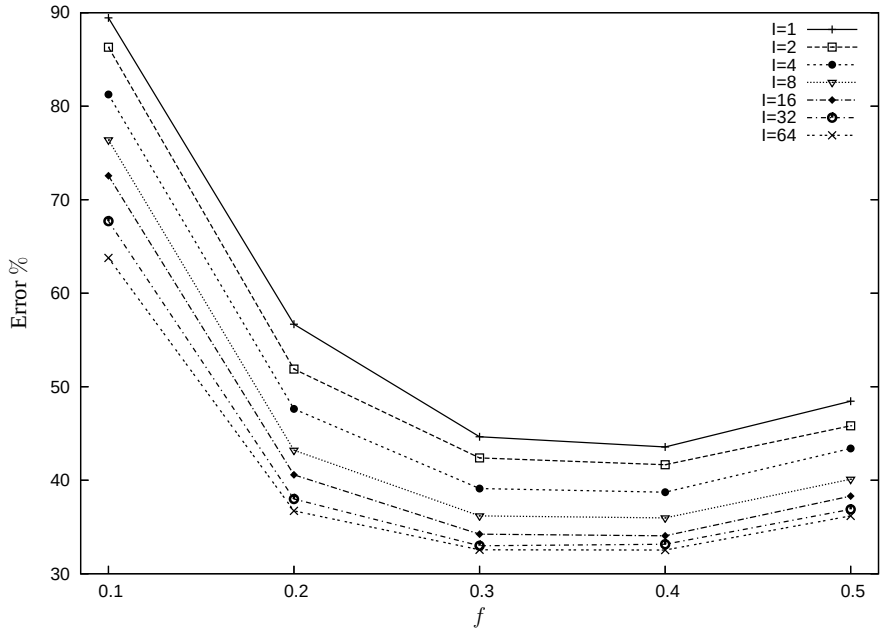


Figura 4.4: Error (en percentatge) de classificació front a factor de càrrega f, per a diferent nombre de components per estat I. Resultats amb el millor sistema amb el mòdul Bernoulli HMM de la Taula 4.4.

El millor resultat per al mòdul PRHLT és un 29.2% amb factor de càrrega 0.4 i 64 components. Amb el mòdul Bernoulli HMM el millor resultat ha sigut un 32.5% amb factor de càrrega 0.4 i 64 components. Amb Bernoulli HMM el factor 0.3 és comporta pràcticament igual al 0.4. El comportament és igual al vist amb nombre d'estats fixes, per tant les conclusions són les mateixes. Es pot veure una lleugera millora respecte a nombre d'estats fixe, en el cas de les bernoullis aquesta millora és pràcticament insignificant.

4.5 *Grammar Scale Factor*

El *Grammar Scale Factor* és un paràmetre α que modifica l'expressió de reconeixement (1.13) de la següent manera:

$$y^* \approx \arg\max_y \; p(y)^\alpha \max_{S_0,s_1,\ldots,s_L,S_T} \prod_t P(x_{i(t)},\ldots,x_{j(t)},s_{i(t)},\ldots,s_{j(t)},S_t \mid \mathcal{H}_{y_t},S_{t-1}). \tag{4.1}$$

Aquest factor s'utilitzava ja amb èxit en el reconeixement automàtic de la parla. No coneixem una explicació formal de les bondats del *Grammar Scale Factor*. Però intuïtivament, degut a la llargària de les seqüències d'observacions, i a què les observacions són vectors

multidimensionals, les probabilitats solen ser molt baixes en comparació amb les probabilitats a priori de cada paraula. Açò provoca que durant el reconeixement les probabilitats a priori siguen ignorades. El que fa el *Grammar Scale Factor* és donar importància a aquestes probabilitats.

En la Figura 4.5 podem observar resultats per als millors sistemes de la secció anterior. Tant en el mòdul PRHLT com el Bernoulli HMM s'aprecia una millora, encara que amb PRHLT un poc més pronunciada. Amb PRHLT s'aplega a un 25.4% i amb Bernoulli HMM a un 29.9%. No obstant s'aprecia com els valors del *Grammar Scale Factor* actuen de manera molt diferent en cada sistema.

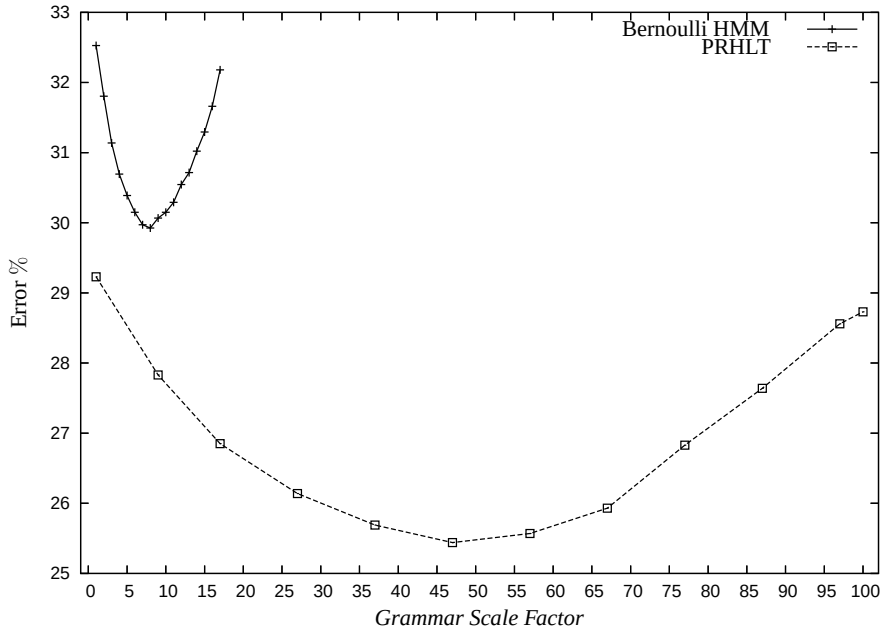


Figura 4.5: Error (en percentatge) de classificació front a *Grammar Scale Factor*. Resultats del mòdul PRHLT amb el millor sistema de la Figura 4.3. Resultats del mòdul Bernoulli HMM amb el millor sistema de la Figura 4.4.

4.6 Nombre d'iteracions

Sobres els millors sistemes de la secció anterior, hem realitzat un estudi de l'error de classificació en test front al nombre d'iteracions d'entrenament (en inicialització sempre 4 iteracions). En la Figura 4.6 es poden veure els resultats. Amb el mòdul Bernoulli HMM a partir de la iteració 5 comença a sobreentrenar-se. El millor resultat obtingut amb aquest mòdul continua sent un 29.9% amb 4 iteracions. En el cas del mòdul PRHLT l'error continua baixant fins la iteració 8 on s'estabilitza. Aquest mòdul aconsegueix baixar unes

dècimes l'error fins aplegar a un 25.0%. En qualsevol cas queda patent que fer més de 4 iteracions no compensa.

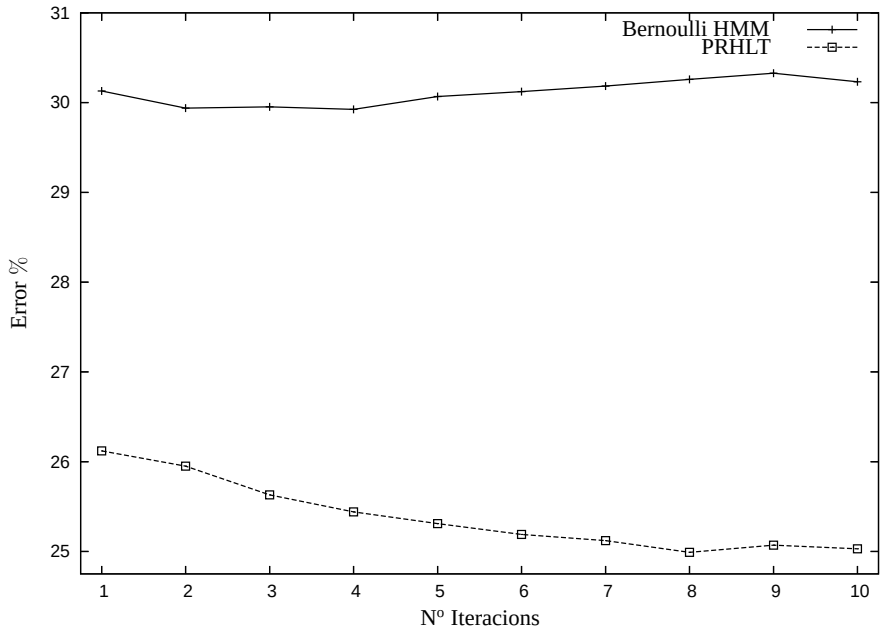


Figura 4.6: Error (en percentatge) de classificació front a nombre d'iteracions d'entrenament. Resultats amb els millors sistemes de la Figura 4.5

Tractant-se de l'últim estudi incloem matrius de confusió amb les deu parelles de paraules amb més errades. En la Taula 4.6 la matriu de confusió per al mòdul PRHLT, i en la Taula 4.7 la matriu de confusió per al mòdul Bernoulli HMM.

Podem observar com els dos sistemes tendeixen a fallar en les mateixes parelles de paraules, si bé és cert, en algunes parelles el mòdul PRHLT sol fallar més que el Bernoulli HMM, i a l'inrevés. Així, per exemple, en la parella [*a,"*] el mòdul PRHLT comet 40 errades i el Bernoulli HMM sols 9. De forma anàloga però a l'inrevés ocorre amb la parella [*to,he*], i més exagerat amb [*to,is*]. La parella amb la que més errades cometen els dos mòduls és [.,,], que per una altra banda és un error comprensible. Es pot observar com els signes de puntuació es solen confondre prou. En general les errades són prou raonables, són errades entre paraules semblats: [*I,!*], [*on,an*], [*is,in*], [*He,the*], etc. Trobem també algunes paralles que costa un poc d'entendre, però que almenys continuen tenint més o menys la mateixa amplària, com ho són: [*to,by*], [*to,is*], [*to,I*], i poques més. I com no, hi han unes poques parelles que sorprèn que es confonguen, a destacar [*a,"*] i [*I,,*]. Així i tot, les errades en la parella [*I,,*] són raonables si tenim en compte com es proporcionen les mostres. Donat que les paraules en el corpus estan aïllades en una caixa de mínima inclusió, tota informació de grandària respecte a altres paraules es perd. A més,

Taula 4.6: Matriu de confusió per al millor resultat del mòdul PRHLT, amb les deu parelles de paraules amb més errades. A l'esquerra paraules verdaderes, dalt paraules reconegudes.

	!	**"**	**’**	**,**	**-**	**.**	**He**	**I**	**Mr.**	**a**	**an**	**by**	**he**	**her**	**in**	**is**	**me**	**on**	**the**	**to**
!	18	1	2	0	0	0	0	4	0	2	0	0	0	0	0	0	0	0	0	0
"	0	214	10	15	0	15	0	2	0	4	0	0	0	0	0	4	0	0	1	1
’	3	7	4	25	1	8	0	0	0	0	0	0	0	0	0	0	0	0	0	0
,	**31**	7	24	676	0	**38**	0	2	0	1	0	0	0	0	0	1	0	0	0	0
-	0	0	0	0	32	**28**	0	1	0	0	0	0	0	0	0	0	0	0	0	0
.	0	10	15	**100**	4	693	0	2	0	1	0	0	0	0	0	0	0	0	0	0
He	0	0	0	0	0	0	20	0	0	0	0	0	7	0	0	0	0	0	**29**	3
I	**48**	0	0	26	0	1	0	26	0	0	0	0	0	0	0	0	0	0	0	0
Mr.	0	0	0	0	0	0	0	2	1	0	0	0	1	**43**	0	0	0	0	6	0
a	0	**40**	1	0	0	2	0	1	0	288	5	1	0	0	4	10	0	1	1	1
an	0	0	0	0	0	0	0	0	0	0	47	0	0	0	1	0	0	1	0	0
by	0	0	0	0	0	1	0	3	0	1	0	86	0	0	0	1	0	1	0	0
he	0	0	0	0	0	0	0	0	0	0	0	0	122	0	0	0	1	0	11	0
her	0	0	0	0	0	0	0	0	0	0	0	0	1	32	0	0	0	0	0	0
in	0	2	2	0	0	1	0	9	0	9	11	0	2	0	237	11	0	3	1	1
is	0	1	0	0	0	0	0	1	0	2	1	0	0	0	**30**	121	0	0	0	4
me	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	10	0	2	0
on	0	4	0	0	0	0	0	0	0	6	**29**	0	0	0	7	0	0	60	0	0
the	0	0	0	0	0	0	5	4	0	1	0	1	6	0	5	1	11	0	921	4
to	0	0	0	0	0	0	1	11	0	1	0	3	0	0	3	0	0	0	1	453

Taula 4.7: Matriu de confusió per al millor resultat del mòdul Bernoulli HMM, amb les deu parelles de paraules amb més errades. A l'esquerra paraules verdaderes, dalt paraules reconegudes.

	!	**"**	**’**	**,**	**-**	**.**	**He**	**I**	**Mr.**	**a**	**an**	**by**	**he**	**her**	**in**	**is**	**me**	**on**	**the**	**to**
!	13	0	1	8	0	0	0	4	0	0	0	0	0	0	0	0	0	0	0	1
"	0	190	11	**35**	1	6	0	10	0	5	0	0	2	0	1	2	0	0	0	1
’	1	8	6	**31**	0	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0
,	16	1	17	736	0	15	0	3	0	0	0	0	0	0	0	1	0	0	0	0
-	0	6	0	1	40	19	0	1	0	0	0	0	0	0	0	0	0	0	0	0
.	0	6	15	**92**	2	714	0	2	0	1	0	0	0	0	0	0	0	0	0	0
He	0	1	0	0	0	0	20	0	0	2	0	0	17	0	1	0	1	0	14	3
I	**24**	0	2	**57**	0	0	0	18	0	0	0	0	0	0	0	0	0	0	0	0
Mr.	0	2	0	0	0	0	2	3	0	0	0	3	0	22	2	2	0	0	0	0
a	0	9	0	2	0	0	0	21	0	322	1	1	0	0	3	5	0	1	2	0
an	0	0	0	0	0	0	0	1	0	0	46	1	0	0	1	0	0	3	0	0
by	0	0	0	0	0	1	0	0	0	5	0	78	0	1	2	0	0	1	0	1
he	0	0	0	0	0	0	1	4	0	0	0	1	121	0	0	0	0	0	4	0
her	0	0	0	0	0	0	0	0	0	0	0	0	2	32	0	1	0	0	1	1
in	0	5	3	1	0	1	0	22	0	6	2	0	1	0	230	5	0	3	5	1
is	0	0	8	0	0	0	0	11	0	2	0	2	1	0	6	104	0	0	0	10
me	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	9	0	1	0
on	0	0	0	0	0	0	0	2	0	11	**32**	0	1	0	0	0	0	60	0	0
the	0	0	0	0	0	0	6	1	0	7	0	5	**49**	0	6	2	**56**	0	795	13
to	0	1	1	0	0	0	0	18	0	4	1	**27**	0	0	4	**38**	0	0	1	382

en una paraula d'una única lletra és molt complicat determinar si aquesta té ascendents i/o descendents. Per tant, moltes vegades és impossible diferenciar una coma aïllada d'una *I*. De fet açò també explicaria l'elevada confusió entre l'apòstrof i la coma.

Capítol 5

Conclusions i treball futur

Hem presentat els HMMs segmentats amb mixtures de Bernoulli als estats, extensió dels HMM amb bernoullis als estats, que ja vam presentar en un treball anterior. Sobre el subcorpus de paraules de l'IAM hem realitzat una experimentació prou detallada amb els HMMs amb mixtures de Bernoulli, comparant-los sempre amb els HMMs amb mixtures de gaussianes amb les característiques que es venen emprant al grup PRHLT. Malgrat que actualment el corpus IAM és un corpus referència en RTM, no hi han molts resultats en la literatura amb el subcorpus de paraules. En aqueix aspecte, a més dels resultats de la nova proposta, hem presentat resultats detallats amb el sistema del grup. A més hem descrit el protocol d'experimentació emprat, per a què puga ser fàcilment reproduït. Açò pot ser molt útil per a provar nous models basats en HMM, ja que les conclusions són fàcilment exportables al reconeixement de frases, sense els inconvenients de la influència del model de llenguatge, i els costos excessius dels experiments amb frases.

El comportament de les bernoullis s'ha estudiat explorant gran quantitat de paràmetres i amb diferents configuracions del sistema. Hem partit d'un sistema senzill amb HMMs de 8 estats i una component sense ningun tipus de preprocés. Després hem anant millorant el sistema provant el comportament amb diferents mòduls de preprocés. I finalment hem realitzat estudis de nombre de components, nombre d'estats, *Grammar Scale Factor* i nombre d'iteracions. Al principi de l'experimentació els errors eren altíssims, al voltant del 80%, i al final hem estat treballant en errors al voltant del 30%. Un error al voltant del 30% és un error alt, que denota la complexitat de la tasca que hem tractat. La baixada de 50 punts s'ha degut principalment al preprocés i a l'augment del nombre de components. Aquest error és 10 punts superior al reportat en [6]. No obstant la comparació no és del tot justa, ja que tal i com s'explica en [5], el preprocés l'apliquen al corpus de línies i després segmenten les línies en paraules. El principal avantatge de fer-ho així resideix a l'hora de la normalització de la grandària. Per exemple, amb un signe de puntuació o una *a* aïllada, és molt més complex normalitzar la grandària quan sols es disposa de la caixa mínima d'inclusió del caràcter que quan el caràcter en qüestió es troba inserit dins d'una línia de text, on el context ens aporta informació sobre les posicions i grandàries relatives

entre caràcters. En qualsevol cas, un 20% és un error alt que torna a ficar en evidència la complexitat de la tasca.

Respecte a la comparativa de les bernoullis amb el sistema del grup PRHLT, amb el del grup s'ha aplegat a obtenir un error del 25.0%, i amb les bernoullis del 29.9%. Açò és una diferència de 5 punts, que tenint en compte l'elevat error no suposa una gran diferència. Destacar com a dada curiosa, que amb una component per estat amb bernoullis s'obtenien errors d'aproximadament 15 punts menys que amb gaussianes. La diferència de sols 5 punts i l'impacte del preprocés, convida a continuar en aquesta línia de treball, estenent les mixtures de Bernoulli a mixtures de Bernoulli invariants a transformacions.

Bibliografia

[1] Leonard E. Baum, Ted Petrie, George Soules, and Norman Weiss. A maximization technique occurring in the statistical analysis of probabilistic functions of markov chains. *The Annals of Mathematical Statistics*, 41(1):164–171, 1970.

[2] R.M. Bozinovic and S.N. Srihari. Off-line cursive script word recognition. 11(1):68–83, January 1989.

[3] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1):1–38, 1977.

[4] A. Giménez-Pastor and A. Juan-Císcar. Bernoulli hmms for off-line handwriting recognition. In *Proc. of the 8th Int. Workshop on Pattern Recognition in Information Systems (PRIS 2008)*, pages 86–91, Barcelona (Spain), June 2008.

[5] Simon Günter. *Multiple Classifier Systems in Offline Cursive Handwriting Recognition*. PhD thesis, Institut für Informatik und angewandte Mathematik, Universität Bern, Bern, Switzerland, Jan 2004. Advisor: H. Bunke.

[6] Simon Günter and Horst Bunke. HMM-based handwritten word recognition: on the optimization of the number of states, training iterations and Gaussian components. *Pattern Recognition*, 37:2069–2079, 2004.

[7] Moisés Pastor i Gadea. *Aportaciones al reconocimiento automático de texto manuscrito*. PhD thesis, Dep. de Sistemes Informàtics i Computació, València, Spain, Oct 2007. Advisors: E. Vidal and A.H. Tosselli.

[8] S. Johansson, G.N. Leech, and H. Goodluck. *Manual of information to accompany the Lancaster-Oslo/Bergen Corpus of British English, for use with digital Computers*. Department of English, University of Oslo, Norway, 1978.

[9] L.M. Lorigo and V. Govindaraju. Offline arabic handwriting recognition: A survey. 28(5):712–724, May 2006.

[10] U. Mahadevan and R. C. Nagabushnam. Gap metrics for word separation in handwritten lines. In *ICDAR '95: Proceedings of the Third International Conference on Document Analysis and Recognition (Volume 1)*, page 124, Washington, DC, USA, 1995. IEEE Computer Society.

[11] U.-V. Marti and H. Bunke. Using a statistical language model to improve the performance of an hmm-based cursive handwriting recognition systems. pages 65–90, 2002.

[12] U.V. Marti and H. Bunke. The iam-database: an english sentence database for offline handwriting recognition. 5(1):39–46, 2002.

[13] Magdi Mohammed and Paul Gader. Handwritten word recognition using segmentation-free hidden markov modeling and segmentation-based dynamic programming techniques. *IEEE Trans. Pattern Anal. Mach. Intell.*, 18(5):548–554, 1996.

[14] Réjean Plamondon and Sargur N. Srihari. On-Line and Off-Line Handwriting Recognition: A Comprehensive Survey. *IEEE Trans. on PAMI*, 22(1):63–84, 2000.

[15] Lawrence Rabiner and Biing-Hwang Juang. *Fundamentals of speech recognition*. Prentice-Hall, 1993.

[16] V. Romero, A. Giménez, and A. Juan. Explicit Modelling of Invariances in Bernoulli Mixtures for Binary Images. In *3rd Iberian Conference on Pattern Recognition and Image Analysis*, volume 4477 of *LNCS*, pages 539–546. Springer-Verlag, Girona (Spain), June 2007.

[17] Giovanni Seni and Edward Cohen. External word segmentation of off-line handwritten text lines. *Pattern Recognition*, 27:41–52, 1994.

[18] J.C. Simon. Off-line cursive word recognition. 80(7):1150–yy, July 1992.

[19] C.Y. Suen, C.P. Nadal, R. Legault, T.A. Mai, and L. Lam. Computer recognition of unconstrained handwritten numerals. 80(7):1162–1180, July 1992.

[20] A. H. Toselli, A. Juan, J. GonzÃ¡lez, I. Salvador, E. Vidal, F. Casacuberta, D. Keysers, and H. Ney. Integrated handwriting recognition and interpretation using finite-state models. *International Journal of Pattern Recognition and Artificial Intelligence*, 18(4):519–539, 2004.

[21] Alejandro Héctor Toselli. *Reconocimiento de Texto Manuscrito Continuo*. PhD thesis, Departamento de Sistemas Informáticos y Computación. Universidad Politécnica de Valencia, Valencia (Spain), March 2004. Advisor(s): Dr. E. Vidal and Dr. A. Juan (in Spanish).

[22] U. v. Marti and H. Bunke. A full english sentence database for off-line handwriting recognition. In *In Proc. Int. Conf. on Document Analysis and Recognition*, pages 705–708, 1999.

[23] U. v. Marti and H. Bunke. Handwritten sentence recognition. In *In Proc. Int. Conf. on Pattern Recognition*, pages 467–470, 2000.

[24] A. Vinciarelli and J. Luettin. O-line cursive script recognition based on continuous density hmm. In *in Proceedings of the 7 th International Workshop on Frontiers in Handwriting Recognition. 2000*, pages 493–498. World Publishing, 2000.

[25] A. Viterbi. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *Information Theory, IEEE Transactions*, 13(2):260 – 269, 1967.

[26] K. Y. Wong, R. G. Casey, and F. M. Wahl. Document analysis system. *IBM Journal of Research and Development*, 26(6):647–656, 1982.

[27] Hanhong Xue and Venu Govindaraju. Hidden Markov Models Combining Discrete Symbols and Continuous Attributes in Handwriting Recognition. *IEEE Trans. on PAMI*, 28:458–462, 2006.

[28] S. Young et al. *The HTK Book*. Cambridge University Engineering Department, 1995.

[29] M. Zimmermann, J.C. Chappelier, and H. Bunke. Offline grammar-based recognition of handwritten sentences. 28(5):818–821, May 2006.

[30] Matthias Zimmermann, Horst Bunke, M. Zimmermann, and H. Bunke. Automatic segmentation of the iam o-line database for handwritten english text. In *in Proceedings of 16 th International Conference on Pattern Recognition*, pages 35–39, 2002.

Índex

Índex de figures

Índex de taules